A STUDY OF SAMPLE ENTROPY TOWARDS PROCESS CAPABILITY

by

ZHENG ZHANG

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Industrial and Manufacturing Systems Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2015

Approved by:

Major Professor
Shing I Chang

Abstract

The process capability is a measurable property of a process related to the specification of a product. Traditionally, process capability analysis (PCA) measurements are expressed by a process capability ratio (PCR). When using a typical PCR to measure process capability, there are certain assumptions, and critics have been made towards PCR, that some the assumptions are violated. Much research has been conducted to ratify the situations when some of the assumptions are violated. This thesis, is going to demonstrate a research towards process capability using Sample Entropy method. The desirable outcome would be that this method can avoid violating the assumptions.

Keywords: Quality Control, Process Capability Analysis, Process Capability Ratio, Sample Entropy

Table of Contents

List of Figures

List of Tables

List of Equations

Chapter 1 - Introduction

A good process control method aims to maintain all production-related elements in a good condition in order to obtain consistent and desirable product quality. Production elements include tools, materials, methods, workers and the combinations (Keller & Pyzdek, 2003). One way to quantify the performance of a process is the measure of process capability.

The process capability is a measurable property of a process related to the specification of a product (Bothe, 1997). A precise and meaningful process capability measurement can provide critical information for process analyst. On the contrary, faulty process capability measurement fails to reveal what is really happening in a process and may lead to incorrect conclusions, which may cause severe consequence. Traditionally, process capability analysis (PCA) measurements are expressed by a process capability ratio (PCR).

1.1 Problem Statement

When using a typical PCR to measure process capability, there are certain assumptions: 1) the quality attribute is normally distributed; 2) the process is statistically in control, and 3) the process mean is centered. Practitioners have emphasized that PCR should be used in a state that the process is statistically in control (Lin and Sheen, 2005), and critics have been made towards PCR, that some the assumptions are violated. Much research has been conducted to ratify the situations when some of the assumptions are violated (Wei *et al* 2009; Pearn *et al*, 2014; Yum and Kim, 2009).

Process capability ratio is widely used in industries, which is an important part of process capability analysis. Montgomery stated the purposes of a PCA study include the following:

1. Predicting how well the process will hold the tolerances
2. Assisting product developer/designer in selecting or modifying a process

3. Assisting in establishing an interval between sampling for process monitoring
4. Specifying performance requirements for new equipment
5. Selecting between competing suppliers and other aspects of supply chain management
6. Planning the sequence or production process when there is an interactive effect of process on tolerances
7. Reducing the variability in a process

PCA should be implemented throughout a product lifecycle, including product design, process design, supply chain management, manufacturing planning, and manufacturing (Montgomery, 2009). Practitioners often use a ratio to quantity PCA results. Kotz (2002) concluded that a state of statistical control should be established before using these indices. Specifically, a univariate quality characteristic (QC) should be independently and identically distributed (i.i.d.) before a PCR index is calculated. If distribution parameters such as mean and/or standard deviation change, the identical distribution assumption will fail. The PCR ratio computation will not be valid.

In order to overcome the disadvantages of PCR, we propose an alternative non-parameter method that can quantitatively measure process parameter changes. The proposed method should acutely identify whether PCR's assumptions are met or not. If the PCR assumptions are violated, the proposed method should be able to pin point the change points in the time series generated for the quality characteristic of interest.

1.2 Potential Solution Approaches

Among several change detection methods, several Entropy methods are most likely to meet our expectation. Entropy has been used as a measure in detecting chaotic time series (Nair, 2014). It treats all time-series indifferently, and reports the stability of a process as reflected by a time series. Though entropy handles non-identically distributed data, it cannot be used for process capability directly. SampEn performs well in identifying variance changes but not mean shifts. In chapter 3 of this research, we explain the discovery of this property in details. Adjustments are needed for implementing SampEn for process capability study.

The fundamental algorithm is derived from Shannon's (1948) theory by Kolmogorov (1998). The original entropy theory was based an infinite time series. Researchers have developed computational entropy heuristics algorithms for finite input data. Approximate Entropy (ApEn) from Pincus (1991), and Sample Entropy (SampEn) by Grassberger (1983, 1988) are most widely used methods. Richman and Moorman (2000) criticized ApEn for outcome inconsistency when sample size differs. In a process capability study, a failure to check the stationary assumption of a production data set will cause inaccurate result, and it may even result in opposite conclusion. The existing Entropy methods introduced so far have potential but can only detect variation changes and are not very consistent.

Grassberger's SampEn algorithm (1983, 1988) produces consistent outcomes for different data lengths. Thus in this research, SampEn is implemented as the foundation of the process capability measurement. Though SampEn is a good tool for detecting noise, it is not able to capture process mean shifts, which is a significant disadvantage because mean shifts are often observed in out-of-control processes. If there is a way to adjust SampEn to enable it to capture mean shifts, it may become a good tool for validating process capability study. In the next section, we will describe the proposed method to enable adjust SampEn for the research question. In chapters 2 and 3, we will introduce and discuss the computational aspect of entropy in more details.

1.4 Expected Results

The decision making related to the use of the proposed method is the following. In cases where only mean shifts are presented in the time series, the combined results from the proposed AdSEn and SampEn would reveal that there are changes in AdSEn but not in SampEn. For variance-change only cases, both AdSEn and SampEn should indicate changes of similar magnitudes. Finally, when both mean and variance have changed, both AdSEn and

SampEn results should indicate changes but with different magnitudes. The proposed AdSEn should indicate a larger entropy change.

We expect that the proposed method will overcome the disadvantage of SampEn algorithm. Specifically, the proposed method should have the following capabilities: 1) AdSEn is able to detect both mean shift and variance change, 2) the numerical measurement has a high level robustness to handle various situations, and 3) the proposed measurement can return a meaningful conclusion regardless of changes in data patterns and distributions.

1.5 Organization

The organization for the rest of this thesis is the following. Chapter 2 describes the deficiency of PCR, and lays the theoretical foundation of the proposed methodology AdSEn. Chapter 3 introduces the proposed methodology and provides a real-world example to demonstrate how the proposed method can be implemented. Chapter 4 presents a simulation study to quantify the properties and robustness of the proposed methodology. Finally, chapter 5 summarizes the proposed research and outlines future research opportunities.

Chapter 2 - Literature Review

In this chapter we will review literature related to process capability analysis and entropy analysis. In section 2.1, we review the literatures on traditional process capability study methods. Section 2.2 discusses the fundamental of entropy, and the choice of the SampEn algorithm over the ApEn algorithm. Finally, we will review other applications of SampEn in section 2.3.

2.1 Process Capability Index Background

Process capability analysis is a vital part of an overall quality-improvement program (Montgomery, 2009). Process capability methodologies have been reviewed by many researchers (Juran, 1974; Kane, 1986; Boyles, 1996; Kotz, 2002; Wu, 2009). Kotz and Johnson (2002) conclude that we need to establish a state of statistical control before process capability ratios (PCR) can be computed. Examples of PCR include C_p (Juran, 1974) , C_{pk} (Kane, 1986) and C_{pc} (Montgomery, 2009). These three PCRs are measurements of uniformity for univariate processes. They are quantitative indices of a process' ability to meet specification requirements under the assumptions that:

1. The quality attribute is normally distributed.
2. The process is statistically in control.
3. The process mean is centered.

We have following PCRs:

$$C_p = \frac{USL - LSL}{6\sigma} \qquad \text{Equation 2.1}$$

where

USL: upper spec limit,

LSL: lower spec limit, and σ is the standard deviation of the QC.

There are also cases that measure one-sided specification limits:

$$C_{pu} = \frac{USL - \mu}{3\sigma} \qquad \text{Equation 2.2}$$

$C_{pl} = \frac{\mu - LSL}{3\sigma}$ Equation 2.3

Montgomery (2009) states that those assumptions are critical to the analysis result. Misusing PCR will lead to invalid results and induce inaccurate conclusions. For example, when a process mean deviates from the center (target), C_p will remain the same value. Note that the third assumption is violated in this case. C_{pk} obeys the assumptions 1 and 2, but relaxed on the third assumption, so it can measure the stability for non-centered processes. In other word, C_{pk} can measure mean shifts in a process.

$C_{pk} = min\ (C_{pu}, C_{pl}) = \frac{min\ (USL-\mu,\ \ \mu-LSL)}{3\sigma}$ Equation 2.4

There are research works relaxing the normal assumption. Practitioners use $\mathrm{C_{pc}}$ to measure non-normally distributed processes. In Equation 2.5, notation X means a series of input variable, $T = \frac{1}{2}(LSL + USL)$ is the process target value. In practice, $6\sqrt{\frac{\pi}{2}}$ is usually set to 7.52 to represent a 6σ control level.

$C_{pc} = \frac{USL-LSL}{6\sqrt{\frac{\pi}{2}}E|X-T|}$ Equation 2.5

However, though constraints are relaxed, all existing PCRs assume that the estimated variance remains the same throughout the entire calculation. In other word, this is an assumption that all observations are from the same distribution although they may be collected over a long period of time. In a real production system, this assumption is rarely true.

2.2 Entropy Background

Entropy has been utilized to measure how chaotic a signal is (Nair, 2014). Shannon (1948) was the first to use the concept of entropy on information theory. Kolmogorov (1998) laid the theoretical background for entropy algorithms. He stated the following assumptions: 1) there exists a constant *p*, *p*ϵ *[0, 1]*; 2) an N sized random table *T* is called *(n,ε)* in $\mathbb{R}_N$, *A=R(T)*, R$\epsilon\mathbb{R}_\mathrm{N}$ with number of elements *V≥n*; 3) the frequency $\pi = \frac{1}{v}\sum_{k\epsilon A} t_k$ satisfies the inequality $/\pi(A - p)/\leq\varepsilon$.

Define (n, ε, p) as the randomness, then there holds the following two theorems:

Theorem 1: If the number of elements of the system $\mathbb{R}_{\mathrm{N}}$ does not exceed $\tau(n,\varepsilon)=\frac{1}{2}e^{2n\varepsilon^2}$ then for any *pϵ [0, 1]*, there exists a table T of size N that is *(n, ε, p)* –random with respect to $\mathbb{R}_N$.

Theorem 2: If $k \leq \frac{(1-2\varepsilon)}{4\varepsilon}, n \leq (k-1)m, and\ N \geq km$, then

$$\boldsymbol{\tau}\left(\boldsymbol{n},\boldsymbol{\varepsilon},\boldsymbol{N},\frac{\mathbf{1}}{\mathbf{2}}\right) \leq \boldsymbol{k2^m} \qquad \text{Equation 2.6}$$

In the original Kolmogorov entropy theory, the length of data taken into consideration is approaching infinity, which is not practical for real-world applications such as quality control. Pincus (1991) proposed an Approximate Entropy (ApEn) to measure system complexity in term of process changes. Each data set should contain at least 1000 data points. ApEn(m, r) takes a positive integer number m as the size of the vector, and a positive number r as the threshold, in term of the size of the overall standard deviation from the process. Pincus (1991) defines that a time series $\boldsymbol{x}$ with equal time interval has N points where $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_N})$. We can divide $\boldsymbol{x}$ into N-m+1 vectors. For each m dimensioned vector, namely $\boldsymbol{v_i} = (\boldsymbol{x_i}, \boldsymbol{x_{i+1}}, \dots, \boldsymbol{x_{i+m-1}})$, 1≤i≤N-m+1. The distance between $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$ is defined as: $d[\boldsymbol{v_i},\ \boldsymbol{v_j}] = max\ (|x_{i+k-1} - x_{j+k-1}|)$, where kϵ[1,m]. Define:

$$\boldsymbol{C_i^m(r)} = \frac{\boldsymbol{the\ count\ of\ (d[v_i,v_j]\leq r)}}{\boldsymbol{N-m+1}} \qquad \text{Equation 2.7}$$

as the count of distance less than r of vector i of m dimensions divided by the number of different vectors N-m+1 and

$$\boldsymbol{C^m(r)} = \frac{\boldsymbol{\sum_1^{N-m+1} C_i^m(r)}}{\boldsymbol{N-m+1}} \qquad \text{Equation 2.8}$$

as the accumulation of $\mathrm{C_i^m(r)}$. Then

$$\Phi^{\boldsymbol{m}} = \boldsymbol{log}\frac{\boldsymbol{\sum_1^{N-m+1} C_i^m(r)}}{\boldsymbol{N-m+1}} = \boldsymbol{log}\ [\boldsymbol{C^m(r)}] \qquad \text{Equation 2.9}$$

The numerical result of ApEn can be calculated by:

$$\boldsymbol{ApEn(m,r)} = \Phi^{\boldsymbol{m+1}} - \Phi^{\boldsymbol{m}} \qquad \text{Equation 2.10}$$

The computation of ApEn can be accomplished by a function approx_entropy() under pracma package in a computer language R (package 'pracma') for calculation purpose (Borchers, 2014).

Richman (2000) and Lake (2002) criticized ApEn for outcome inconsistency when sample size differs. A possible remedy is the use of sample entropy (SampEn) proposed by Grassberger (1983, 1988). Richman (2000) and Lake (2002) made a comparison of SampEn to approximate entropy and pointed out that SampEn does not include self-match, which means the distance defined by (Pincus, 1991) $d[\boldsymbol{v_i},\ \boldsymbol{v_j}] = max\ (|\boldsymbol{x_{i+k-1}} - \boldsymbol{x_{j+k-1}}|)$ does not consider the case when i=j in SampEn. That is the equations (2.8) and (2.9) can be revised to the Equations 2.11 and 2.12.

$$\boldsymbol{C_i^m(r)} = \frac{\boldsymbol{the\ count\ of}\ (\boldsymbol{d}[v_i,v_j] \leq r)}{\boldsymbol{N-m}} \qquad \text{Equation 2.11}$$

$$\boldsymbol{C^m(r)} = \frac{\sum_1^{N-m} C_i^m(r)}{\boldsymbol{N-m}} \qquad \text{Equation 2.12}$$

Similar to ApEn, this counting mechanism enables SampEn to detect variance changes. The difference in $max\ (|\boldsymbol{x_{i+k-1}} - \boldsymbol{x_{j+k-1}}|)$ records the largest difference in neighbor vectors $\boldsymbol{v_i}$ and $\boldsymbol{v_j}$. The selected threshold r is a 0 to 1 fraction of overall standard deviation of the dataset. If the maximum distance is greater than a selected threshold r, that Equation 2.11 will not record this count. If the sample dataset becomes noisier (i.e. with a higher standard deviation) as time progresses, a larger $d[\boldsymbol{v_i},\ \boldsymbol{v_j}]$ would occur, meaning that the less counts will be recorded for $d[\boldsymbol{v_i},\ \boldsymbol{v_j}] < r$. Equation 2.13 provides the numerical result of SampEn computation.

$$\boldsymbol{SampEn(m,r)} = -\boldsymbol{log}\left(\frac{C^{m+1}(r)}{C^m(r)}\right). \qquad \text{Equation 2.13}$$

According to Richman (2000), SampEn output shows higher consistency in different data length than ApEn. For the purpose of change detection in process parameters, we would require a reliable tool. Consequently, SampEn should be chosen over ApEn in this research. There is also a function sample_entropy() available under the pracma package, for performing SampEn algorithm calculation (Borchers, 2014).

Multi-scale Entropy (MSE) proposed by Costa (2002), is based on a multiple time scale concept. The author utilized the concept of multiple time series in one spanned physical system. The idea is to divide a time series into t equal length segment, and performed SampEn algorithm to each segment. Given a one-dimensional discrete time series, $\{\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_n}\}$, he constructed consecutive coarse-grained time series, $\{\boldsymbol{y}^{(\mathbf{t})}\}$, determined by the scale factor, t, according to the equation: $y_j^{(\tau)} = \frac{1}{t}\sum_{i=(j-1)\cdot \mathrm{t}+1}^{j\cdot \mathrm{t}} x_i$. For scale $\mathrm{t} = 1$, the time series $\{\boldsymbol{y}^{(\mathbf{1})}\}$ is simply the original time series. SampEn was calculated in an integral of a series of continuous variable x. In our research, we are going to adopt the concept of multiscale to provide more information for each segment, but the analytical computation will not be identical to the original method.

It is a good idea to examine the entire time series for detecting out-of-control portions. This can be accomplished by implementing analysis on various segments of this time series. The numerical outputs would allow the detection of heterogeneous distributions with a time series. Thus in our research, the multi-scale concept is adapted, and SampEn will be utilized as the numerical analytic tool on segments of the entire process data set or time series.

2.3 Entropy Applications

SampEn applications can be found in physical and biological applications. Aboy (2007) implemented SampEn on biomedical analysis. This characterization study provides additional insights regarding the interpretability of SampEn in the context of biomedical signal analysis.

In manufacturing, entropy has been used as a diagnostic tool. Liu and Han (2014) utilized multiscale SampEn (MSE) for roller bearing fault diagnosis, more specifically, to the experimental bearing vibration signals analysis. In this article, MSE result is not the only tool used, the author implemented the MSE result as a summary statistics for BP (back propagation) neural network model. A similar application of multiscale entropy on fault diagnostics can be found within manufacturing practitioners in recent years, such as Wu, et. al. (2012).

Inspirationally, Martı´nez-Olvera (2012) has utilized entropy as a tool of quality assurance. The author proposed the use of entropy as an indicator measuring manufacturing complexity by developing an indicator of the impact the BOM (bill of material) structure. He also provided an entropy-based complexity measure that allows an objective mean of comparing system performance. Though not using SampEn, his adoption of entropy to measure enterprise resource management, manufacturing complexity, and quality assurance is very close to goal of the proposed research of this thesis.

Another motivational research is from Oprean and Bucur (2012). Based on Shannon's original entropy guideline, he developed quality-entropy algorithm. This research identified two sources of quality entropy – one defined by Markovian and the other by Bernoulli properties. The quality entropy computations allow multiple quality characteristics to be summarized at a specific time. The simulation is successful in accessing tolerance in that whether the quality characteristic is perceived as tolerable or not. Since one of the assumptions is based on Markovian property to estimate probabilities, it may become restrictive for some applications, for example, extending the application to machine learning.

2.4 Literature Review Conclusion

From our literature search, we can conclude that:

1) PCA is critical to an enterprise;

2) Practitioners have started using entropy in the field of quality assurance; and

3) SampEn is not widely used in manufacturing.

In this study, we aim to explore the feasibility of utilizing SampEn towards a PCA study. The existing SampEn algorithm is not yet capable of handling PCA yet. Basic research is needed to enabling it.

Chapter 3 - Proposed Methodology: Input Adjusted SampEn

The goal of this research is to study the use of SampEn towards process capability analysis. We have discovered that the algorithm based on Equation 2.11-2.13, SampEn is good at detecting variance changes. It records the cumulative probability of data point falling in a threshold when vector length is set to m. As explained in section 2.2, this cumulative mechanism is based on the overall variance of a time series, and it enables SampEn to detect variance changes. However, the algorithm developed by Grassberger (1988) does not have any element related to mean level shifts detection. For example, consider a time series shown in Figure 3.1 where the simulated data are from three different distributions. The first data segment is from *N(0, 1)*, the second from *N(1, 1)*, and *N(2, 1)* for the third. This distribution combination is repeated 100 times and the values of SampEn are generated for each segment. Figure 3.2 shows the boxplots of these SampEn values according to their segments. It is clear that the entropy levels do not differ much from segment to segment. Consequently, SampEn cannot be applied directly for process capability measurement. Adjustments are needed to enable SampEn to detect mean shifts.

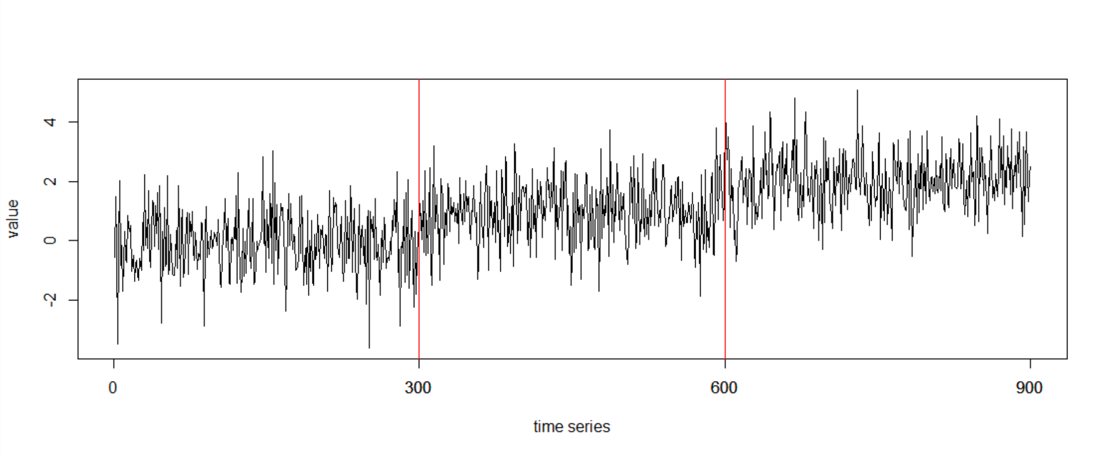

Figure 3-1 Example time series dataset, distributed as *N(0, 1)*, *N(1, 1)*, and *N(2, 1)*

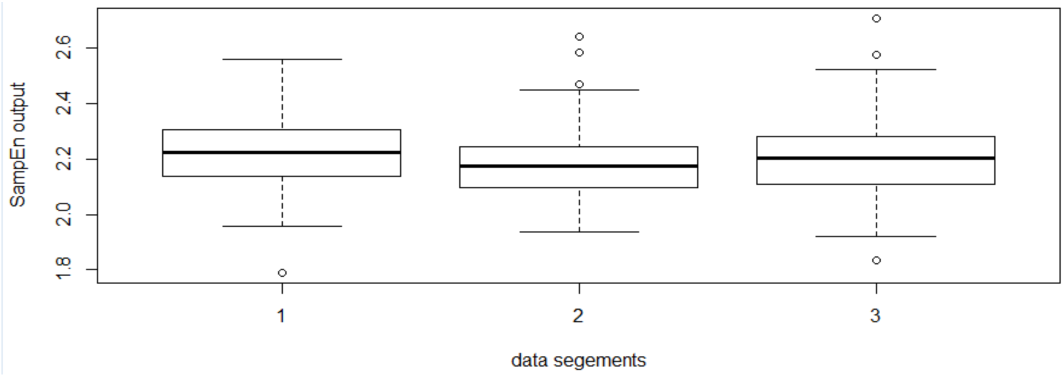

Figure 3-2 Boxplots from 100 replicated experiments for the data distributions of Figure 3.1.

We will consider two approaches to tackle this problem:

1. **Cumulative Sliding Window and Multi-Scale Approach:** Implement SampEn algorithm on cumulative sliding windows (Du, 2014) and apply multi-scale mechanism (Costa, 2002) on the time series of interest. By studying the outcome of SampEn algorithm, we may be able to determine whether this approach can be applied for process capability studies or not.
2. **Input Transformation Approach:** Modify the inputs to SampEn to enable the SampEn algorithm to detect both mean and variance abnormity.

In the following sections in chapter 3, we will discuss each approach in details.

3.1. Cumulative Sliding Window and Multi-Scale Approach

The most direct approach is to apply SampEn computation on different parts of a time series. This proposed approach follows these steps:

1. Break the entire dataset into several equal length subsets assuming the first subset $\boldsymbol{x_1}$ is the template,
2. Select the threshold $\boldsymbol{r}$ according to $\boldsymbol{x_1}$'s standard deviation,
3. Calculate the SampEn of an initial small segment data denoted as $\boldsymbol{e_1}$. When the next segment of data is available, it is merged into $\boldsymbol{e_1}$, denoted as $\boldsymbol{e_2}$. Again, we calculate the SampEn on $\boldsymbol{e_2}$. In a cumulative fashion, all entropies of entire dataset can be calculated on $\boldsymbol{e} = (\boldsymbol{e_1}, \boldsymbol{e_2}, \dots, \boldsymbol{e_k})$ assuming k segments in the dataset under study.

By cumulative sliding window, we mean that for example, in sample data set shown in Figure 3-3, $\boldsymbol{e_1}$ represents the SampEn output of the first segment, from 1 to 300 in Figure 3-1, $\boldsymbol{e_2}$ is the time series from 1 to 600, and $\boldsymbol{e_k}$, where k=3 is the segment from 1 to 900. The goal of this exercise is to make sure that the process is stationary, i.e., the process mean and standard deviation stay unchanged. Under the stationary condition, the result from existing process capability study is valid.

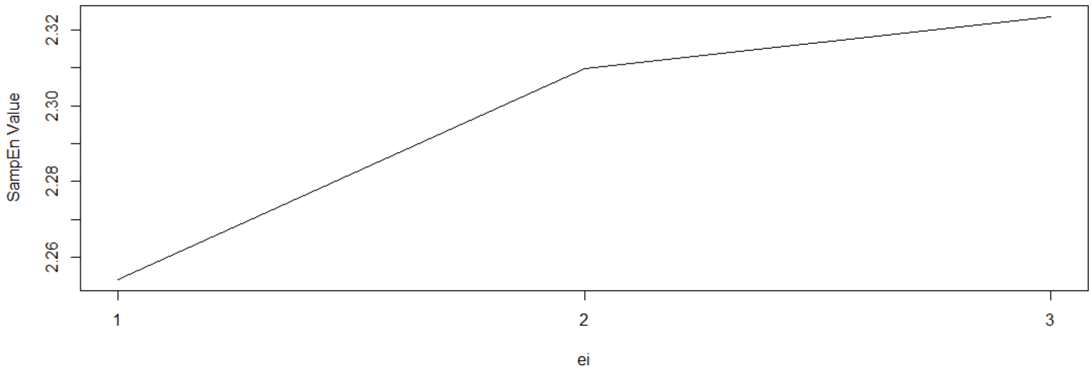

Figure 3-3 SampEn output with cumulative sliding window method for $\boldsymbol{e_i}$

This approach has the ability of detecting mean shift in most scenarios, but it cannot return a conclusive quantitative result consistently because the entropy output values depend on both data lengths and data change patterns. We have found that the longer the mean shift portion is, the higher the value will be. The difference can be seen from Figures 3.4 and 3.5. Dataset for both figures are consisted of two segments, 900 data points in total. The first segment is *N(0, 1)*, and *N(1, 1)* for the second. The only difference is that the data length of the segments. For Figures 3.4 and 3.5, the first segment is 600, the second 300, and for Figure 3.6 and 3.7, both segments are 450 in length. According to the different patterns revealed from Figure 3.4 and 3.5 we can see that this fixed length cumulative sliding window will be subset data length dependent, the output does not return a consistent evaluation of a process.

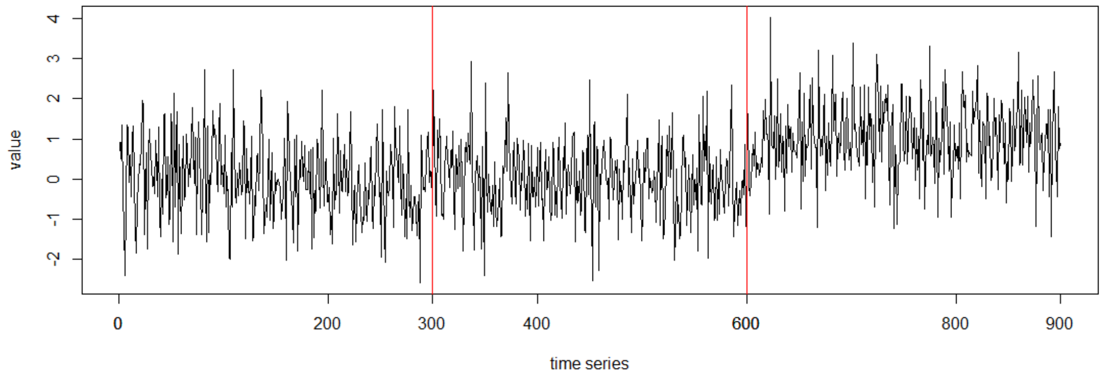

Figure 3-4 Time series plot of *ts=c(rnorm(600,0,1),rnorm(300,1,1))*

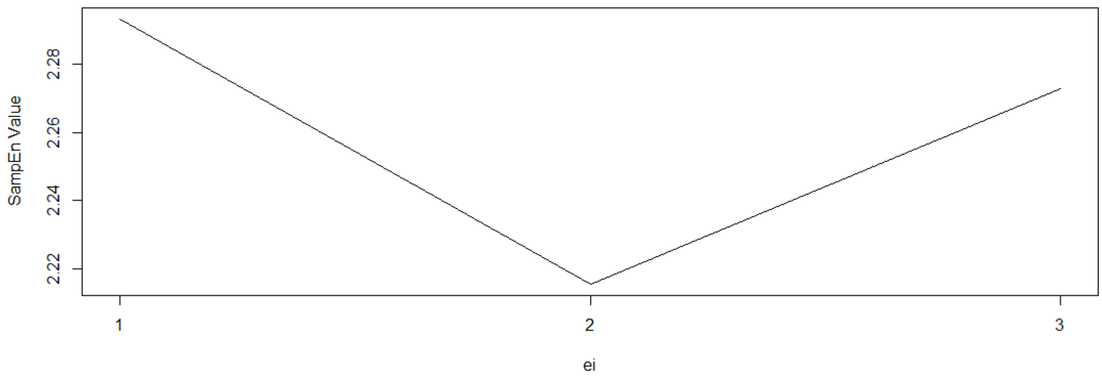

Figure 3-5 Dataset consisted of first 600 points from *N(0, 1)* distribution and followed with 300 points from *N(1,1)*

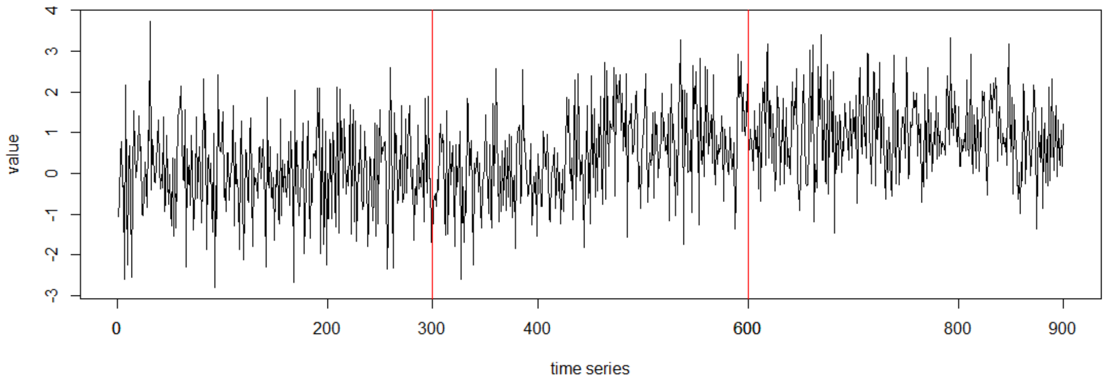

Figure 3-6 Time series plot of *ts=c(rnorm(450,0,1),rnorm(450,1,1))*

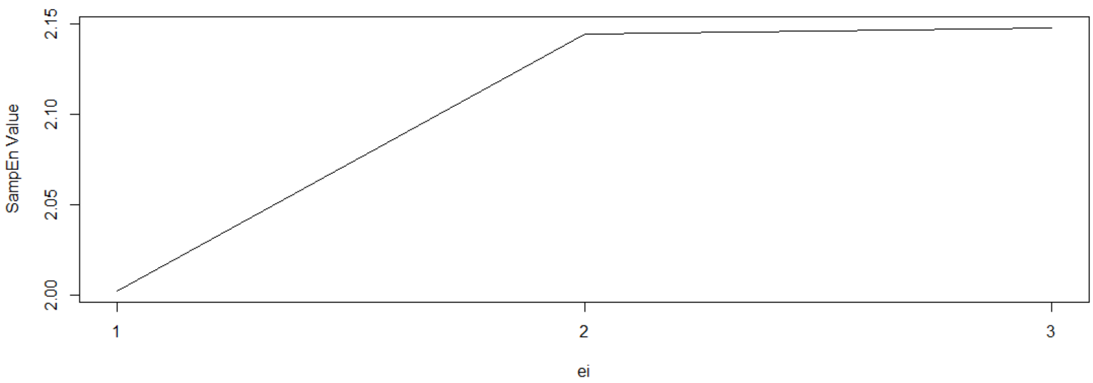

Figure 3-7 Dataset consisted of first 450 points from *N(0, 1)* distribution and followed with 450 points from *N(1,1)*

Regarding to the change patterns, the case under consideration is the mean level shift up *x* unit, and then drops down 2*x* unit so that each segment has ±x unit mean shift from original mean level as shown in Figure 3.8 and 3.9.

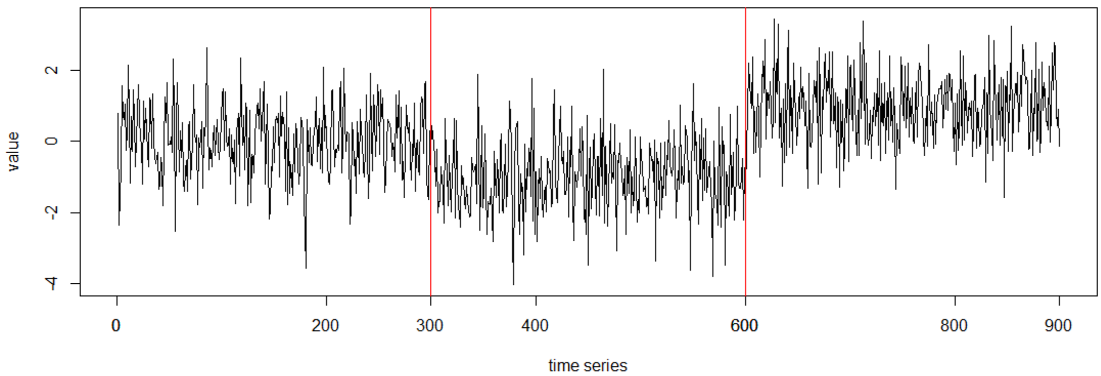

Figure 3-8 This dataset is consisted of 3 segments, the first is from *N(0, 1)*, followed with *N(-1,1)*, and thirdly *N(1,1)*

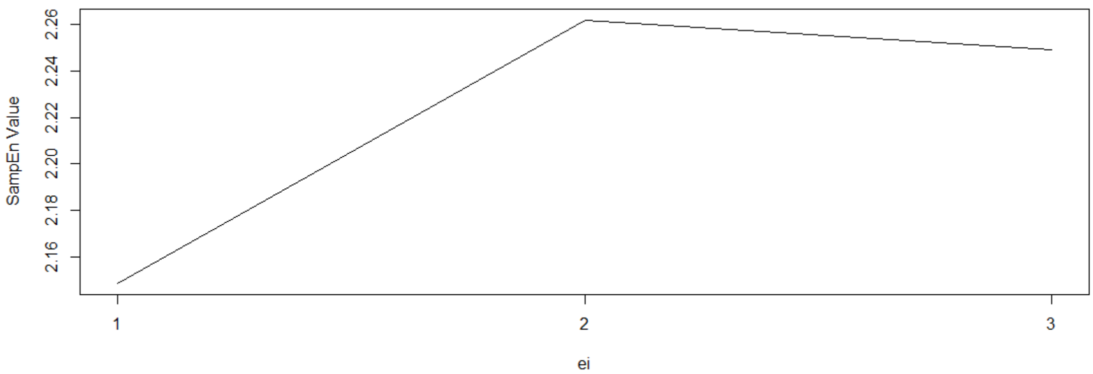

Figure 3-9 SampEn output of dataset from Figure 3.6

In Figure 3.9, though larger mean shifts in the last segment, the SampEn computation results show that the entropy value of $\boldsymbol{e_3}$ (1-900 observation) is smaller than that of the second subset $\boldsymbol{e_2}$ (1-600 observations). So this cumulative-entropy computational approach will not be able to distinguish the magnitude of changes in this case. This example also confirms that the existing SampEn algorithm cannot be used for detecting mean level changes within a time series.

Another alternative is using the concept of multi-scale method (Costa 2002), that is, to take each segment individually instead of cumulatively. However, revisiting Figure 3.1 as an example of a time series with three segments, we observe that, the second and third segments return the same entropy values as the template since there is no change in variance. The multi-scale may be helpful in detecting variance changes, but not mean shifts. Thus other approaches should be considered to make SampEn a tool for process capability.

3.2 Input Adjustment Approach

In order to enable SampEn to be sensitive to time series mean shifts, we propose to make adjustments to the input data. The proposed strategy is to embed a mean-level change into the existing input data stream. Let the transformed value y be a representation of the original dataset x, thus both mean and variance changed of x can be represented in SampEn outcome of y. We now consider several possible alternatives for transforming the raw time series x into y.

3.2.1. Multiplier Transformation

Assume data set $\boldsymbol{x} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_m}\}$ has a mean of μ, and standard deviation σ, then when $y = c \cdot \boldsymbol{x}$ where c is the multiplier that transforms the raw data $\boldsymbol{x}$ into $\boldsymbol{y}$. The mean of $\boldsymbol{y}$ will be $c \cdot \mu$. Note that $Var(\boldsymbol{y}) = Var(c\boldsymbol{x}) = c^2 Var(\boldsymbol{x})$, and, therefore: $Std(\boldsymbol{y}) = c \cdot Std(\boldsymbol{x})$. Here μ can be either the process target value or the sample mean of template dataset. This fact lays the foundation of $y = c \cdot \boldsymbol{x}$ transformation when this multiplier c is a constant. It is easier to represent mean shift in term of target μ, or the estimated mean of the template data subset.

A. Plan 1: Multiplier $c_i = \frac{\mu_i}{\mu}$

In this method, μ is defined as expected mean of input variable $\boldsymbol{x}$, and $\overline{x_i}$ is the estimated mean of the *i*th segment. By multiplying c, it is transformed into y. The variance change in x will be reflected in y according to $Std(y) = c_i \cdot Std(x)$, and the mean shift can be represented by this transformation via multiplier $\frac{\bar{x}}{\mu}$. Note that $c_i = \frac{\overline{x_i}}{\mu}$ contains the element of mean, so any deviation from μ can be captured by the ratio $\frac{\overline{x_i}}{\mu}$. The multiplier will pass the mean deviation in x into y with the relationship $Std(\boldsymbol{y}) = c_i \cdot Std(\boldsymbol{x})$, thus mean shift becomes detectable with this multiplier transformation.

However, $\frac{\overline{x_i}}{\mu}$ multiplier has disadvantages. First, when the target μ is 0, y will always be infinity. Plan 1 also suffers another scale issue in that the mean levels are different for different problems. There will be unintended results just due to the magnitude of means. For example, μ =10000, then a shift to 10,001 will result in no effect because $\frac{\overline{x_i}}{\mu}$ will be very close to one, the transformed input variable y will be almost identical to $\boldsymbol{x}$, the input variable before transformation. But if μ =1 then an increase to 2 would double the magnification effect. In this scenario, the use of $\frac{\overline{x_i}}{\mu}$ as the multiplier will fail, so other transformation methods should be considered.

It is important to point out that in multiplier transformation, c_i has to be a constant for equation $Std(\boldsymbol{y}) = a_i \cdot Std(\boldsymbol{x})$ to stand. In our research, $\overline{x_i}$ is estimated from the *i*th data subset. However, by the time of transformation, $\overline{x_i}$ is a known value, consequently, the multiplier $\frac{\overline{x_i}}{\mu}$ is a constant in transformation, $Std(\boldsymbol{y}) = c_i \cdot Std(\boldsymbol{x})$ holds.

B. Plan 2: Multiplier $c_i = \frac{\mu_i - \mu}{\sigma}$

In another approach in addressing the scale issue of the multiplier $\boldsymbol{y}$ can also be transformed from $\boldsymbol{x}$ with the multiplier $\frac{\mu_i - \mu}{\sigma}$. In this case, μ_i can be estimated by $\overline{x_i}$. The mean level shift can be expressed by $\overline{x_\iota} - \mu$ in terms of process standard deviation σ. Comparing to plan 1, the transformed output y will not be over-magnified. However, this approach still does not solve the issue of possible zero for the multiplier when mean difference, i.e., $\overline{x_\iota} - \mu$ is 0.

3.2.2. Cox-Box Transformation

In multiplier transformation approach, we can notice that when the multiplier $\overline{x_\iota} - \mu$ is 0, y will become 0, such that the transformation fails to represent what is happening in input variable x_is. It is possible to apply a transformation that the power factor makes justifications of the input data. Box-Cox transformation, $y = \boldsymbol{x}^p$ (Mayers, 2009) is an empirical method used for power family transformation. It turned out in our research that when we select p to be $\frac{\overline{x_\iota}}{\mu}$, it does not avoid the case when $\mu = 0$, the output is not available. It does not solve the problem that $\frac{\overline{x_\iota} - \mu}{\sigma}$ =0 neither because when $\frac{\overline{x_\iota} - \mu}{\sigma}$ =0, the output of $\boldsymbol{x}^p$ will be constantly 1, which is not a desirable result.

Learned from sections 3.2.1 and 3.2.2, the solution is not a matter of Multiplier transformation, but the multiplier itself is the key. It has to avoid the multiplier equals to 0 cases, and does not over magnify the original dataset. Cox-Box transformation, is likely to exaggerate dataset when $\boldsymbol{x} > 1$, and *p* is greater than 1, $\boldsymbol{y} = \boldsymbol{x} \cdot \frac{\overline{x_\iota}}{\mu}$ may also overly enlarge the variance from $\boldsymbol{x}$ into $\boldsymbol{y}$. Plan 2, $y_{ij} = x_{ij} \frac{\overline{x_\iota} - \mu}{\sigma}$, seems to be an improvement over plan 1. However, we still need to overcome the drawback of “negative magnification” when there is no mean shift.

3.3. Proposed Transformation

The proposed transformation is

$$\boldsymbol{y_{ij}} = \boldsymbol{x_{ij}}\left(\left|\frac{\boldsymbol{\mu_i} - \boldsymbol{\mu}}{\boldsymbol{\sigma}}\right| + \mathbf{1}\right) \qquad \textbf{Equation 3.1}$$

Let $\delta_i = \boldsymbol{\bar{\mu}_i - \mu}, \boldsymbol{y_{ij}} = (\frac{\delta_i}{\sigma} + 1) \times \boldsymbol{x_{ij}}$. For this equation to stand, δ_i has to be a constant. As discussed in section 3.2.1, the proposed method is a retro-perspective analysis. That is, variable μ_i can be estimated by $\overline{x_\iota}$. In practice, $\overline{x_\iota}$ is can be calculated, thus $\overline{x_\iota}$ is a known value, such that δ_i becomes a constant, and Equation 3.1 can be recognized as a format of $\boldsymbol{y} = c \cdot \boldsymbol{x}$ transformation. Consequently, it will have the property that $Std(\boldsymbol{y}) = c \cdot Std(\boldsymbol{x})$.

$\boldsymbol{y_{ij}} = \boldsymbol{x_{ij}}(\left|\frac{\mu_i - \mu}{\sigma}\right| + \boldsymbol{1})$ transformation based on plan 2, $y_{ij} = x_{ij}\frac{\mu_i - \mu}{\sigma}$. We propose to add a constant 1 and the absolute value of $\frac{\overline{x_\iota} - \mu}{\sigma}$. The absolute value is used because both increase and decrease in mean levels are considered changes. When $\overline{x_\iota} - \mu = 0$, it will maintain the equivalence $y_{ij} = x_{ij}$, which is the desired result. With this method, σ_y is equal to $\sigma_x \cdot (\left|\frac{\bar{x}_i - \mu}{\sigma}\right| + 1)$, both mean and variance changes in a dataset x will be reflected in σ_y. Note that $E(y_{ij}) = E(x_{ij})(\left|\frac{\bar{x}_i - \mu}{\sigma}\right| + 1)$. As discussed in 3.2.1, $(\left|\frac{\bar{x}_i - \mu}{\sigma}\right| + 1)$ is a known constant, thus $E(y_{ij}) = E(x_{ij})(\left|\frac{\bar{x}_i - \mu}{\sigma}\right| + 1)$ stands, and the mean shift in original dataset x_i can be transformed to the variance of y_i via $Var(y_i) = (\left|\frac{\bar{x}_i - \mu}{\sigma}\right| + 1)^2 \cdot Var(x_i)$.

The SampEn algorithm (Equation 2.5) will be able to detect both mean shifts and variance changes via adjusted input y using a multi-scale method by 1) dividing the entire dataset into segments; 2) plotting out the SampEn value from each segment; and 3) identifying pattern changes according to the entropy values from all segments in orders.

The flowchart of implementing this transformation is shown in Figure 3.3. The following steps describe how the proposed entropy method can be used to identify process mean and variance changes in details:

1. Import data from collection devices
2. If there is a known target and control standard deviation, use the existing μ and σ. Otherwise, select a reference sub dataset that is deemed representative of the process under study and calculate the sample μ and σ in the transformation function. To avoid

overestimating the within-sample variation, we use a sub sample size of n observations to compute a range=X_{max}-X_{min} and then compute $\hat{\sigma} = \bar{R}/d_2$ where $\bar{R}$ is the average of all ranges in this representative sub dataset and d_2 is a constant related to n (Montgomery, 2009). For example, a segment has 100 observations. If a sample size n=5 is used to compute the ranges, twenty ranges are available to compute $\bar{R}$. In this case, $d_2 = 2.326$. Then the process mean estimate is the sample mean and the process standard deviation is $\hat{\sigma} = \bar{R}/d_2$.

3. Input Transformation: Transform $\boldsymbol{x}$ into $\boldsymbol{y}$, that is, $y_{ij} = x_{ij}(\left|\frac{\bar{x}_i-\hat{\mu}}{\hat{\sigma}}\right| + 1)$.
4. Standardize y_{ij}: use $Y_{ij} = \frac{y_{ij}-\hat{\mu}}{\hat{\sigma}}$.
5. Set manipulation parameter for SampEn. Users can define parameters including threshold *r*, length m of comparing vector, delay τ, and select multi-scale resolution *k*, which determines how many subsets in the original dataset will be broken into. As explained in section 2.2, threshold r sets the selection distance between vectors, *m* is the dimension of the vector τ set the way of selecting vectors (for best use of full size data set, τ is recommended to be 1), and also the scale *k* is the number of slices that users want to segment into. In application, If user does not have a specific concept of the parameters, he/she only has to define scale number k, the rest of the parameters will be set as *r*=0.2, *m*=2, and τ=1 as default.
6. Perform SampEn calculation and present multi-scale plots with different *r's*.
7. If the data pattern plot is consistently revealed from the multiscale plot, terminate this algorithm and conclude that the one distribution assumption is violated. Otherwise, increase the resolution parameter, by choosing higher scales to see the dataset in more details. Example can be seen in chapter 4. For instance, in section 4.3, from Figure 4.10 to 4.13, we can see that patterns appear consistent when scale k =4, 8, 12, and 16. The example in chapter 4 is a series of boxplots because multiple simulations are run. In real

cases, each segment has only one entropy value. A dotted line can be used to connect these values to reveal the pattern which represents the changes of process parameters.

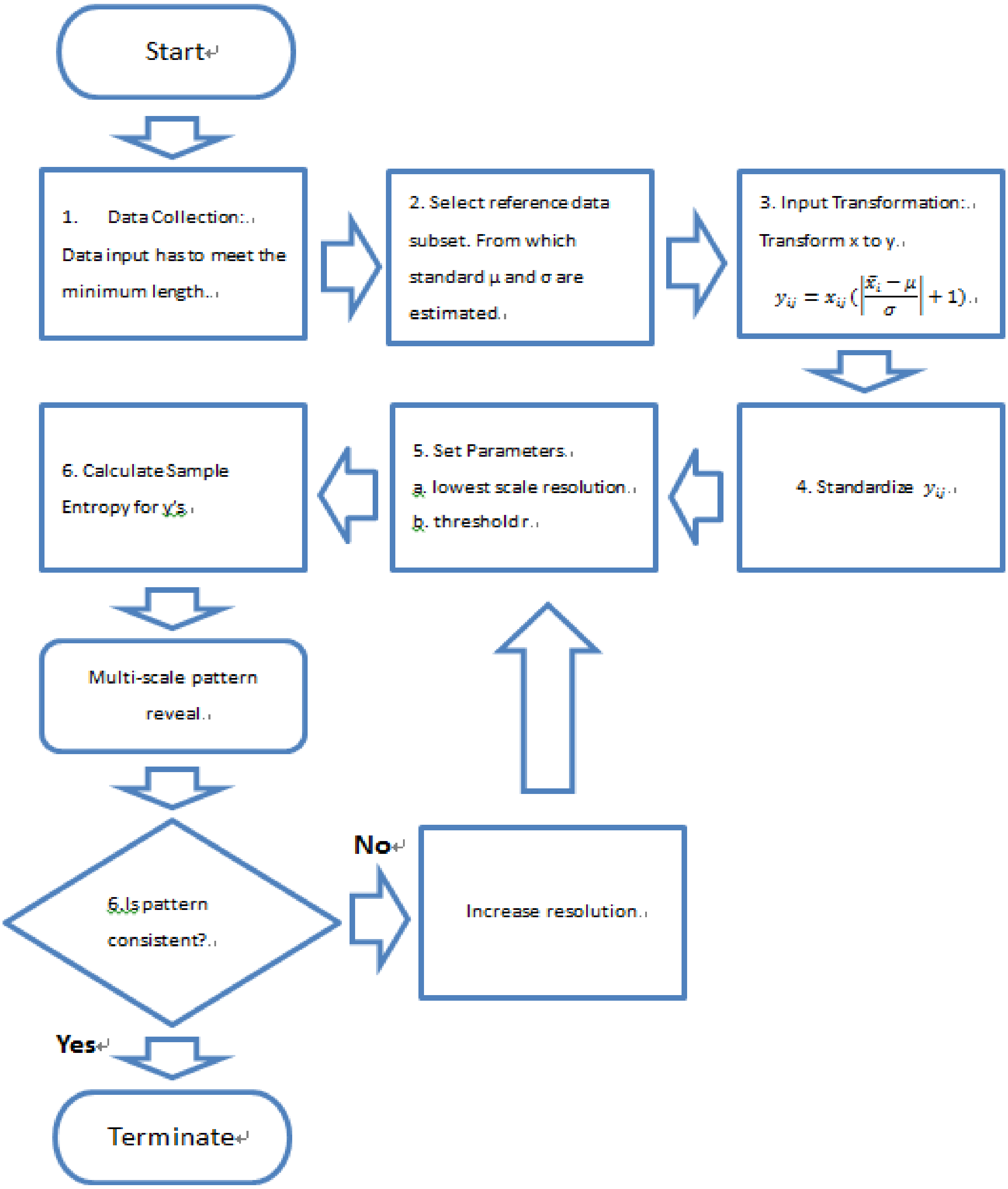

Figure 3-10 Adjusted SampEn WorkFlow Chart

3.4 Properties of the Proposed AdSEn

Unlike C_{pk}, AdSEn does not have strict assumptions. It handles data from different distributions. However in our research, examples are all performed in normal distributed simulation data. AdSEn has some useful properties. In this section, we will discuss in details that the how the changes of $\boldsymbol{x}$ impact $\boldsymbol{y}$. Let's assume that x_0~$N(\mu_0, \sigma_0{}^2)$, and x_i~$N(\mu_i, \sigma_i{}^2)$. In the proposed transformation $y_{ij} = x_{ij} \cdot (\left|\frac{\mu_i - \mu_0}{\sigma_0}\right| + 1)$, μ_i is estimated by $\bar{x}_i$, where x_i is the i^{th} subset (or segment) of the process data, and x_{ij} is the j^{th} element in vector $\boldsymbol{x_i}$.

Data template $\boldsymbol{x_0}$ has transformed to $\boldsymbol{y_0}$. Since $\boldsymbol{x_0}$ has the estimated mean of μ_0, thus $\boldsymbol{y_0} = \boldsymbol{x_0}$. Any other dataset $\boldsymbol{x_i}$ will be transformed to $\boldsymbol{y_i}$, so $\boldsymbol{y_i}$ is the transformed subset for of $\boldsymbol{x_i}$ i.e. segment i, and y_{ij} is the corresponding transformed observation x_{ij}. Consequently, we will have a relationship for the standard deviation of $\sigma_{y_{ij}}$, which is the variance of y_{ij} and σ_0. If μ_1 has an increase of $h \cdot \sigma_0$ shifted from μ_0, $\mu_1 = \mathrm{h}\sigma_0 + \mu_0$ and $\sigma_1 = g\sigma_0$, then we can have $\sigma_{y_i} = \left(\left|\frac{\mu_1 - \mu_0}{\sigma_0}\right| + 1\right) \cdot \frac{\sigma_1}{\sigma_0} \cdot \sigma_{y_0}$.

Performing standardization for transformed variable $Y_{ij} = \frac{y_{ij} - \mu_0}{\sigma_0}$, such that the template dataset will have a mean of 0, and standard deviation of 1, equivalent to $\sigma_{Y_i} = (h + 1) \cdot g$. Thus the relationship will be defined as:

$$\boldsymbol{\sigma_{Y_i} = h \cdot g + g} \qquad \textbf{Equation 3.2}$$

If mean level shift h is zero i.e. mean unchanged, the standard deviation of the transformed variable y_{ij} stays unchanged σ_0 when $\boldsymbol{g}$ =1. Therefore, $\boldsymbol{h}$ =0 and $\boldsymbol{g}$ =1 is the case when the process is stationary. On the other hand, if there is any increase of $\boldsymbol{h}$ or $\boldsymbol{g}$, the net result is the increase of variation as reflected in Equation 3.2. Specifically, if $\boldsymbol{h}$ increases, σ_{y_i} will increase, and so that the SampEn output will increase. The change can be revealed in the final AdSEn result. Similarly, if g increases, the σ_{y_i} also increases.

We now consider a case where the historical process and variance are not available but the first segment of the time series is used as the template. The mechanism is to divide the entropy value of $\boldsymbol{y_i}$ by the entropy of $\boldsymbol{y_0}$ of the first segment. Specifically, a boxplot is plotted; the median is used to define the critical value. However, the input datasets can be very random. It is very difficult to precisely predict the distribution of each subset. It is difficult to draw any conclusion from one SampEn output to make conclusions without any statistics about the proposed SampEn.

We propose to express the change in a confidence percentage. The proposed method is to take the SampEn value $\boldsymbol{e_i}$ of each segment, and divide it by the SampEn of the template dataset $\boldsymbol{e_0}$, denote the ratio of i^{th} segment as ρ_i.

A simulation is replicated 1000 times according to different r values. Standard deviation difference from $\sigma_{Y_i} = 1.1$ to 2, which is $h \cdot g + g = 1.1{\sim}2$. Sequences of boxplots are generated. From the boxplots, the ratio value of the entropy value at segment *i* vs. that of the template segment at 95^{th} percentile, 75^{th} percentile, and the median are recorded in the table, as demonstrated in Table 3.1. Each table value is the ratio of SampEn values ρ_i based on 1000 simulation runs.

User Guidelines for Table 3.1

The magnitude of deviation is measured by the combinatory effect of standard deviation $\sigma_{Y_i} = h \cdot g + g$, since the calculation result is from a standardized input data. The deviation from target standard deviation can be simply measured by $\sigma_{Y_i} - 1$. To look up the table, we use the first column to be predefined r value, which is predetermined according to step 3 in section 3.3. Then we search for the AdSEn output value v along the specific row. When scanning through the selected row, if the AdSEn output v is in between two values listed in the same percentile value, say 95, then we can conclude we will encounter 95% of the cases that the process parameters have changed. The magnitude of combined deviation can be read from the corresponding value listed on the top row of the table. It is possible that the output ρ occurs in multiple places. For example, we selected r=0.1, then we look at the row of 0.1, and if *v=1.33*, we can find it in 95 percentile that $\sigma_{Y_i} = 1.1$. We can also find *v=1.33* in 75 percentile of $\sigma_{Y_i} = 1.6$. We have higher confidence (95 percentile) that the combined deviation is 1.1sigma than a 1.6 sigma shift (75 percentile). In the case of $\sigma_{Y_i} = 1.1$, the process has at least 0.1 combinatory standard deviation increase, and in 75% of the cases, the process has at least 0.6 combinatory standard deviation increase. In addition, the combined deviation from the template (e.g. the first segment), is a combinational effect due to mean shift only, variance shift only, or a combination of both. It is not possible to identify the exact source(s) of deviation without further data analyses. The details of the simulation experiments will be discussed in chapter 4.4, and an example of using Table 3.1 will be presented in section 3.5.

3.5 AdSEn Application Example

This example is from the extruder machine sensors of local manufacturing company. The dataset was taken from an extrusion machine. The quality characteristic of interest is the machine temperature in a zone when activated.

The dataset is shown in Figure 3.11. AdSEn is performed. The dataset is segmented into 10 subsets, using the first segment as the template, and the threshold value *r* is set to be 0.2 times the standard deviation of the template subset. We have the result and ratio shown in Table 3.2. The first row is the AdSEn output, and the second row is the AdSEn ratio, which is calculated by divide the i^{th} segment's AdSEn output by the 1^{st} AdSEn output. Any ratio smaller than one means there is change in the process' favor, but when the ratio is greater than one, it raises the alarm, such that we use Table 3.1 for interpretation.

Table 3.1 AdSEn Ratio Values ρ as a Function of r, Standard Deviation Change, and Percentile

r	Standard Deviation Change														
	1.1			1.2			1.3			1.4			1.5		
	50	75	95	50	75	95	50	75	95	50	75	95	50	75	95
0.1	1.04	1.14	1.32	1.06	1.18	1.42	1.08	1.22	1.50	1.11	1.26	1.56	1.15	1.30	1.66
0.2	1.04	1.10	1.18	1.09	1.14	1.23	1.12	1.18	1.28	1.15	1.22	1.34	1.18	1.25	1.37
0.3	1.05	1.10	1.17	1.10	1.14	1.22	1.15	1.19	1.27	1.18	1.24	1.32	1.22	1.27	1.36
0.4	1.06	1.11	1.17	1.11	1.16	1.23	1.17	1.21	1.28	1.22	1.26	1.33	1.26	1.31	1.40
0.5	1.07	1.11	1.19	1.13	1.18	1.26	1.20	1.25	1.33	1.25	1.31	1.39	1.31	1.36	1.43
0.6	1.08	1.14	1.22	1.16	1.21	1.29	1.23	1.28	1.37	1.29	1.36	1.43	1.35	1.40	1.49
0.7	1.09	1.15	1.23	1.17	1.23	1.32	1.25	1.32	1.40	1.32	1.38	1.47	1.39	1.45	1.55
0.8	1.10	1.17	1.27	1.19	1.26	1.36	1.28	1.35	1.44	1.37	1.43	1.53	1.45	1.51	1.61
0.9	1.11	1.18	1.31	1.22	1.29	1.39	1.32	1.39	1.50	1.40	1.48	1.59	1.50	1.58	1.69
1	1.13	1.21	1.33	1.24	1.32	1.44	1.35	1.43	1.56	1.45	1.54	1.66	1.56	1.63	1.76

r	Standard Deviation Change														
	1.6			1.7			1.8			1.9			2		
	50	75	95	50	75	95	50	75	95	50	75	95	50	75	95
0.1	1.17	1.33	1.76	1.20	1.41	Inf	1.22	1.44	Inf	1.26	1.46	Inf	1.29	1.54	Inf
0.2	1.21	1.29	1.40	1.24	1.33	1.47	1.27	1.36	1.49	1.31	1.40	1.59	1.32	1.42	1.58
0.3	1.26	1.32	1.40	1.29	1.35	1.45	1.33	1.38	1.49	1.35	1.42	1.54	1.38	1.46	1.58
0.4	1.30	1.35	1.43	1.35	1.40	1.48	1.38	1.44	1.52	1.42	1.48	1.57	1.45	1.52	1.63
0.5	1.35	1.41	1.48	1.40	1.46	1.54	1.45	1.50	1.60	1.48	1.54	1.63	1.53	1.58	1.69
0.6	1.40	1.46	1.54	1.46	1.52	1.60	1.51	1.57	1.66	1.55	1.61	1.71	1.60	1.66	1.75
0.7	1.46	1.52	1.62	1.52	1.58	1.68	1.57	1.64	1.74	1.63	1.69	1.79	1.68	1.74	1.84
0.8	1.51	1.58	1.68	1.58	1.65	1.75	1.66	1.73	1.84	1.71	1.78	1.88	1.77	1.84	1.95
0.9	1.57	1.66	1.77	1.65	1.74	1.85	1.73	1.81	1.92	1.80	1.88	2.01	1.87	1.94	2.06
1	1.64	1.73	1.86	1.73	1.82	1.94	1.81	1.89	2.03	1.88	1.98	2.11	1.96	2.05	2.18

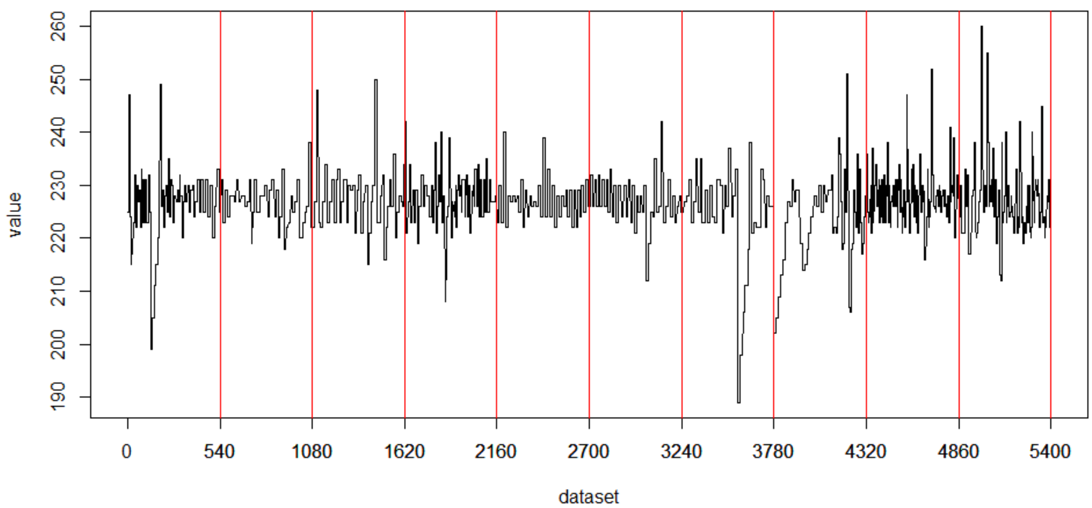

Figure 3-11 A Real-World Temperature Dataset

Table 3.2 AdSEn Output and Ratio Table of each segments, for ratio greater than one, the process if out of control, if less than one, the process if in control

Segments	1	2	3	4	5	6	7	8	9	10
AdSEn Output	0.28	0.13	0.16	0.26	0.13	0.17	0.13	0.16	0.42	0.38
AdSEn Ratio	1.00	0.48	0.58	0.94	0.48	0.62	0.47	0.58	1.51	1.38

We demonstrated the use of the proposed AdSEn in this example. First of all, we implemented the transformation towards the input variables: $\boldsymbol{y_{ij}} = \boldsymbol{x_{ij}}(\left|\frac{\mu_i-\mu}{\sigma}\right| + \mathbf{1})$. Then we run the SampEn algorithm on the transformed input data $\boldsymbol{y_{ij}}$. The outcome is listed in Table 3.2. Row 1 records the AdSEn results of each segment. Row 2 records the ratio of AdSEn output of the i^{th} segment to the that of the 1^{st} segment (aka. template). The larger the ratio is, the bigger difference exists between the i^{th} segment and the template segment.

From this result, we observe that segments 9 and 10 have the ratios larger than 1. Since the SampEn coefficient *r* value is selected at 0.2, we look at the row *r*= 0.2 in Table 3.1. Using the 9^{th} segment as an example, the 95^{th} percentile in 1.8 standard deviation of change has the ratio 1.49. In other words, we can conclude with 95% confidence that the combined change 1.8 in both mean and variance has been contributed to 1.49 times of the template entropy. Simulation studies for the properties of the proposed method will be presented in Chapter 4.

Chapter 4 - Simulation Analysis

SampEn is a popular tool for tackling data noise, and data complexity (Nair 2014). It has a user specified threshold based on the overall standard deviation to capture non-conforming data points. As stated in the previous chapter, this original SampEn technique cannot detect process mean changes, and the proposed adjusted SampEn (AdSEn) is expected to maintain the ability of finding process variance change, and in the meanwhile, has the detectability of detecting mean shifts.

In this chapter, different simulated data patterns will be used to demonstrate the performance of the proposed AdSEn. In section 4.1 and 4.2, the selection of threshold in implementing AdSEn will be discussed. Specifically, experiments are conducted to explore the selection of r and the sensitivity of detecting small changes in either variance or mean. In section 4.3, the mixture simulations where both variance and mean change simultaneously, experiments are conducted to study the selection of scale k and the impact of resolution r in detecting pattern changes. In section 4.4, experiments are performed to discover the numerical solutions, and a series of tables are generated on purpose of drawing a numerical conclusion.

4.1. Case 1: Variance Change Only Pattern

4.1.1Experiment Design and Procedure

The purpose of this experiment is to verify that the AdSEn method maintains the fundamental capability from SampEn of detecting variance changes. This part of simulation is performed on a known set of simulated data, the entire data set is consisted of three portions, 300 data point each, from different distributions, which has the same mean level but different variances. The first 300 data points are from the standard normal distribution, *N(0,1)*, and the following two portions are from *N(0,2)* and *N(0,3)*. The data set is shown in Figure 4.1.

This experiment explores how small a change in variance change only cases that the proposed AdSEn can detect. In this section, we will firstly perform experiment in a larger range of variance, the first 300 data points are from the standard normal distribution, *N(0,1)*, and the following two portions are from *N(0,2)* and *N(0,3)*, and then gradually narrow down to smaller variance deviations. We expect that the proposed method is capable of detecting a variance change in 0.01.

The experiment is replicated 100 times from different random seeds. Boxplots are drawn for each individual threshold value R. In small sample size, SampEn does not give good result on low r values. Studies for lower r value SampEn application on small dataset are still needed. In this section, discussions are mainly focused on larger *r* values. For practitioners, Figure 4.2 provides a guideline for selection of resolution *r* when a clear-cut standard is available. In lower values of *r* in Figure 4.2, we can see that the boxplot looks abnormal. The data output shown in Figure 4.3 shows that there are entropy outcomes such as not available (NaN) and infinite result (Inf). The cause of NaN and Inf is that denominator in Equation 2.12 $SampEn(m,r) = -log(\frac{C^{m+1}(r)}{C^{m}(r)})$, the denominator is 0, or approximating 0. To ensure better results, experiments are only performed on higher r values in this research to avoid the NaN and Inf output cases.

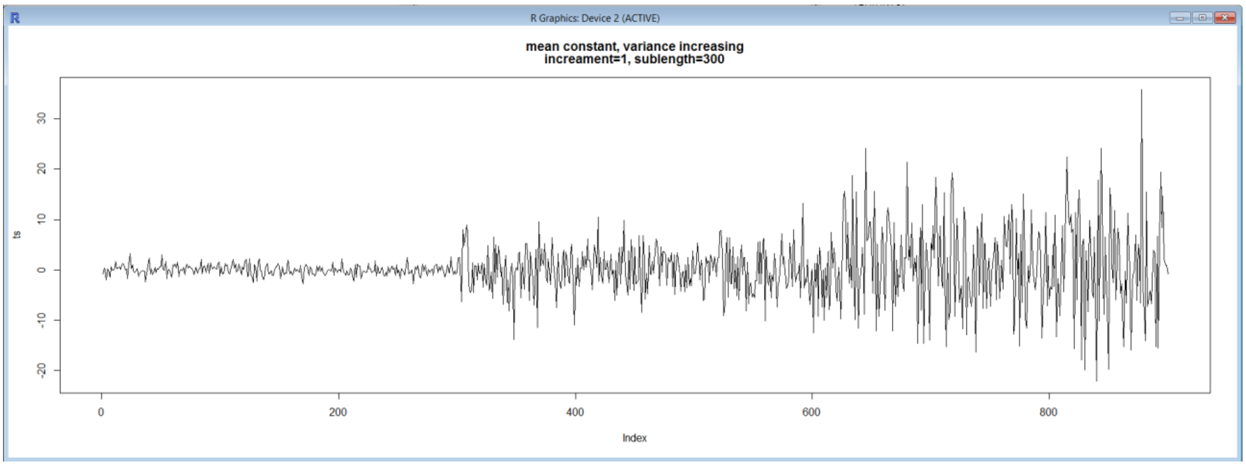

Figure 4-1 Variance change only data set

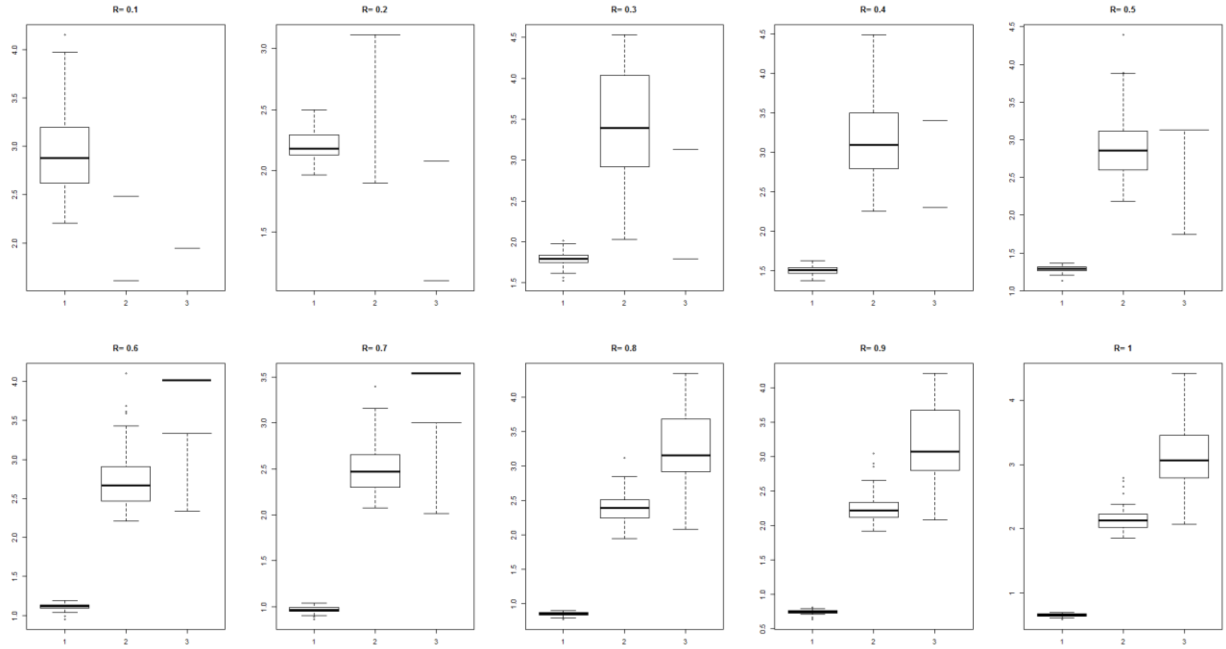

Figure 4-2 Multi-R plot for Variance change from 1to 3

```
> multlist[[1]]
  [1] 2.931194      Inf      NaN 2.959365      Inf      NaN 3.288402      Inf
  [9]      Inf 2.323787      Inf      NaN 3.417727      Inf      NaN 3.576550
 [17]      Inf      NaN 3.083438      Inf      Inf 3.081910 2.639057      Inf
 [25] 3.098590      NaN      Inf 2.631889      Inf      Inf 3.776585      Inf
 [33]      NaN 3.194583      Inf      Inf 2.624669      Inf      Inf 2.786382
 [41]      Inf      NaN 2.883403      Inf      Inf 3.020425      Inf      Inf
 [49] 2.847812      Inf      Inf 3.161247      Inf      Inf 2.876386      Inf
 [57]      NaN 3.332205      Inf      Inf 2.921624      Inf      Inf 2.913902
 [65]      Inf      NaN 2.536579      Inf      Inf 2.436116 2.639057      Inf
 [73] 3.152736      Inf      NaN 3.564827      Inf      NaN 2.740840      Inf
 [81]      Inf 3.098590 2.564949      NaN 3.518980      Inf      NaN 3.784190
 [89]      Inf      Inf 3.392829      Inf      NaN 3.097515      Inf      Inf
 [97] 2.592537      Inf      NaN 3.576550      Inf      Inf 2.583998      Inf
```

Figure 4-3 Example SampEn output of Inf/NaN cases

With selected threshold parameter *r*, a study to identify minimum detectability is performed. Since the entropy output shows large abnormality in *r*= 0.1~0.4, experiments will be performed on *r*=0.6~1.0 to provide more meaningful results. Datasets for experiment will be simulated from normal distribution. As shown in Figure 4.4, *N(0, 1)* to *N(0, 2)*, 6 subsets with 300 data points, variance increase in the increment of *0.2σ* for each segment. Trials and errors were performed to ensure the lower bound of AdSEn can be detected. In

Figure 4.4, we observe that when boxplots do not overlap with each other, the conclusion can be made that there is a change in process parameters.

A simulation of smaller increment size in variation is performed as shown in Figure 4.5, *r*=0.5~1, dataset variance increases 0.01 from *N(0, 1)* to *N(0, 1.06)*. From the figures, we observe that when *r* is large, the neighboring boxplot of the template is slightly overlapping, but also have a significant difference in majority of the distribution, since 25 percentile to 75 percentile region (called the interquartile) is not overlapping at all. These figures demonstrate that as small as 0.01 variance change (or 0.1 standard deviation change in the variance change only cases) is detectable. This result agrees with the conclusions drawn in section 3.4 and 3.5.

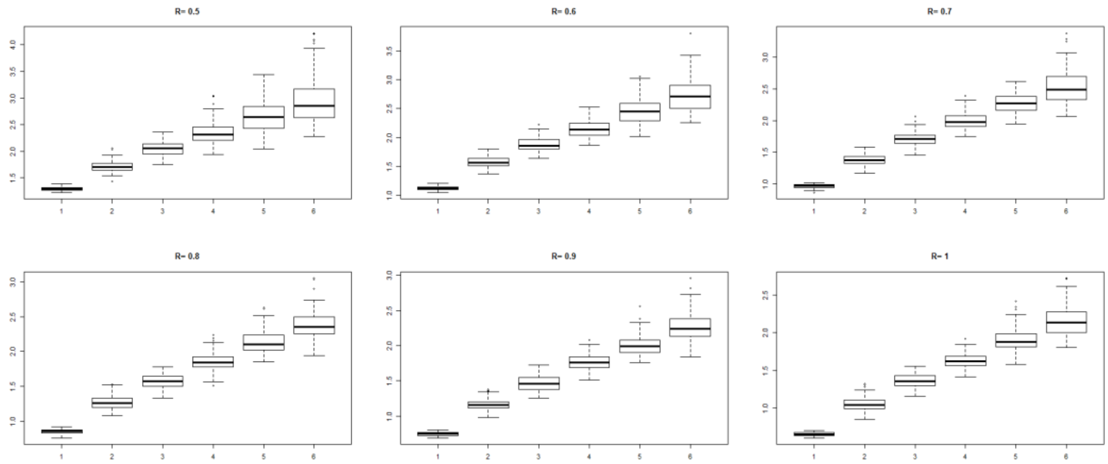

Figure 4-4 Boxplot of *r*=0.5~1.0, dataset variance increment 0.2 from N(0,1)

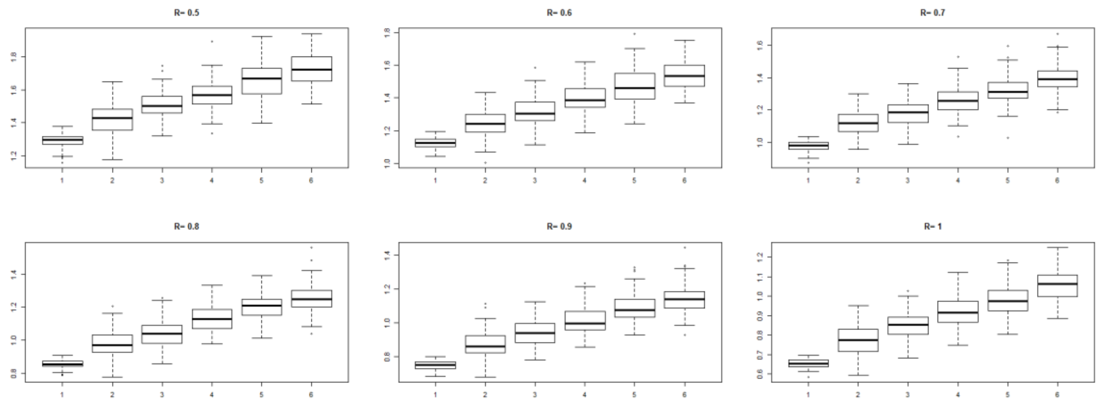

Figure 4-5 Boxplot of *r*=0.5~1, dataset variance increment 0.01 from N(0,1)

4.1.2 Result and Discussion

As shown by the higher r values in Figure 4.5, we can conclude that AdSEn maintains the ability to detect the variance change which SampEn also is capable of. We can also see that threshold selection is important when implementing AdSEn. When r is too small, the function will not have a quantitative conclusion. Though results can be interpreted qualitatively, quantitative details will be more conclusive and persuasive. Pushing to its limit, AdSEn can detect a variance change as small as 0.1σ.

1.3 Proposed Method

To enable SampEn detecting mean shift, we propose a method called Adjusted Sample Entropy (AdSEn) which is based on Grassberger's (1983, 1988) SampEn algorithm. However, since the original SampEn itself does not detect mean shifts, we propose a transformation on an input time series $\boldsymbol{x}$. The proposed method transforms original $\boldsymbol{x}$ to a new time series $\boldsymbol{y}$, such that the variance change and mean shift will both be reflected via the transformation. Then we perform Grassberger's SampEn algorithm on the new standardized time series $\boldsymbol{Y}$. The proposed AdSEn should be able to numerically identify the stability of a process. Brassberger's SampEn algorithm should also be applied to the original input time series $\boldsymbol{x}$ as well. The combinational use of both entropy outputs should guide users for proper decision-making regarding to process capability studies.

4.2. Case 2: Mean Level Shift Only

4.2.1Experiment Design and Procedure

The purpose of this experiment is to test whether AdSEn method has obtained the capability of detecting mean shifts. This part of simulation is performed on a known set of simulated data, the entire data set is consisted of three portions, 300 data point each, from different distributions, which has the same variance but different mean levels. The first 300 data points are from *N(0, 1)*, and the following two portions are from *N(1, 1)* and *N(2, 1)*.

The data set is shown in Figure 4.6. Similar to section 4.1, via the same approach we also performed lowest detectability experiment for mean level shift only cases.

4.2.2 Result and Conclusion

As shown by higher *r* values in Figure 4.7, we can conclude that AdSEn is capable of detecting mean-shift-only data patterns. We also observe that threshold selection is important when AdSEn is implemented. When *r* is too small, the proposed AdSEn does not provide a quantitative conclusion.

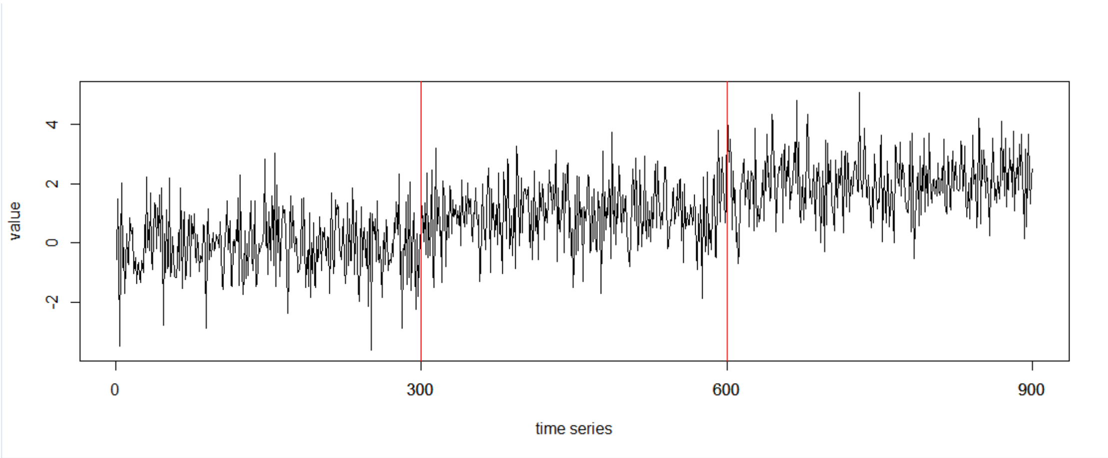

Figure 4-6 Mean shift only data set

This experiment is also replicated 100 times from different random seeds. Boxplots are drawn for each individual threshold value *r* for threshold selection when the specification limits are not given. There are also abnormal boxplot results due to NaN and Inf entropy output when small *r* values are used.

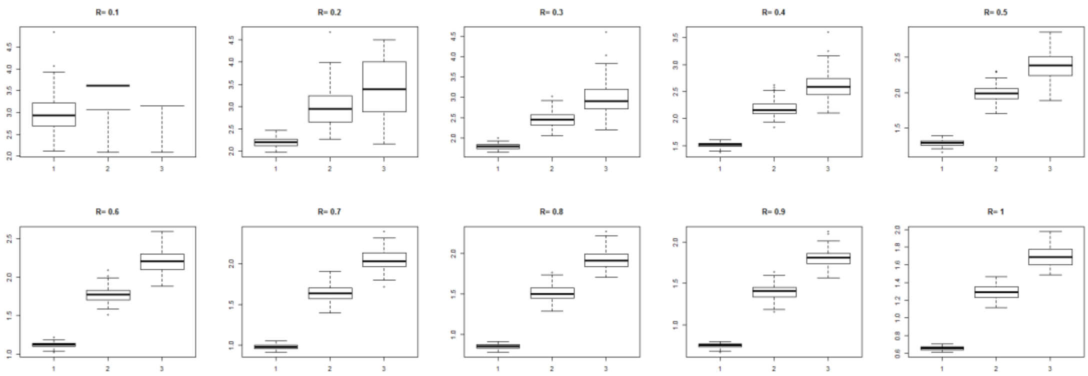

Figure 4-7 Multi-R plot for Mean Shift from 0 to 2

In a sensitivity study, we set out to show that the proposed AdSEn is capable of detecting a process mean shift as small as 0.1 σ. One hundred simulations are performed for *r* from 0.2~1.0, and the results are shown in the boxplots in Figure 4.8. The first segment contains the data from the simulated dataset N(0, 1). The rest of the data segments have an incremental increase of 0.1 mean shift up to N(0.5, 1). Six boxplots represent the distributions of ratios ρ between the i^{th} segment and the first segment. From the boxplots, we observe that the large (> 0.2) mean shifts can be easily detected because the boxplots do not overlap with that of the template (i.e. the first boxplot). The trend demonstrated by the boxplots show that the proposed AdSEn can be used to identify mean shifts larger than 0.2. Since the process standard deviation is 1 in this case, sigma is 1.

When the mean shift is as small as 0.1 sigma shift, the identification through the use of AdSEn will depend on the choice of *r*. For example, in the case of *r*=0.9, interquartile range of the *2nd* boxplot is the largest. However, the second boxplot and the first boxplot (for the template) do not overlap. When r=0.2 is chosen, on the other hand, the first and the second boxplots overlap, meaning that it is likely that 0.1 sigma mean shift may not be detected in some simulation runs. In general, we recommend the choice of a large r value (e.g. *r* >0.4) to allow the proposed AdSEn to perform better in identifying a potential process mean shift, large or small.

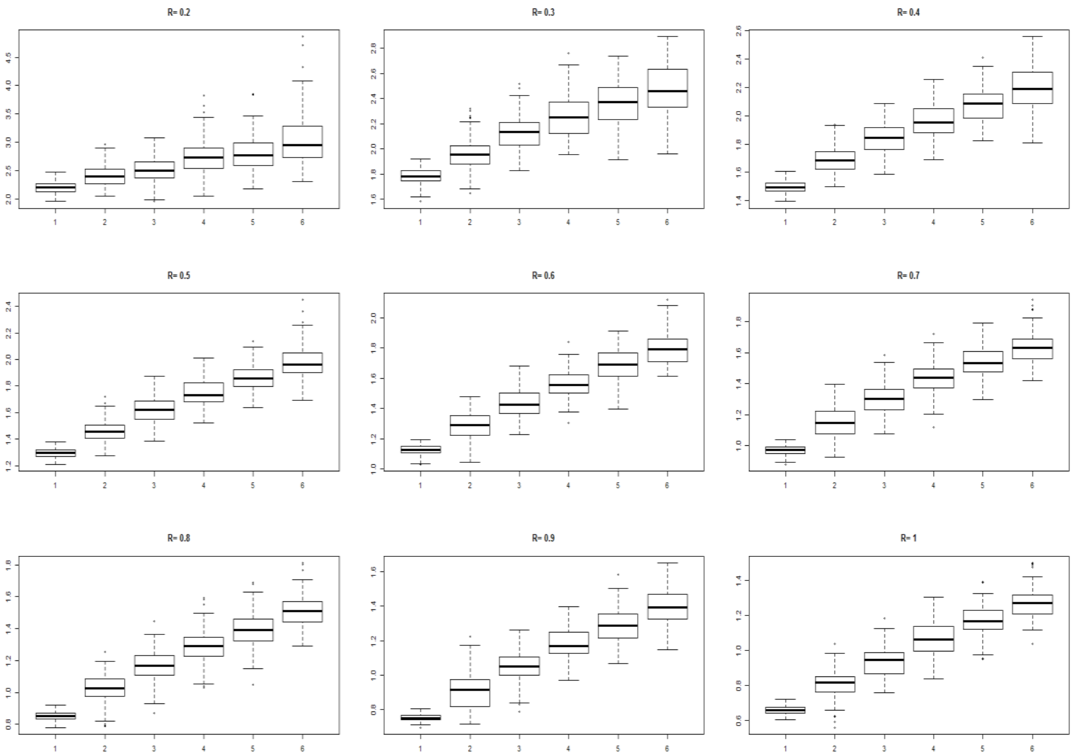

Figure 4-8 Multi-Scale Boxplot for multiple r=0.2~1.0

4.3 Both Mean and Variance Deviated Data

4.3.1 Experiment Design and Procedure

The purpose of this experiment is to test the performance of multi-scale method, in detecting pattern shift. This part of simulation is performed on a known set of simulated data, the entire dataset $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{x_3}, \boldsymbol{x_4})$ is consisted of four portions, 1200 data point each, from different distributions. The template dataset x_1 is the first section, all x_{1j}'s are from *N(0, 1)*, x_{2j} is from *N(1, 1)*, x_{3j} is from *N(0, 2)*, and x_{4j} is from *N(1, 2)*. The data set is shown in figure 4.9. Similar to sections 4.1 and 4.2, the experiments were performed for mean shift and variance change combined cases via the same approach.

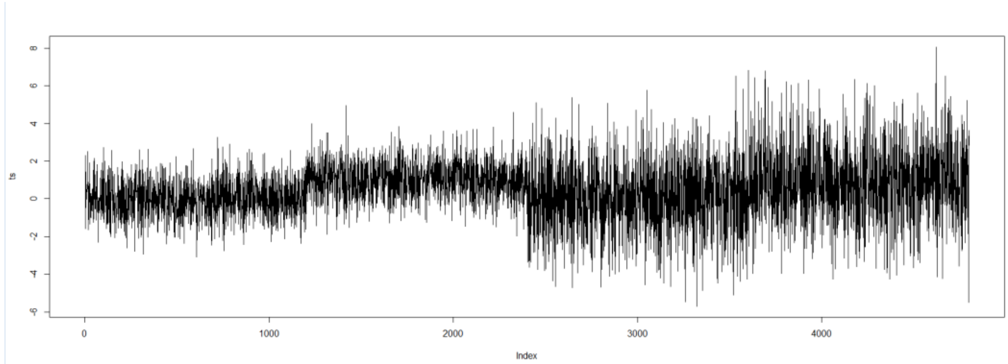

Figure 4-9 Dataset $x = (x_1, x_2, x_3, x_4)$

The experiment is also replicated 100 times from different random seeds. Boxplots are drawn for a controlled *r* value 0.6, for the purpose of high upper limit, which can tolerate 2 sigma mean shift and 2*x* variance change. The detectability of mean shift and variance has been verified from 4.1 and 4.2, and the sensitivity analysis is also concluded that when r=0.6, AdSEn can detect a 0.2 sigma mean shift and 0.2*x* standard deviation change, thus detectability and sensitivity experiment will not be repeated in this section. The purpose of this section is to discover the pattern detection of adjusted SampEn multi-scale method.

4.3.2 Result and Conclusion

This experiment results are shown in figure 4.10~4.13. From all the different scales, we can see that scale 4, 8, 12, and 16 are revealing the exact same result, other than that, the rest of all the boxplots does not appear to be in common with any of the others. We can conclude that in this case, the dataset pattern occurs at 4 equal length points.

It is important to mention two drawbacks of this method. First, this result is not a general case. Though the proposed multi-scale approach seems to work perfectly in this scenario, it might not perform as well in data pattern shift occurring in segments with unequal length. Adjusting/increasing scale number can help making a better approximation. When no common patterns found in the boxplots, we can either increase the resolution by increasing scale, or apply change point algorithm to define the pattern. However, the effort to decide change point will seem costly when we see that AdSEn returns a good detection based on a sufficient approximation.

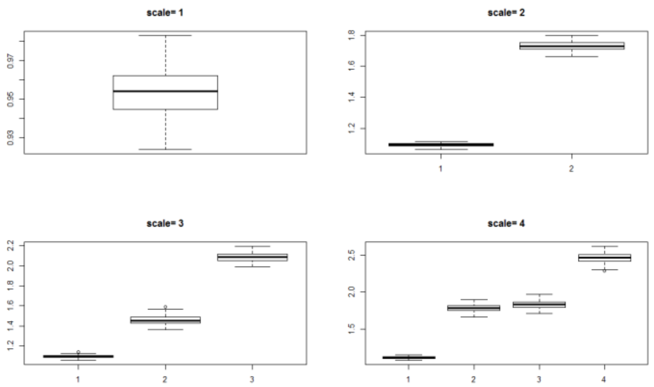

Figure 4-10: Scale 1 to 4

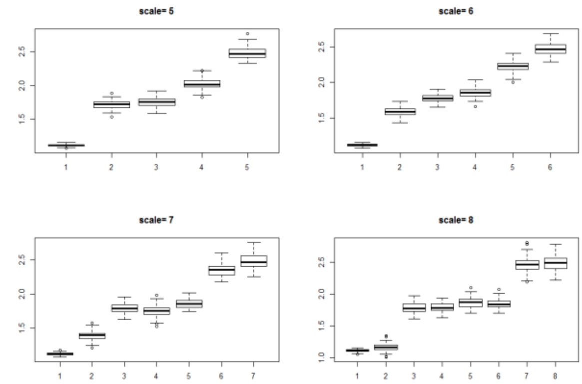

Figure 4-11: Scale 5 to 8

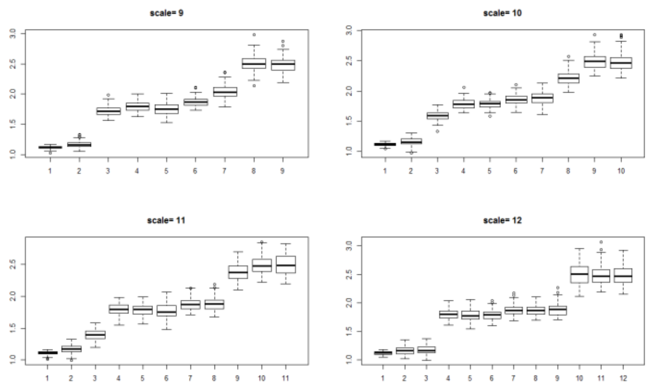

Figure 4-12: Scale 9 to 12

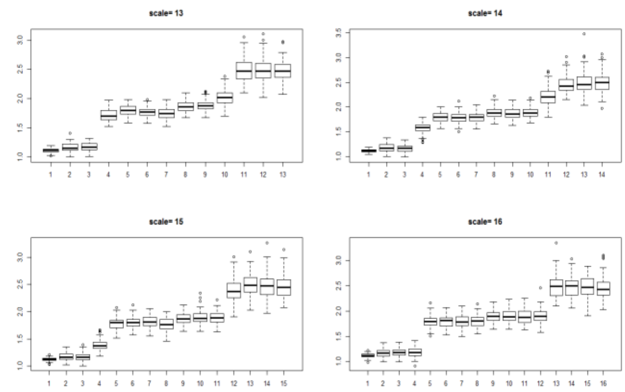

Figure 4-13: Scale 13 to 16

Secondly, this approach might face situations when mean and variance changes may compensate each other, for example, when variance becomes smaller but the mean level shifts in a time series segment. Using Equation 3.1, $\boldsymbol{y_{ij}} = \boldsymbol{x_{ij}}(\left|\frac{\boldsymbol{\mu_i - \mu}}{\boldsymbol{\sigma}}\right| + \mathbf{1})$, we observe that the new incoming data has a decrease of variance in original input variable $\boldsymbol{x_{ij}}$, however a large mean level shift would cause $\left|\frac{\boldsymbol{\mu_i - \mu}}{\boldsymbol{\sigma}}\right|$ to increase. The net result is that the decrease of variance would compensate for the increase in $\left|\frac{\boldsymbol{\mu_i - \mu}}{\boldsymbol{\sigma}}\right|$.

In the cases of both mean shifts and variance changes, we should examine the entropy values of both original time series $\boldsymbol{x_{ij}}$and transformed data $\boldsymbol{y_{ij}}$. If the entropy value generated from $\boldsymbol{x_{ij}}$ decreases but that of $\boldsymbol{y_{ij}}$ stays the same, this may be an indication that the scenario described in the previous paragraph may take place.

4.4 Generate AdSEn Ratio Values

4.4.1Experiment Design and Procedure

To generate AdSEn Ratio Values, experiment were run 1000 replicates for different size of changes. From section 3.4, we know that $\boldsymbol{\sigma_{Y_i} = h \cdot g + g}$. In the conducted experiments, $\sigma_{Y_i}/\sigma_{Y_0}$, since we used standardized data, $\frac{\sigma_{Y_i}}{\sigma_{Y_0}} = \sigma_{Y_i}$, and σ_{y_i} was controlled within 1.1~2..

The selection of proper data length for this experiment is critical. Even though SampEn claims less dependent and more consistent than ApEn (Richman, 2000), the recommended length of ApEn is 10^{m}~20^{m}, as explained in section 2.2, m is the size of vector (Pincus, 1994) to ensure the robustness of calculation result, and (Ahmed, 2012) suggests that SampEn use N≥300. However from the experiment result, 300 failed to guarantee effective output of SampEn, the recommended data length of ApEn is adopted. Since m=2 is used throughout the all the experiments, N=20^{m}=400 was employed in this one as well.

Table 4.1 The Percentage of Replicates Table of Ratio Greater than 1 (Note: If ratio is greater than one, then there is change)

r	$\sigma_{Y_{ij}}$ changes									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2**
0.1	60.80%	66.50%	69.00%	74.30%	78.10%	81.90%	85.10%	83.80%	85.10%	86.60%
0.2	73.20%	86.50%	94.40%	96.70%	99.90%	99.00%	99.50%	99.80%	99.50%	99.90%
0.3	80.60%	94.40%	98.80%	99.70%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.4	83.80%	96.10%	99.50%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.5	84.80%	96.90%	99.60%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.6	87.60%	98.10%	99.60%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.7	83.50%	98.30%	99.80%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.8	87.90%	98.00%	99.80%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
0.9	88.60%	98.30%	99.90%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
1	87.40%	97.90%	99.80%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

4.4.2 Result and Conclusion

From the 1000 times of simulation results, we have generated the ratios of entropy from a segment of interest to that from a standard segment. Table 4.1 records the percentage that the replicates. At the certain shift level, the percentage of the cases that ratio value ρ_i greater than 1 are recorded.

In Tables 4.1 to 4.4, ρ values of desired percentiles are calculated. Table 4.2 records the value of ρ at 95th percentile. If the number is larger than the number in the table, that means we have at least 95% confidence that the process has been deviated the corresponding value in σ_{y_0}. The ρ at 75th percentile and median level are recorded in Tables 4.3 and 4.4.

Table 4.2 AdSEn Ratio Values ρ as a Function of r, Standard Deviation Change, at 95th Percentile

95%										
r	**$\sigma_{Y_{ij}}$ Change**									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2**
0.1	1.32	1.42	1.50	1.56	1.66	1.76	Inf	Inf	Inf	Inf
0.2	1.18	1.23	1.28	1.34	1.37	1.40	1.47	1.49	1.59	1.58
0.3	1.17	1.22	1.27	1.32	1.36	1.40	1.45	1.49	1.54	1.58
0.4	1.17	1.23	1.28	1.33	1.40	1.43	1.48	1.52	1.57	1.63
0.5	1.19	1.26	1.33	1.39	1.43	1.48	1.54	1.60	1.63	1.69
0.6	1.22	1.29	1.37	1.43	1.49	1.54	1.60	1.66	1.71	1.75
0.7	1.23	1.32	1.40	1.47	1.55	1.62	1.68	1.74	1.79	1.84
0.8	1.27	1.36	1.44	1.53	1.61	1.68	1.75	1.84	1.88	1.95
0.9	1.31	1.39	1.50	1.59	1.69	1.77	1.85	1.92	2.01	2.06
1	1.33	1.44	1.56	1.66	1.76	1.86	1.94	2.03	2.11	2.18

We can also notice that there are some cells in the tables when r=0.1, the cells are showing "Inf", and some of the result are very inconsistent. As discussed in sections 4.1 and 4.2, we do not recommend the use of small r values. It is difficult for SampEn algorithm to ensure good performance in small sample size when small r values are used.

Table 4.3 AdSEn Ratio Values ρ as a Function of r, Standard Deviation Change, at 75th Percentile

75%										
r	$\sigma_{Y_{ij}}$ **Change**									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2**
0.1	1.14	1.18	1.22	1.26	1.30	1.33	1.41	1.44	1.46	1.54
0.2	1.10	1.14	1.18	1.22	1.25	1.29	1.33	1.36	1.40	1.42
0.3	1.10	1.14	1.19	1.24	1.27	1.32	1.35	1.38	1.42	1.46
0.4	1.11	1.16	1.21	1.26	1.31	1.35	1.40	1.44	1.48	1.52
0.5	1.11	1.18	1.25	1.31	1.36	1.41	1.46	1.50	1.54	1.58
0.6	1.14	1.21	1.28	1.36	1.40	1.46	1.52	1.57	1.61	1.66
0.7	1.15	1.23	1.32	1.38	1.45	1.52	1.58	1.64	1.69	1.74
0.8	1.17	1.26	1.35	1.43	1.51	1.58	1.65	1.73	1.78	1.84
0.9	1.18	1.29	1.39	1.48	1.58	1.66	1.74	1.81	1.88	1.94
1	1.21	1.32	1.43	1.54	1.63	1.73	1.82	1.89	1.98	2.05

Table 4.4 AdSEn Ratio Values ρ as a Function of r, Standard Deviation Change, at 50th Percentile

50%										
R	$\sigma_{Y_{ij}}$ **Change**									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**1.7**	**1.8**	**1.9**	**2**
0.1	1.04	1.06	1.08	1.11	1.15	1.17	1.20	1.22	1.26	1.29
0.2	1.04	1.09	1.12	1.15	1.18	1.21	1.24	1.27	1.31	1.32
0.3	1.05	1.10	1.15	1.18	1.22	1.26	1.29	1.33	1.35	1.38
0.4	1.06	1.11	1.17	1.22	1.26	1.30	1.35	1.38	1.42	1.45
0.5	1.07	1.13	1.20	1.25	1.31	1.35	1.40	1.45	1.48	1.53
0.6	1.08	1.16	1.23	1.29	1.35	1.40	1.46	1.51	1.55	1.60
0.7	1.09	1.17	1.25	1.32	1.39	1.46	1.52	1.57	1.63	1.68
0.8	1.10	1.19	1.28	1.37	1.45	1.51	1.58	1.66	1.71	1.77
0.9	1.11	1.22	1.32	1.40	1.50	1.57	1.65	1.73	1.80	1.87
1	1.13	1.24	1.35	1.45	1.56	1.64	1.73	1.81	1.88	1.96

Chapter 5 - Discussion and Future Work

From the simulations and the real case example, we find that AdSEn is a promising tool based on SampEn. The proposed transformation $y_{ij} = x_{ij}(\left|\frac{\bar{x}_i-\mu}{\sigma}\right| + 1)$ enables SampEn algorithm detecting mean shift. A table-look-up technique is developed as the numerical solution for AdSEn, the example in section 3.5 proves that this technique is feasible. In chapter 4, the replicated simulation studies illustrated that the proposed tool can identify process changes before a proper process capability studies are performed.

However, our research is limited in time and computing resources, we still have some research questions to be answered in future studies. Although AdSEn performs well in our example in section 3.5, it may not be able to handle all cases. For example, when very small standard deviation change occurs. From Equation 3.1 $\boldsymbol{\sigma_{Y_i}} = \boldsymbol{h} \cdot \boldsymbol{g} + \boldsymbol{g}$, we found that in some cases, mean level change $\boldsymbol{h}$ and standard deviation change $\boldsymbol{g}$ will compensate each other. Though the impact of standard deviation can be transformed perfectly towards $\boldsymbol{\sigma_{Y_i}}$, $\boldsymbol{h} \cdot \boldsymbol{g}$ causes co-effect in the output, mean level change $\boldsymbol{h}$ becomes meaningful. If a small standard deviation change g occurs, the effect of $\boldsymbol{h}$ will be under expressed.

Since a high $\boldsymbol{h}$ might be compensated by a low $\boldsymbol{g}$, where returns small change, no change, or even a change towards $\boldsymbol{\sigma_{y_i}} < 1$. We suggest a variance test prior to AdSEn process to distinguish the cases that whether $\boldsymbol{\sigma_{x_i}} > \boldsymbol{\sigma_0}$ or $\boldsymbol{\sigma_{x_i}} > \boldsymbol{\sigma_0}$. If $\boldsymbol{\sigma_{x_i}} > \boldsymbol{\sigma_0}$ or $\boldsymbol{\sigma_{x_i}} = \boldsymbol{\sigma_0}$, AdSEn algorithm will perform perfectly. Otherwise, we need to develop another method to tackle this situation.

$$\boldsymbol{\sigma_{y_i}} = \begin{cases} \boldsymbol{h} + \boldsymbol{g} & \boldsymbol{\sigma_{x_i}} \geq \boldsymbol{\sigma_0} \\ some\ other\ method & \boldsymbol{\sigma_{x_i}} < \boldsymbol{\sigma_0} \end{cases}$$

In this research, simulations are all performed under normal distributions for demonstration purposes. The goal of this research is to enable SampEn to be a capable tool for detecting changes in a time series, thus simulation of other type of non-normal distributions are needed to further verify the proposed AdSEn. We expect that the proposed AdSEn is capable of handling all kinds of data distributions.

This method can also be extended into multivariate cases. It is commonly recognized that process capability analysis should be able to handle multiple variables (Tano, 2012). In most manufacturing applications, there are more than one QCs to be considered, and some may also interact with each other. In traditional PCA studies, Tano (2012) have proposed multivariate PCRs. This is also an important topic to extend this research.

Xie (2010) pointed out that SampEn(m, r, N) is not defined if no template and forward match occurs in the case of small r and dataset length N. Moreover, the value of SampEn is discontinuous and may vary significantly with a slight change of the tolerance r. Xie (2010) claims that a modified SampEn (mSampEn) can overcome these difficulties. From our experiment results, Xie's opinion is very valuable. Kong (2011) has implemented mSampEn in measurement to classify ventricular tachycardia and fibrillation. However, not very many applications are found in manufacturing applications. It is worth topic for future studies towards process capability.

References

Aboy, M., Cuesta-Frau, D., Austin, D., & Mic ́o-Tormos, P. (2007). Characterization of Sample Entropy in the Context of Biomedical. *Conference of the IEEE EMBS Proceedings of the 29th Annual International*, (pp. 5943-5946).

Ahmed, M. U., & Mandic, D. P. (2012). Multivariate Multiscale Entropy Analysis. *IEEE SIGNAL PROCESSING LETTERS, 19*(2), 91-94.

Borchers, H. W. (2014, 10 12). Package ‘pracma’. (1.7.3), 21-23.

Bothe, D. (1997). *Measuring Process Capability: Techniques and Calculation ofr Quality and Manufacturing Engineers.* McGraw-Hill.

Boyles. (1996). Exploratory Capability Analysis. *Journal of Quality Technology, 28*, 91-98.

Costa, M., Goldberger, A. L., & Peng, C. K. (2002). Multiscale Entropy Analysis of Complex Physiologic Time Series. *PHYSICAL REVIEW LETTERS, 89*(6), 068102, page number 4.

Du, L., Song, Q., & Jia, X. (2014). Detecting concept drift: An information entropy based method using an adaptive sliding window. *Intelligent Data Analysis, 18*(3), 337-364.

Grassberger, P. (1988). Finite sample corrections to entropy and dimension. *Physics Letters A, 128*(6-7), 369-373.

Grassberger, P., & Procaccia, I. (1983). Estimation of the Kolmogorov entropy from a chaotic signal. *Physical Review A, 28*(4), 2591--2593.

Juran, J. M. (1974). *Juran's Quality Control Handbook* (3 ed.). New York, N.Y: McGraw-Hill.

Kane, V. E. (1986). Process Capability Indices. *Journal of Quality Technology, 18*, 41-52.

Keller, A. P., & Pyzdek, T. (2003). *Quality Engineering Handbook (Quality and Reliability).* CRC Press.

Kolmogorov, A. N. (1998). On tables of random numbers. *Theoretical Computer Science*, 387-395.

Kong, D.-R., & Xie, H.-B. (2011). Use of modified sample entropy measurement to classify ventricular tachycardia and fibrillation. *Measurement, 44*, 653-662.

Kotz, S., & Johnson, N. J. (2002). Process Capability Indices - A Review, 1992-2000. *Journal of Quality Technology*, *34*(1), 2-53.

Lake, D. E., Richman, J. S., Griffin, M., & Moorman, J. (2002). Sample entropy analysis of neonatal heart rate variability. *Am J Physiol Regul Integr Comp Physiol, 283*, 789-797.

Liu, H., & Han, M. (2014). A fault diagnosis method based on local mean decomposition. *Mechanism and Machine Theory*, 67-78.

Martı´nez-Olvera, C. (2012). An entropy-based approach for assessing a product's BOM blocking. *International Journal of Production Research, 50*(4), 1155-1170.

Montgomery, D. (2009). *Introduction to Statistical Quality control* (6 ed.). John Wiley & Sons, Inc.

Myers, R. H., Montgomery, D. C., & Anderson-Cook, C. M. (2009). *Response Surface Methodology* (3 ed.). John Wiley & Sons, Inc.

Nair, S. S., & Joseph, K. P. (2014). Chaotic Analysis of the Electroretinographic. *BioMed Research International, 2014*(Article ID 503920), 8 pages. Retrieved from BioMed Research International: doi:10.1155/2014/503920

Oprean, C., & Bucur, A. (2013). Modeling and simulation of the quality's entropy. *Quality & Quantity, 47*(6), 3403-3409.

Pearn, W., Wu, C., & Wu, C. (2014). Estimating Process Capacity Index Cpk: Classical Approach versus Baysian Approach. *Journal of Statistical Computation and Simulation*. doi:10.1080/00949655.2014.914211

Pincus, S. M. (1991). Approximate entropy as a measure of system complexity. *Mathematics, 88*, 2297-2031.

Pincus, S., & Goldberger, A. (1994). Physiological time-series analysis: what does regularity quantify? *Heart and Circulatory Physiology, 266*(4), 1643-1656.

Richman, J. S., & Moorman, J. (2000). Physiological time-series analysis. *Am J Physiol Heart Circ Physiol*, *278*, 2039–2049.

Shannon, C. E. (1948). A Mathematical Theory of Communication*. *Mobile Computing and Communications Review, 5*(1), 3-55.

Tano, I., & Vännman, K. (2013). A Multivariate Process Capability Index Based on the First Principal Component Only. *QualityReliability Engineering International, 29*(7), 987-1003.

Wei, W., Pearn, S., & Kotz, S. (2009). An overview of theory and practice on process capability indices for quality assurance. *International Journal of Production Economics*, *117*, 338-359.

Wu, S., Wu, P.-H., Wu, C.-W., Ding, J.-J., & Wang, C.-C. (2012). Bearing Fault Diagnosis Based on Multiscale Permutation Entropy and Support Vector Machine. *Entropy, 14*, 1343-1356.

Xie, H.-B., Guo, J.-Y., & Zheng, Y.-P. (2010). Using the modified sample entropy to detect determinism. *Using the modified sample entropy to detect determinism, 374*(38), 3926-3931.

Yum, B.-J., & Kim, K.-W. (2009). A bibliography of the literature on process capability indices: 2000–2009. *Quality and Reliability Engineering International, 27(3),* 251-268.