Phase 1 Analysis for Manufacturing Process Control

PROJECT REPORT

ISEN 614 – ADVANCED QUALITY CONTROL

SPRING 2019

Adaiyibo E. Kio (318005235)

Shang-Chin Lee (128003573)

Anthony Mulenga (916005319)

Smith K. Shah (227008337)

EXECUTIVE SUMMARY

The project is based on a set of data collected from a manufacturing process, in which both in-control and out-of-control data are present. The aim of the project was to develop a method or a procedure to identify the data that are in-control and those that are out-of-control so that the estimating the in-control distribution parameters can be computed and a monitoring scheme can be set up to detect out-of-control data in the future. This is essentially a Multivariate Anomaly Detection Phase I analysis.

The data set contains 552 data records with 209 variables. The large number of variables in the dataset necessitated the use of a dimension reduction method, so that the multivariate anomaly detection methods can be effectively applied. Principal Component Analysis (PCA) was used as the dimension reduction method in this work. 4 PCs were found to account for 80% of the variability in the data set and these 4 PCs were used for anomaly detection.

Multivariate control charts – Hotelling T^2 and m-CUSUM – were utilized for the 4 PCs to determine the out-of-control data points, and remove them in order to achieve an in-control system. This was an iterative process that involved alternating the implementation of the Hotelling T^2 and m-CUSUM control charts.

Firstly, the Hotelling T^2 control chart was used to remove out-of-control data points, especially those of a spiky nature. The m-CUSUM control chart was then used to remove out-of-control data points of any other sort that might still be present in the data set. This process was repeated until there were no out-of-control data points in the data set regardless of which control chart was being used.

The parameters of this final in-control data set can now be used for any future process monitoring and anomaly detection purposes. In addition, we also survey phase 1 analysis methods that is not included in lecture materials to explore more state-of-art techniques in this area. The results are presented in Appendix B. The study utilized some open source software packages on R and Python.

Introduction

Problem Statement

This project calls for analysis on a set of data collected from a manufacturing process. The data set contains both in-control and out-of-control data. The objective of the project is to develop a method or procedure in which we can identify both the in and out-of-control data. These data points can be identified by completing a phase I analysis, whose end result isolates in-control data, and can be used for monitoring of future data. The data that we are given has the following characteristics, per the project guidelines (Zou, 2019):

(a) This is a manufacturing related dataset, and it has a total of 552 data records.

(b) Each row is a data record. Each data record contains 209 values.

(c) This is a multivariate detection dataset in which p = 209. The sample size is n = 1.

(d) The physical meaning of each value is omitted.

From this data, we are to identify the in-control data and out of control data, ending up with a control chart that can be used for future data. Before we can begin with our analysis, however, we need to consider the data we are given, and the decisions we need to make regarding it in order to successfully complete the analysis.

Principal Component Analysis (PCA) and Data Reduction Plots

A factor to consider in our process is the sheer amount of data that we have. Although we can account for a multiple variables using a multivariate analysis scheme, it can still be quite cumbersome and inefficient with a large number of variables. A further extension of the curse of dimensionality, even when using multivariate charts, a high variable count can lead to an aggregation of small noise effects that can eventually overwhelm the signal and make it hard to reject the null hypothesis. Therefore, it is important to conduct the analysis with a vital few data points, versus a trivial many that may cloud up our results. In order to do this, we make use of the process of the Principal Component Analysis. The principal component analysis allows us to identify these vital few variables—the principal components—that can explain the overall data set well because they can explain a large percentage of the variance that is present in the data. After doing the principal component analysis, we will have a way of telling what percentage of the variance is explained, but we still need to figure out how many of these principal components to use. To aid us in this decision, we will also use Pareto and Scree Plots that allow us to see graphically the relationship between principal components and variances, helping as to make a decision on how many principal components to use.

Multivariate Data Set

As the project statement says, this data set contains multivariate data, with a total of 209 variables. It is entirely possible to use multiple univariate charts to analyze the data, but that would mean at least 209 individual charts such as X-bar, Moving Range, etc. Therefore, it is more efficient for us to instead multivariate charts such T^2, m-CUSMUM, and m-EWMA in order to complete the analysis. Multivariate charts allow us to avoid the pitfalls that come with using multiple univariate charts, especially for large sets of data, such as is in our case. These include having inflated values for α and β error (leading to a lot of missed detections and false alarms) and failing to capture the change in correlation between variables. Using a multivariate analysis scheme essentially allows us to avoid some the pitfalls of the curse of dimensionality.

METHODOLOGY

After having considered the data and project constraints that we were given, we are now ready to begin the analysis. Our analysis is a phase 1 analysis. As mentioned, a phase 1 analysis focuses on finding the in-control training data, which at the end of we will the distribution and control parameters that can be applied to future data. The phase I analysis is an iterative procedure by which we train the data on a control chart. In our case, we will use multivariate control charts in an iterative manner until we get to all in-control data. First, we will begin with data reduction through Principal Component Analysis (PCA), and Scree Plots and Pareto Plots. Then, we shift to identifying the in and out-of-control data points are using two multivariate control charts, T^2 and m-CUSUM. We iterate the use of our control charts until all data points are in control under both conditions.

Data Reduction Pre-Analysis

Before we formally began our PCA, we decided to do some analysis on our data that showed. This analysis shows that we have a unimodal function, meaning that it has only one highest value. However, since this is a manufacturing process, we need to preserve the relationship between the variables, and thus the original magnitude in any differences carry a significant weight. For this reason, the principal component analysis was done using a covariance matrix, rather than a correlation matrix

Step 1: Data Reduction

To complete our PCA, we used the programming language R. Using built in R-code, and specifying the use of a covariance matrix, we were able to come up with a characterization of principal components versus the variances. However, this characterization is not enough, as this still leaves 209 variables. In order to find out the vital few variables that explain a large amount of the variance in the data, we need to look at the Scree and Pareto plots to reduce the dimension. These plots use the results of the PCA—namely the eigenvalues of each component—and graphically show the importance of each principal component.

A Scree plot plots, the eigenvalue λ_i versus principal component *i*. We look for an elbow (bend) in the plot, and take the principal components to use as the number at the bend: at this point the remaining eigenvalues are relatively small and all about the same size (Zou, 2019).

A Pareto Plot orders the eigenvalues from largest to smallest, and it is used to compare the eigenvalue-Principal component pair (λ_i) against the variance. The goal is to select the eigenvalues whose aggregated effects can explain a good percentage of the total variation in the data (Zou, 2019). The plots that we got are below:

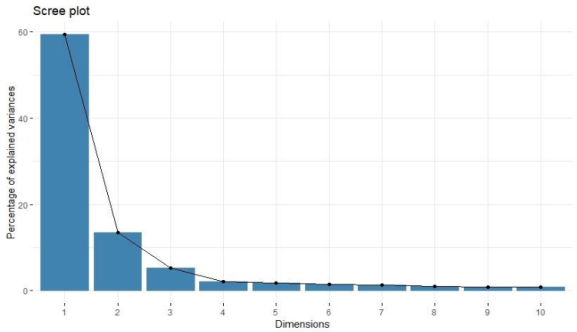


Fig 01 Scree Plot of Results. The x-axis has dimensions, and the y-axis Percent of Variance Explained. The first four sum up to about 80% Variance explained.

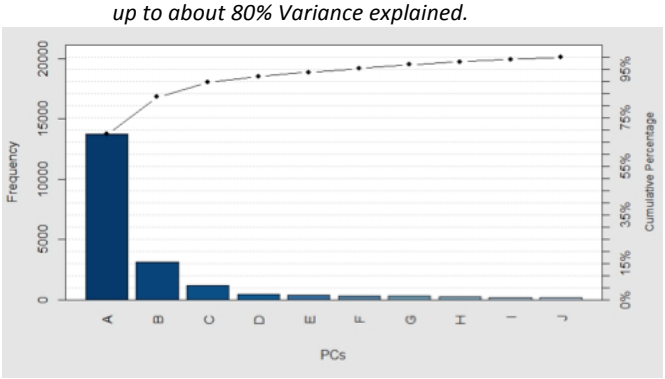


Fig 02 Pareto Chart of Results. Around fourth PC, 80% Cumulative Percentage

From our plot, we can see that having the first four components can explain a good the proportion of the data. The exact number of variance is found to be 80.1%.

Step 2: Use T^2 Chart to Start Multivariate Detection

We begin our multivariate detection with a Hotelling T^2 chart. A Hotelling T^2 is a popular chart that is multivariate detection. For our formulation, we use Case III, Phase I, when the mean and covariance are not known. In this case, the equations for T^2 is shown by the case where n=1 in the following figure. Table 01 at the bottom of the page summarizes the formulation for Hotelling T^2 charts. We used α=0.0027 for 3-sigma control limits.

We use built in R-code in order to complete our analysis, and use the T^2 chart to identify some the out of control chart data, and then move on to m-CUSUSM.

Step 3: Use m-CUSUM Chart

After, checking what is out of control on a T^2 chart, we move on to a m-CUSUM chart. The m-CUSUM chart can sense some of the data that the T^2 chart overlooked. The T^2 chart is good at sensing large spikes, large sustained shifts, but not the other way around. In order to account for these changes, we have to use m-CUSUM and m-EWMA. The one that is used is up to preference, but in practice, m-CUSUM is preferred as it has been around longer, and there is no appreciable advantage to using m-EWMA. For this reason, we also go with m-CUSUM, and use literature to help guide our decisions on how to select values for k and h. The formulation for m-CUSUM is:

$$MC_i = \max\{0, (\mathbf{C}_i^T \mathbf{\Sigma}^{-1} \mathbf{C}_i) - \mathrm{k} \cdot n_i\}$$

$$\mathbf{C}_i = \sum_{j=i-n+1}^{i} X_j - \mu_0$$

$$n_i = \begin{cases} n_{i-1} + 1 & when\ MC_{i-1} > 0, keep\ accumulating. \\ 1 & otherwise, start\ CUSUM\ over. \end{cases}$$

Where k is a user defined constant (approximately half the statistical distance between means), and we set it to 1.5. R also asks for h, another constant that is used in m-CUSUM, and we set it to 6. Information from Hamed (2016) and Santos-Fernandez (2017) was used as guide in selecting values. α is set to 0.0027. Using the built in R package with this formulation, we run and see if we have any out of control data.

Step 4: Iterate between step T^2 and m-CUSUM

We now iterate between T^2 and m-CUSUM charts until both charts no longer sense that any of the data is out of control. Whenever we hit this point, it means that our data is in-control, and we have a result that can be applied to future data.

Scenarios	**Test statistic (T^2)**	UCL
$n = 1$	$T^2 \coloneqq (x_j - \underline{x})^T S^{-1} (x_j - \underline{x})$	approximately $\chi^2_{1-\alpha}(p)$
$n > 1$	$T^2 \coloneqq n\left(\underline{x} - \underline{\underline{x}}\right)^T S^{-1} \left(\underline{x} - \underline{\underline{x}}\right)$	$\frac{(m-1)(n-1)}{m(n-1)+1-p} F_{1-\alpha}(p, m(n-1)+1-p)$

Table 01 Hotelling T^2 equations for Case III, Phase I analysis

RESULTS AND CONCLUSION

Results

As mentioned previously, our PCA and data reduction investigation found that with 4 principal components, we could explain 80.1% of the variance in the data. Using these principal components, we ran iterations in R to distinguish between the in and out-of-control data. To do this, we iterated between, both T^2 and m-CUSUM until there were no more alarms detected (out-of-control points removed) by both charts. The results are summarized in the table below:

Itera-tion	Cycle	Control Chart	Out-of-Control Points Removed
1	1	T^2	12
	2	T^2	6
	3	T^2	2
	4	T^2	0
	Total	T^2	20
1	1	m-CUSUM	61
	2	m-CUSUM	19
	3	m-CUSUM	2
	4	m-CUSUM	7
	5	m-CUSUM	1
	6	m-CUSUM	0
	Total	m-CUSUM	90
2	1	T^2	6
	2	T^2	2
	3	T^2	0
	Total	T^2	8
2	1	m-CUSUM	0
	Total	m-CUSUM	0

Table 02 Result of the Trials

Overall, it took 2 iterations each on T^2 and m-CUSUM to remove a total of 118 data points, leaving with us with a total of 434 data points out the original 552. This amounts to about 78.6% of the original data being preserved.

We also present the initial and final charts in the T^2 and m-CUSUM iterations, as a graphical representation of the before and after out-of-control data removal. The full set of graphs that detail all iterations of data removal are shown in the Appendix. The x-axis represents observations, and the y-axis is the T^2 statistical index. As shown in Figure, outliers are found to locate in the beginning and tail section, and we will remove all these out-of-control data without further inspection since we don't have the information of what kind of manufacturing data we are analyzing. The process will keep iterating until no outlier is found from control charts.

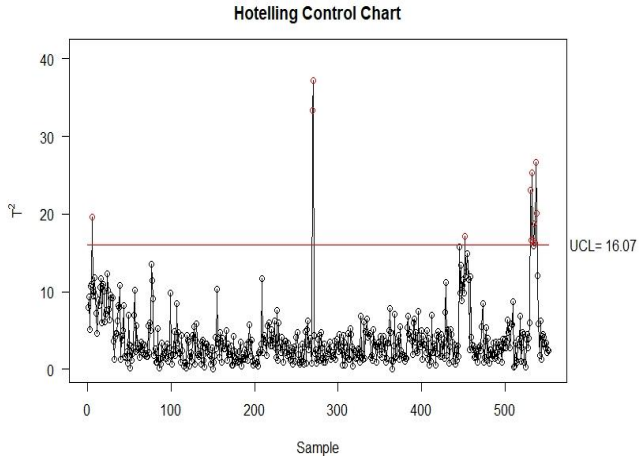


Fig 03 Initial Hotelling Chart. Initially 12 OOC points, UCL=16.07

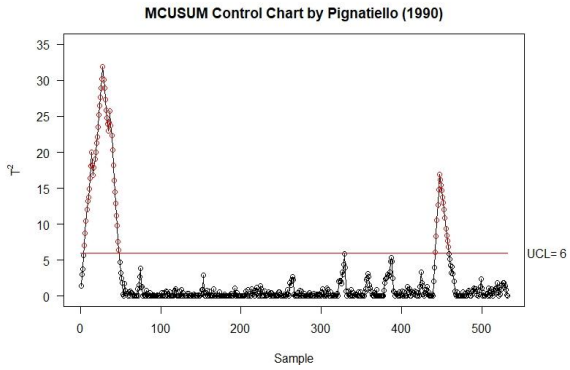


Fig 04 Initial m-CUSUM Chart. Initially, 61 OOC points, UCL=6

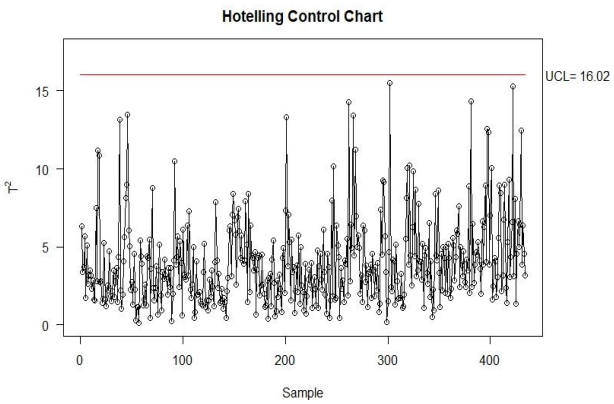


Fig 05 Final Hotelling T^2 Chart, All Points IC, UCL=16.02

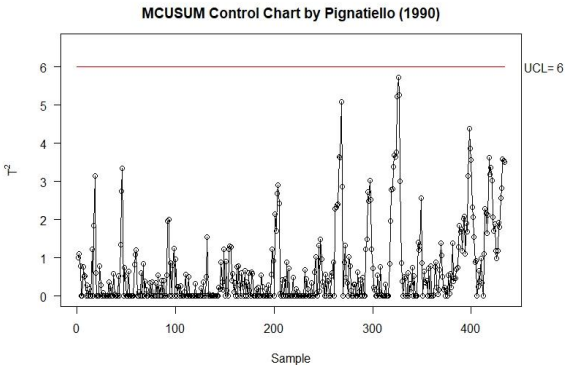


Fig 06 Final m-CUSUM Control Chart, All Points IC, UCL=6

Conclusion and Discussion

From our investigation, we were able to complete a PCA and data reduction that gave us 80.1% variance explained using 4 principal components. This gives our model not only a robust coverage of the variance (a large amount of the data is explained), but also shrinks the data to a point that noise effects are not that much of a worry and the process is efficient as well. Given that we are left with 434 data points, it means that a good portion of our original dataset was in-control. With this our phase 1 analysis is done, and we have built a model for in-control data. We can say that we 434 in-control points in the data, and the in-control model has a UCL of 16.02 for T^2 and 6 for m-CUSUM.

Other Steps

From what we have learned in this course, we do not have many options when facing phase 1 problems, and many research papers have indicated that phase 1 analysis tools are lacking compared to phase 2 analysis. In Appendix B, we will show several methods that have been suggested to help phase 1 analysis.

REFERENCES AND SOFTWARE

References

Fricker, R. (2013). Comparing Methods to Better Understand and Improve Biosurveillance Performance. In *Introduction to Statistical Methods for Biosurveillance: With an Emphasis on Syndromic Surveillance* (pp. 281-302). Cambridge: Cambridge University Press. doi:10.1017/CBO9781139047906.011

Hamed, M S, et al. Scientific Advances Publishers. "MCUSUM Control Chart Procedure: Monitoring The Process Mean With Application ." *Journal of Statistics: Advances in Theory and Applications*, vol. 16, no. 1, 2016, pp. 105–132. doi:http://dx.doi.org/10.18642/jsata_7100121721.

Santos-Fernandez, E. (2017, May 29). MSQC-package: Multivariate Statistical Quality Control in MSQC: Multivariate Statistical Quality Control. Retrieved from https://rdrr.io/cran/MSQC/man/MSQC-package.html

Ding, Y., Zeng, L., & Zhou, S. (2006). Phase I analysis for monitoring nonlinear profiles in manufacturing processes. Journal of Quality Technology, 38(3), 199-216.

Jones-Farmer, L. A., Woodall, W. H., Steiner, S. H., & Champ, C. W. (2014). An overview of phase I analysis for process improvement and monitoring. Journal of Quality Technology, 46(3), 265-280.

Ding, Y., Zeng, L., & Zhou, S. (2006). Phase I analysis for monitoring nonlinear profiles in manufacturing processes. Journal of Quality Technology, 38(3), 199-216.

Jones-Farmer, L. A., Woodall, W. H., Steiner, S. H., & Champ, C. W. (2014). An overview of phase I analysis for process improvement and monitoring. Journal of Quality Technology, 46(3), 265-280.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Dubourg, V. (2011). Scikit-learn: Machine learning in python. Journal of Machine Learning Research, 12(Oct), 2825-2830.

Sullivan, J. H. (2002). Detection of multiple change points from clustering individual observations. Journal of Quality Technology, 34(4), 371-383.

Truong, C., Oudre, L., & Vayatis, N. (2018). Ruptures: Change point detection in python. arXiv Preprint arXiv:1801.00826.

Zou, N. (2019). "Description of Course Project." ISEN 614-600 Handout. Handout Retrieved from https://tamu.blackboard.com

Zou, N. (2019). "Class Notes." ISEN 614-600 Handout. Handout Retrieved from https://tamu.blackboard.com

Software: Packages

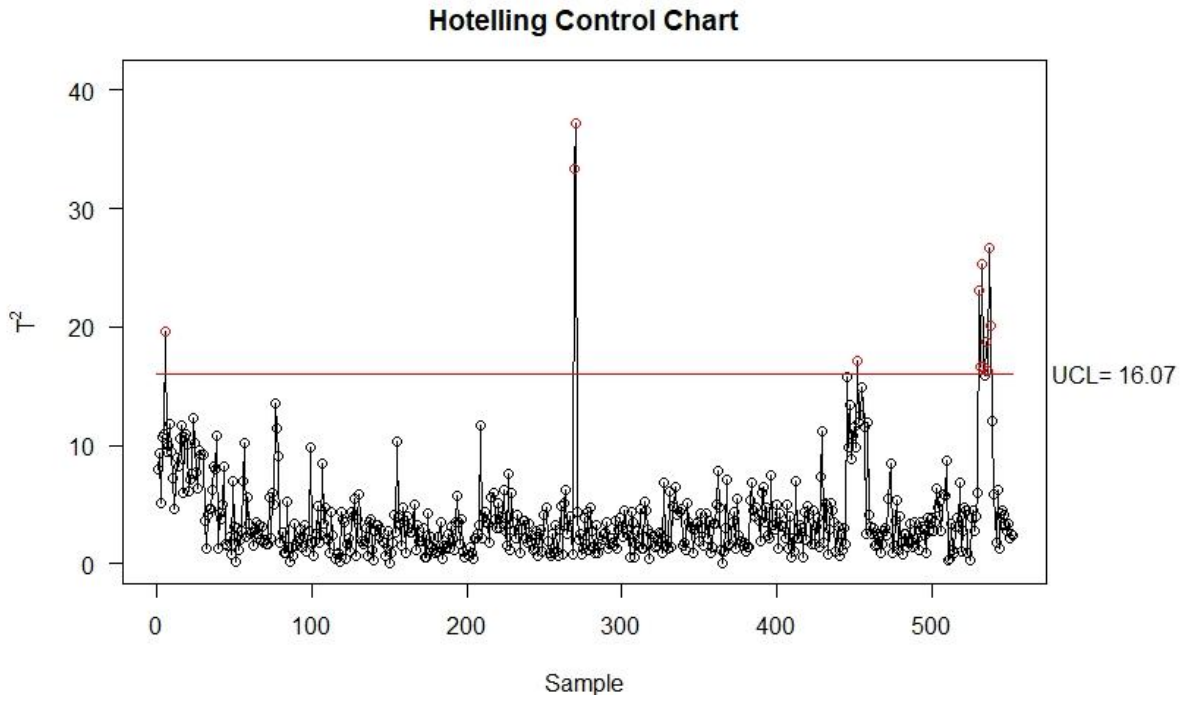
R: qcc, MSQC, factoextra

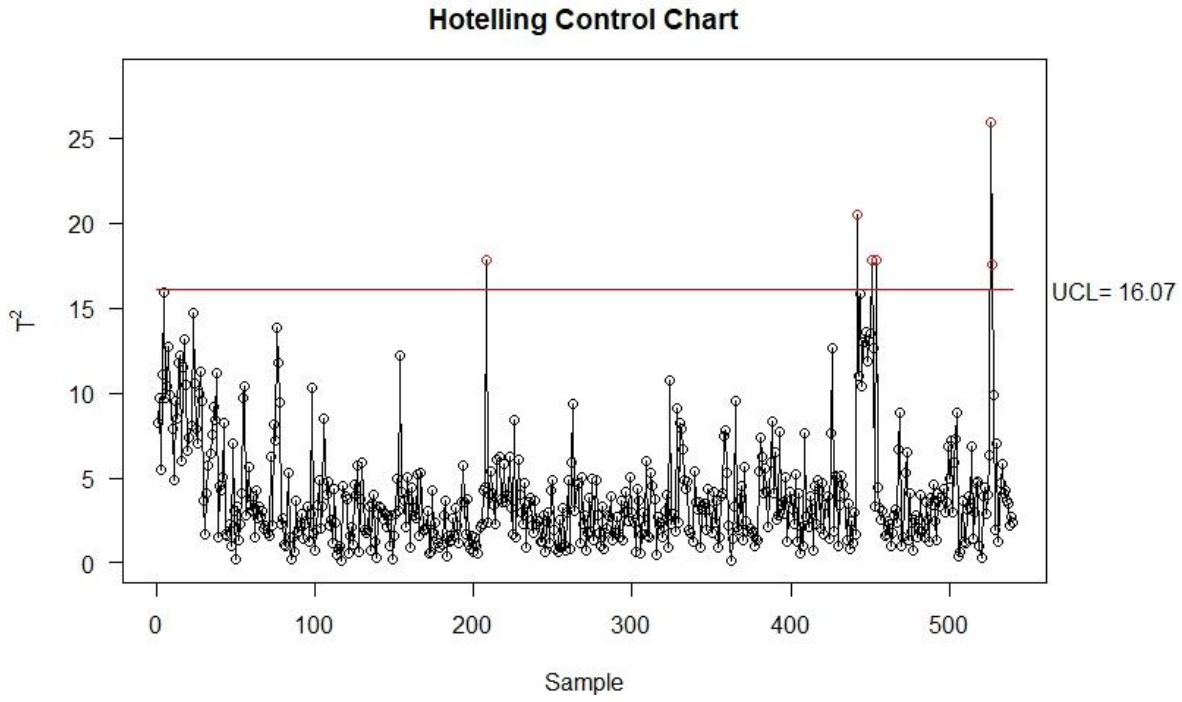
Python

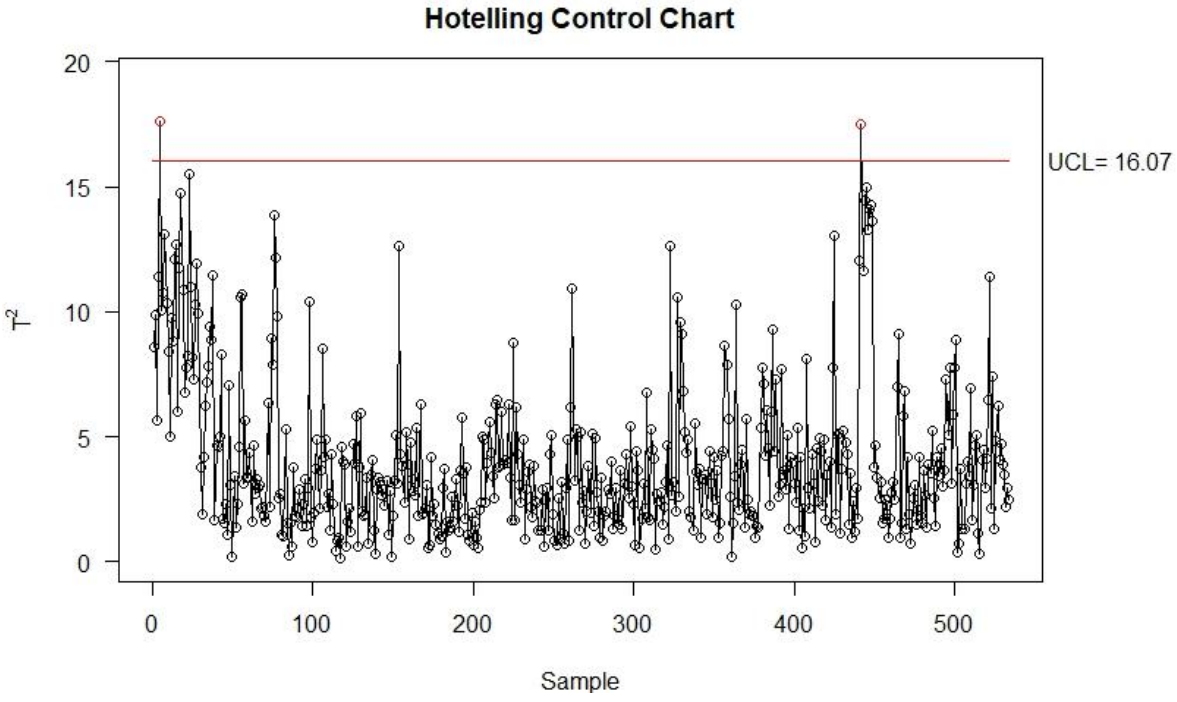
APPENDIX

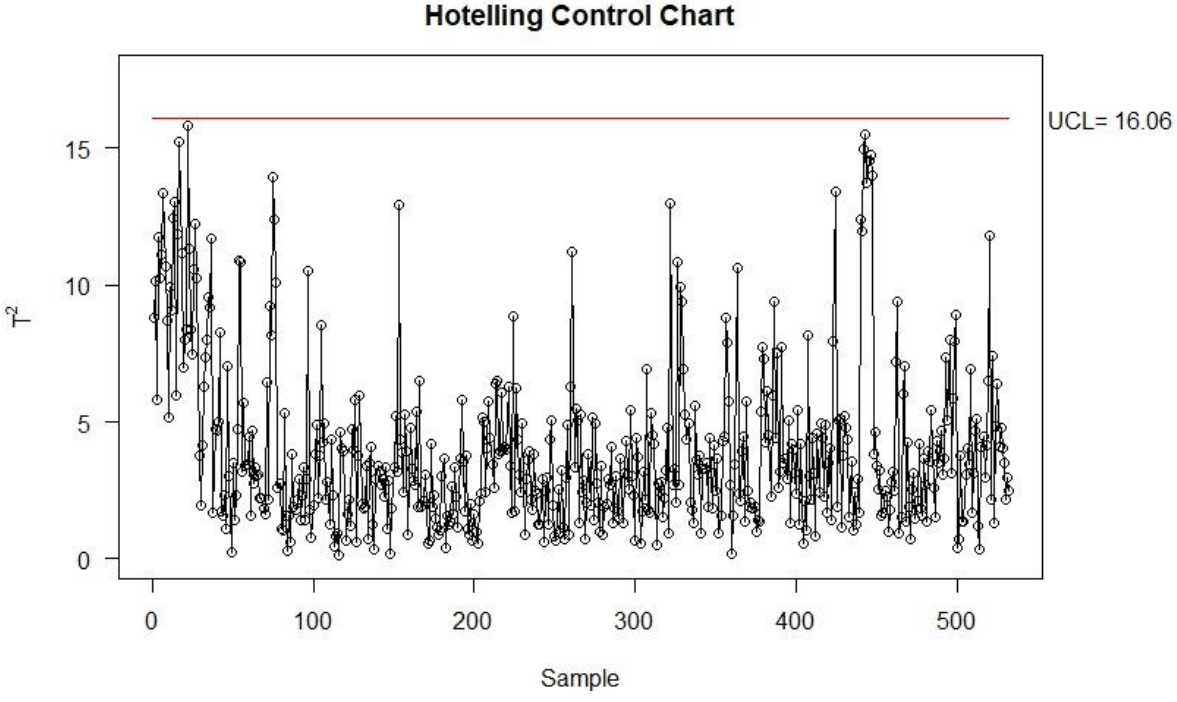
Appendix A: All Charts from iterations

T^2 – First set of iterations: 20 out-of-control points removed.

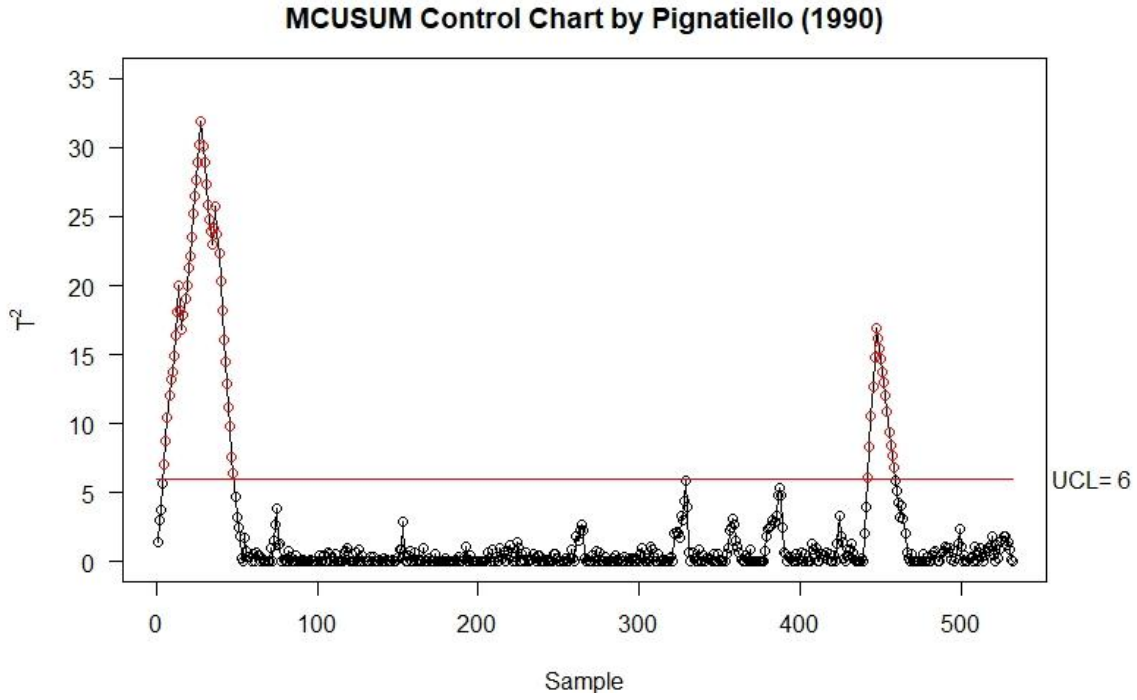


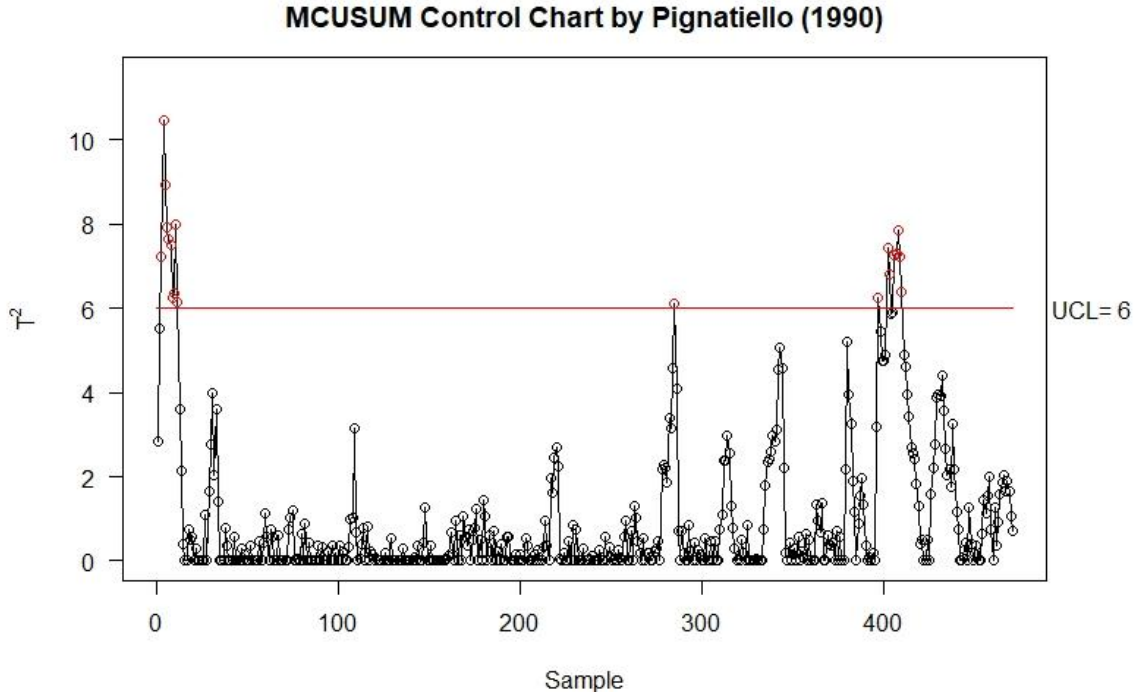


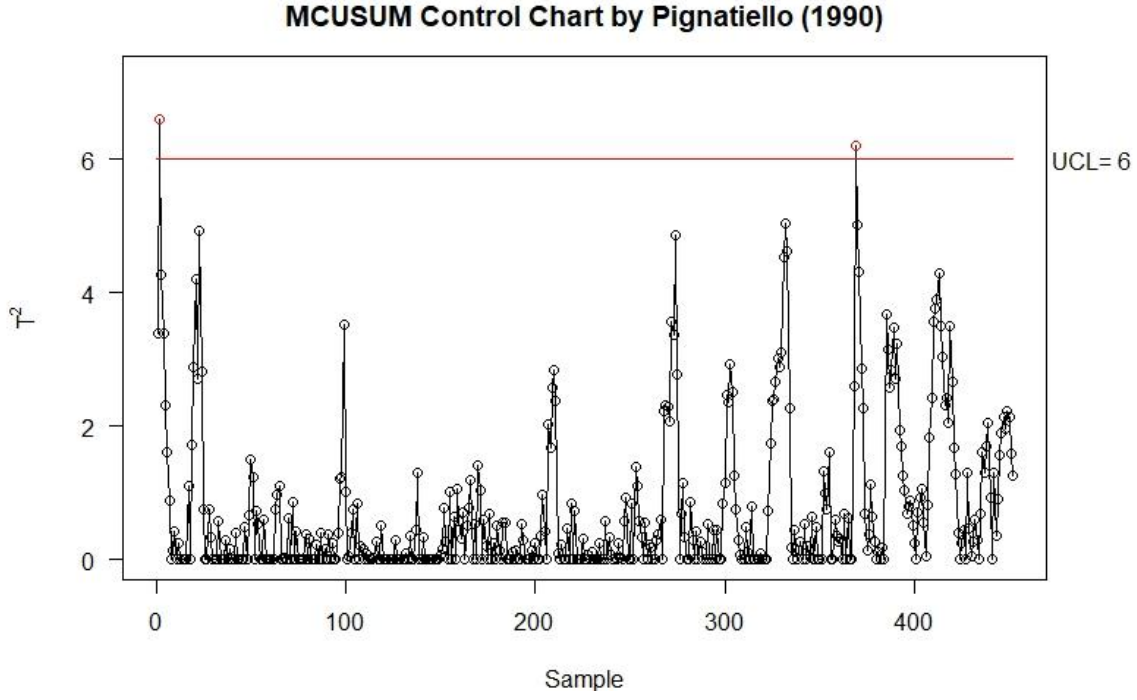

Hotelling Control Chart
20
15
10
5
0
T^2
0
100
200
300
400
500
Sample
UCL= 16.07

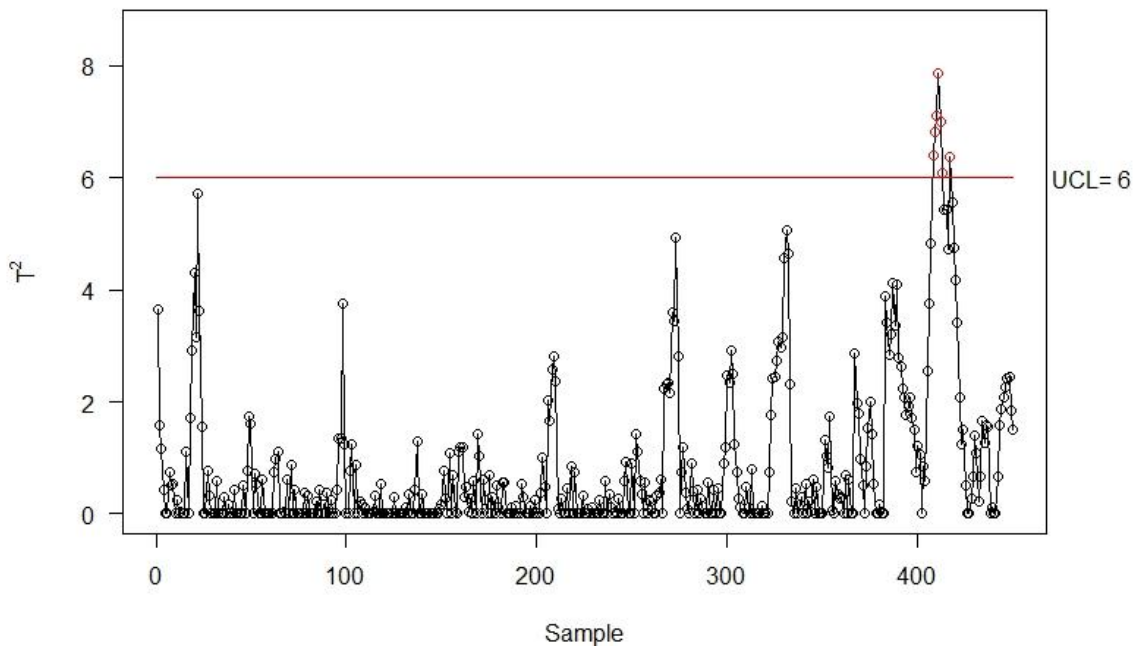

Hotelling Control Chart
15
10
5
0
T^2
0
100
200
300
400
500
Sample
UCL= 16.06

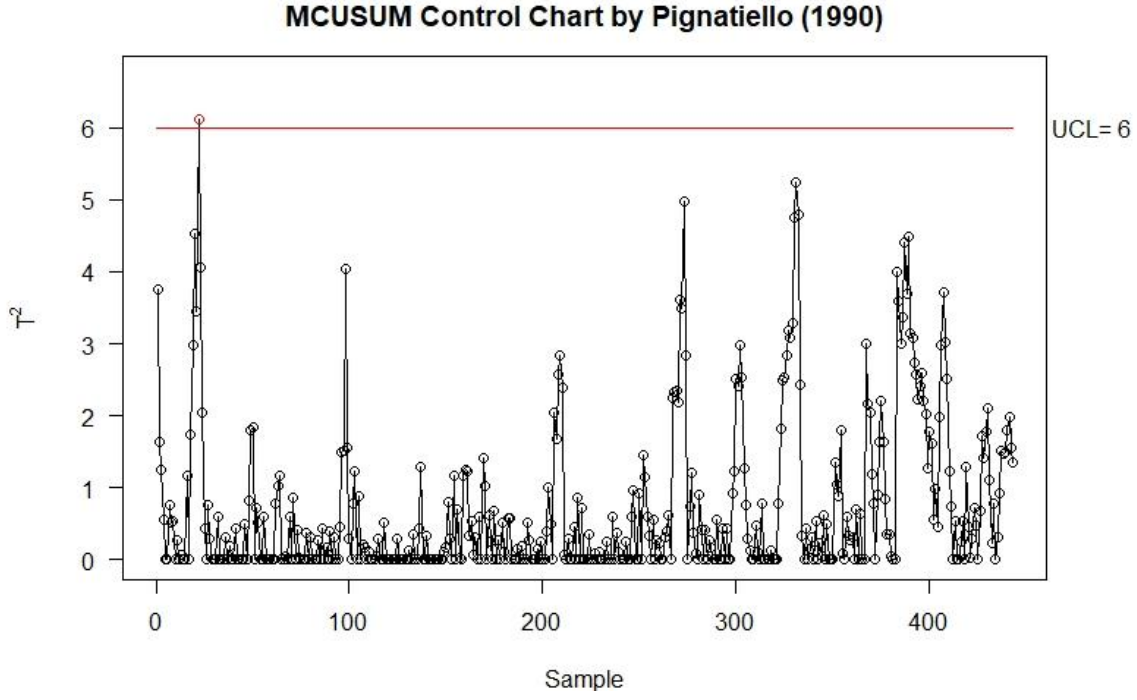
MCUSUM – First set of iterations: 90 out-of-control points removed.

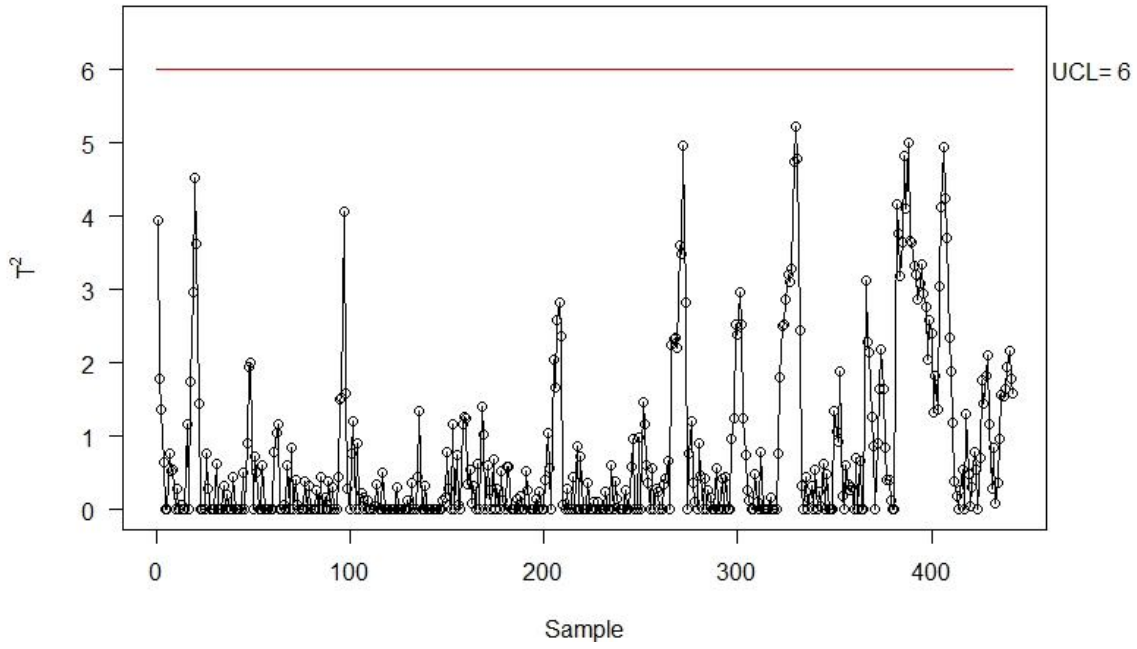




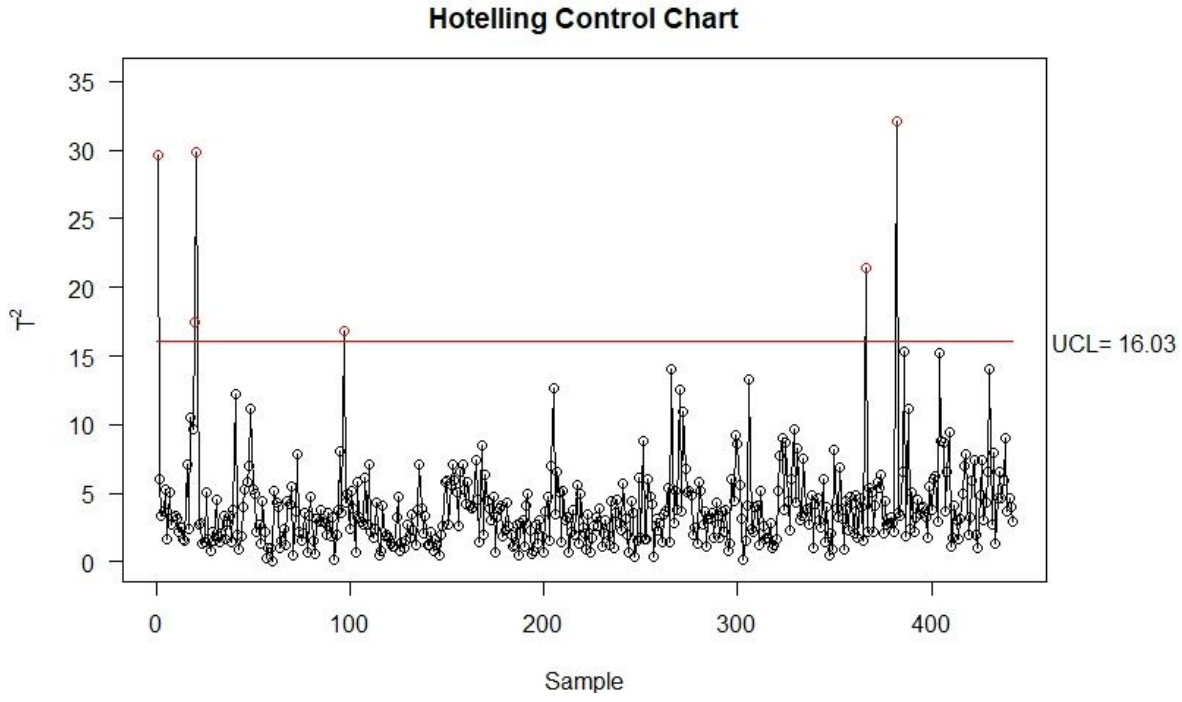

MCUSUM Control Chart by Pignatiello (1990)
UCL= 6
T^2
Sample
0
2
4
6
0
100
200
300
400

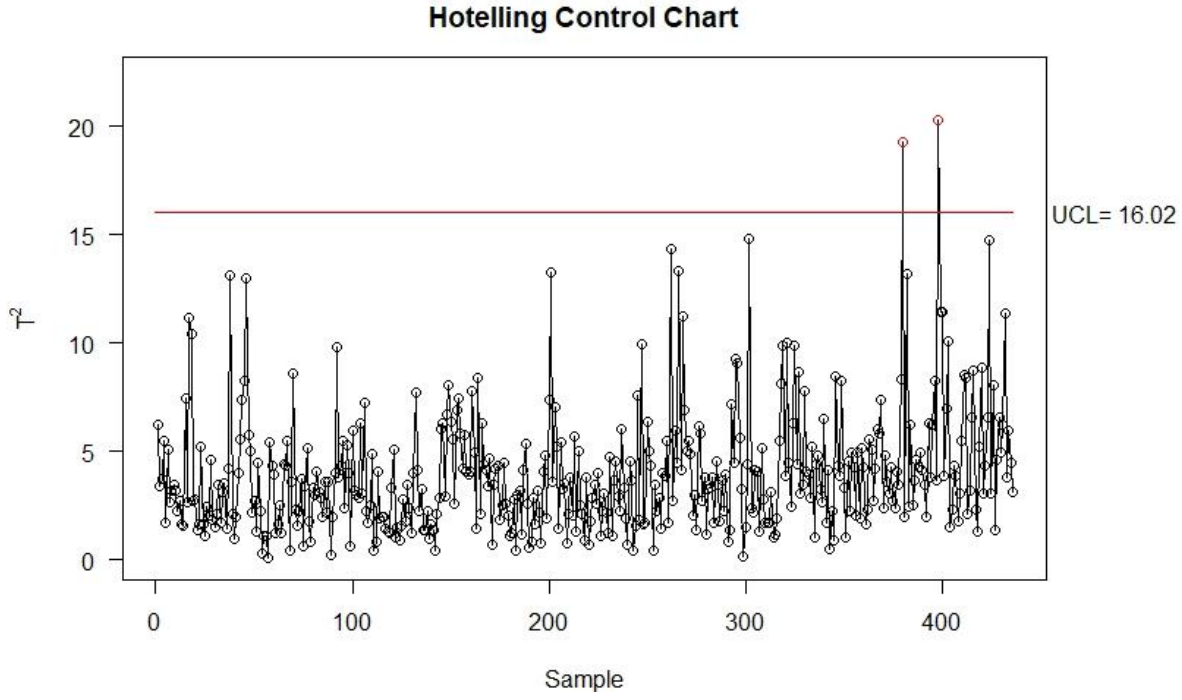

MCUSUM Control Chart by Pignatiello (1990)
UCL= 6
T^2
Sample
0
2
4
6
8
0
100
200
300
400

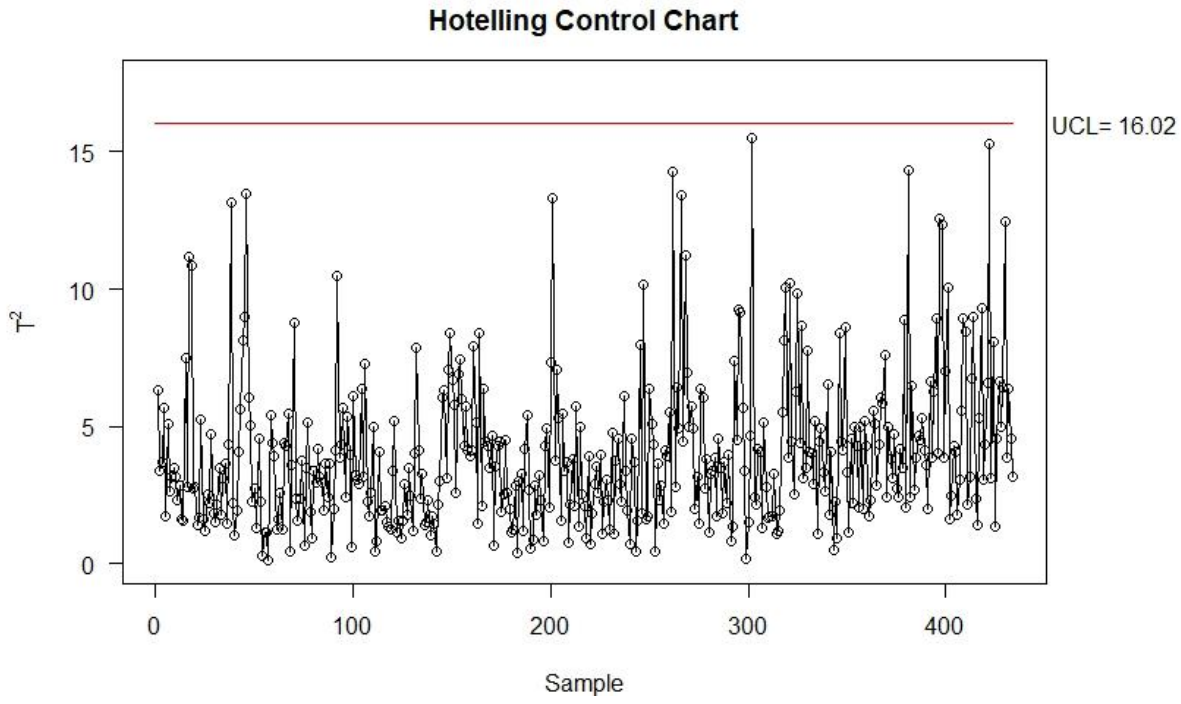

MCUSUM Control Chart by Pignatiello (1990)
UCL= 6
T^2
Sample
0
1
2
3
4
5
6
0
100
200
300
400


MCUSUM Control Chart by Pignatiello (1990)
UCL= 6
T^2
Sample
0
1
2
3
4
5
6
0
100
200
300
400

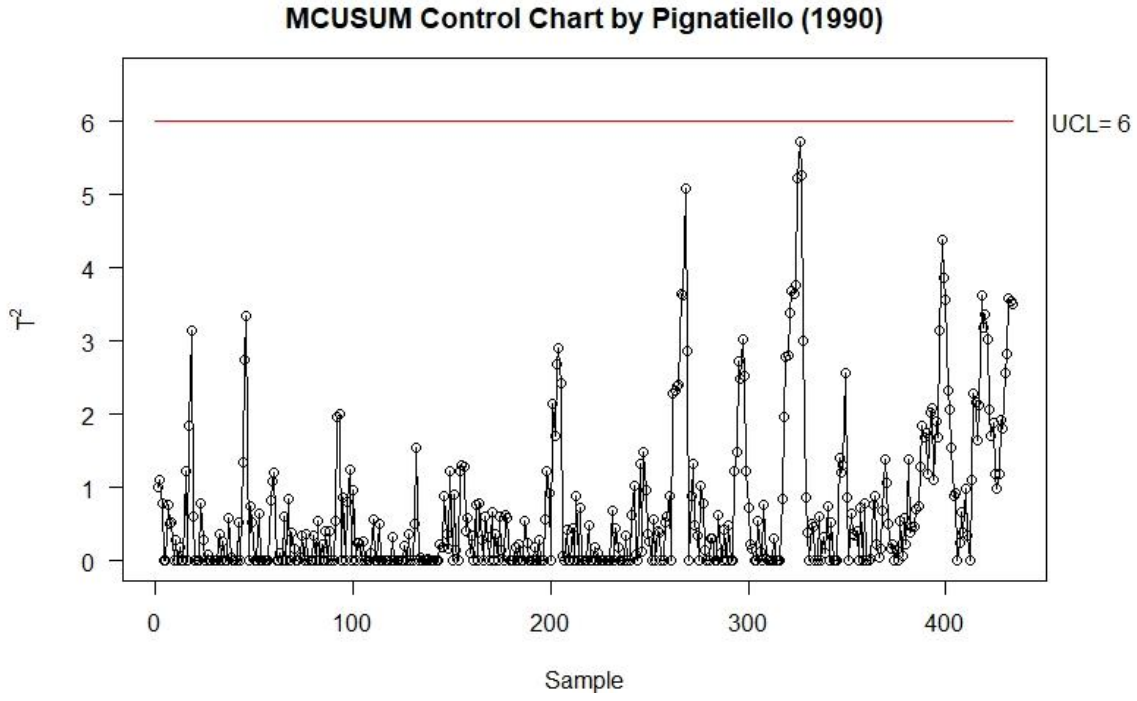
T^2 – Second set of iterations: 8 out-of-control points removed.







MCUSUM – First set of iterations: 0 out-of-control points removed.



Appendix B: Alternative method for phase 1 analysis

In the previous sections, we introduced PCA (Principal Components Analysis) and retrospective control chart (Hotelling T^2) to help us reducing data dimension and removing outliers. According to the definition of PCA, the most informative direction (or highest variability) will be used to span a new subspace, however, it does not guarantee that projected data in the new subspace can maximize the separation of any clustering structure that may exist in the origin data. In addition, projected data in new subspace is not linearly independent in each axis leading to complexity of telling each PCs' contribution to outliers. Hence, some researchers suggested a different dimension reduction method, which is called Independent Components Analysis (Ding, Zeng, & Zhou, 2006) .

Independent Components Analysis (ICA)

In short, ICA intends to find non-Gaussainity of the data which indicating the directions of distinct cluster (or called out-of-control cluster). For example, there exists a bivariate in-control data but contaminated by an out-of-control data cluster. As Figure 1. shown, when PCA is performed, it will chose e_1 direction, which contains largest variability of the data. On the other hand, ICA will chose direction e_2, which is the mean shift between out-of-control cluster and in-control cluster.

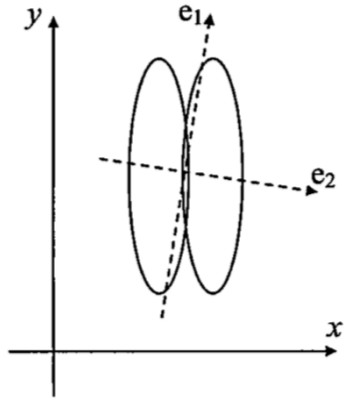


Figure 1. Difference of PCA and ICA

Mathematical concepts of ICA is far more complex than PCA, but sharing some similarity. In short, ICA focus on finding most non-Gaussainity as PCA trying to find maximum variability. The level of non-Gaussainity can be determined by following entropy function :

$$H(Y) = -\int f_Y(y)\log\left[f_Y(y)\right]dy = -E_Y[f_Y(y)]$$

Where *Y* is continuous random variables with density function $f_Y(y)$. When *f* is equal to Gaussian distribution we will get smallest value. In addition, another important property of ICA is that each axis of ICs is fully linear independent, suggesting that univariate control chart will be able to use in multivariate data set. In this report, we use *FastICA* method to conduct following ICA analysis which is implemented in *sklearn* open source package on Python (Pedregosa et al., 2011) .

Removing Outliers

For multivariate dataset, using Hotelling T^2 control chart is the most common way. Although T^2 is able to process multivariate data to generate a statistical index, its power of detecting outliers may not be ideal, especially in detecting small mean shift. Some multivariate detecting methods like m-CUSUM or m-EWMA were suggested to solve this issue, but both of these methods require pure in-control data to estimate its control limits with Monte-Carlo method. Hence, in phase 1 stage, we still need other options for data classifying. In this sections we will try different removing methods, since we already have applied ICA, which each axis is linear independent, we are able to use the simplest univariate control chart once again without concerning the correlation of each axis. Following the rule of retrospective control chart, if a point in one or few IC control chart is classified out-of-control then the point will be removed and steps will keep again until there is no more remove point. If this report, we use univariate IC analysis with *at least one rule*, which means if a specific data point is classified out-of-control by any of IC control chart then it will be removed. For *3* sigma condition, the corresponding alpha error is *0.0027* and total false alarm probability equals to $1-(-0.0027)^m$, where m equals to amount of ICs (Jones-Farmer, Woodall, Steiner, & Champ, 2014).

CPD

Another way to find out outlier is Change-Point method, which implies a distribution-free classification (Sullivan, 2002) . Since that previous mentioned outlier removing methods are based on assumption of the in-control data follows normal distribution. While for the real-world collected industrial data, our first hand historical data may not always follows normal-distribution and our assumption of normal distribution may lead to slow detecting issue. And Change-Point method is design to observe one or more sustained process change.

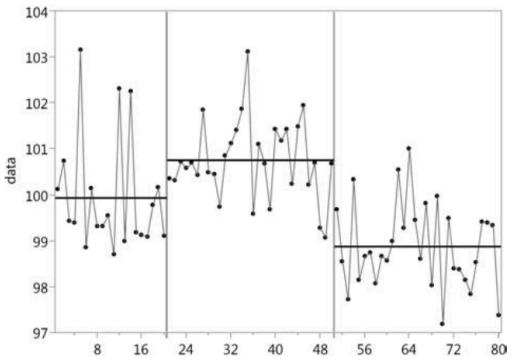


Figure 2. Illustration of data using change-point with the mean

In this report, we will not go through the full algorithms of Change-Point method, while with a brief introduction. Suppose, there exists *m* independent observation with *R* shifts in the mean, and the shift locations are T_r , $r = 1,\ldots, R$,

such that $0 < T_1 < \ldots < T_R < m$. So in the end of the algorithm, our data should be cut into several segments, each segment means a mean shift. Also, the algorithm will try achieve the least R instead of treating each point as a segment. After the processing is done, the outlier or shifted segment would be decided to remove from interpreting its physical meanings. In this report, since all the physical meanings of each characteristics are omitted, we will directly remove the distinct segments and consider the remain data as in-control data. An open source package called *ruputure* are using to implement change-point detection on Python (Truong, Oudre, & Vayatis, 2018).

Assumption

As mentioned before, ICA is aimed to find non-Gaussian direction of the data, we should assume that for the given dataset in this project, the in-control data follows Gaussian distribution but become non-Gaussian when combining with out-of-control data, and amount of out-of-control data should not surpass the number of in-control data. By doing so, we able to find a solution in ICA. Plus, for simplify the progress for finding optimize number of ICs, we assume that the number of ICs is equal to PCs, and data dose not standardized in order to save more information since all physical meaning are omitted.

Results

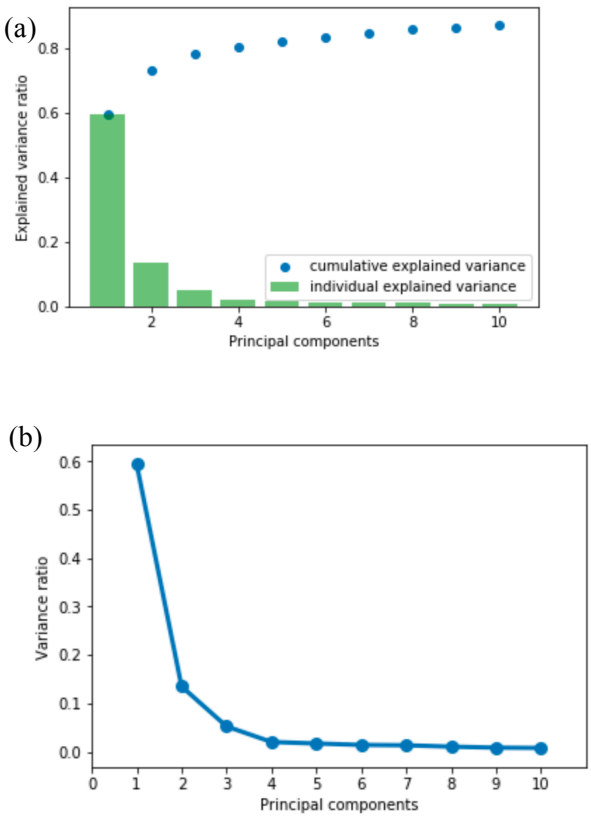


Figure 3. (a) Pareto chart and (b) Scree chart for selecting PCs and ICs

Based on the assumption we mentioned before, we use 4 ICs to transform origin data into a new subspace for following analysis. And each IC's profiles and histograms are present in Figure 4.

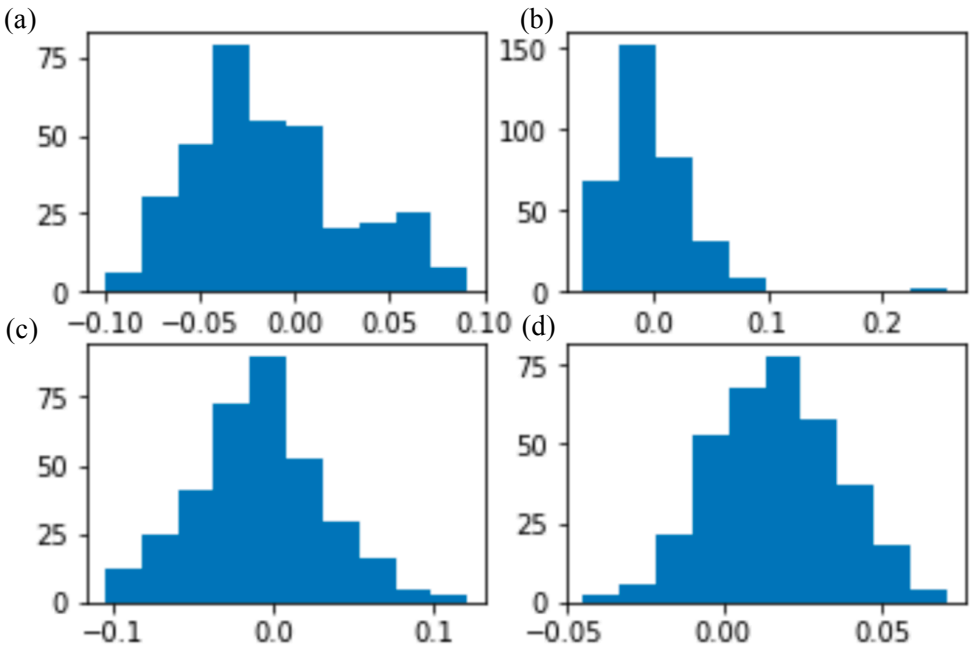


Figure 4. (a) IC1's histogram. (b) IC2's histogram. (c) IC3's histogram. (d) IC4's histogram

Univariate mean chart and CPD

In this section, we will applied univariate control chart and changed-point detection on four ICs to sperate out-of-control cluster. As we known from the description of origin dataset, the sample size equals to 1, so univariate mean chart will take each observation as sample mean. We use either-or out-of-control rule to filter outliers, the observation will be removed once it is classified out of control by any of IC chart. The control limit is determined by 3 sigma rule with alpha error 1.07% for applying "either-or" rule with 4 ICs, The control limit will be updated by new data without outliers and keep updating until no outliers occur. For convenience, we only present the first iteration and the very last one, as figure 5 and figure 6 show, which 480 observation remained as in-control data and total 72 observations are considered to be out-of-control.

On the other hand, a procedure combining with ICA and CPD (changed point detection) is executed. The results are shown in Figure 7. The First CPD shows 6 change points and dividing dataset into 7 segments. It is obviously to see process shift happens in segment 1 and 5 in 1st CPD, but if we look at the rest of CPDs, for example, 3rd CPD which is suggesting that after observation #452 the process has changed. By excluding all these shifts, we obtain a new data set of in control data with number of observations is 345. If we look at the histogram of each method's result,

the profile looks similar to each other. We may conclude that for practical phase 1 analysis, choosing ICA with CPD can covered more situation since it need no assumption of samples.

Now, go back to our different in-control data sets as shown in figure 8, we need to decide which one is better. If we follow the assumption which in-control data are sampled from normal distribution then we should use the first method because it gives us a much more normal distribution-like result. However, if our assumptions are made wrong, the in-control data are not following normal distribution or amount of in-control data is not sufficient, we should try to use CPD.

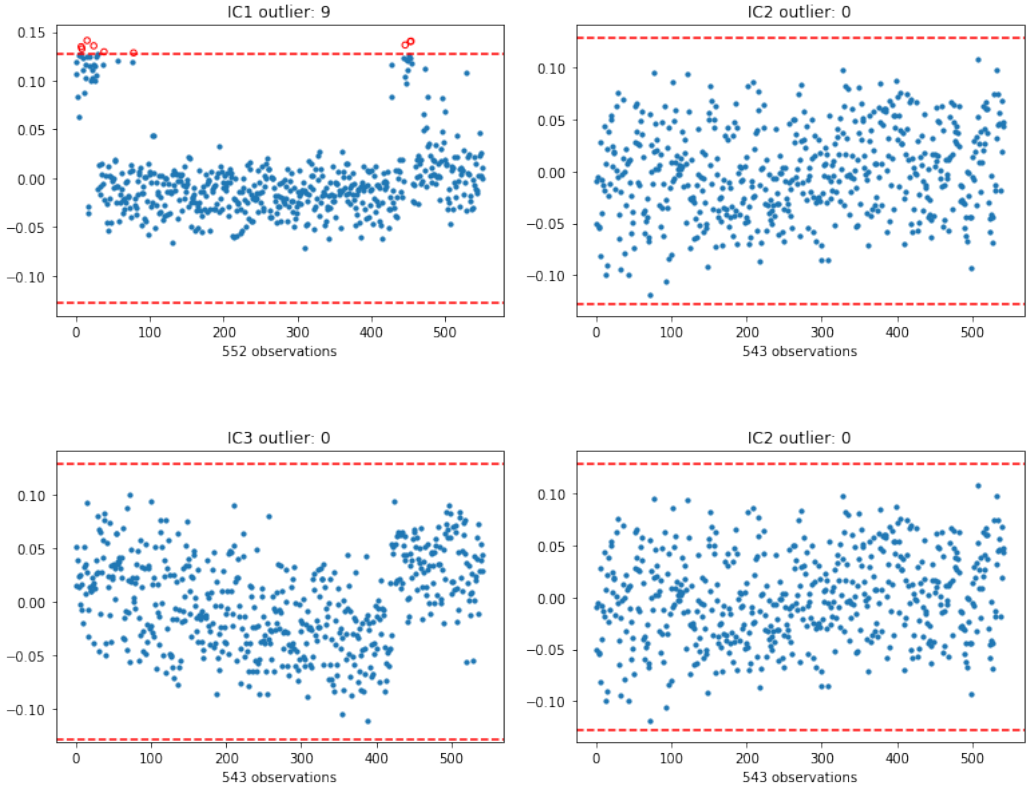


Figure. 5 the First step of univariate mean chart

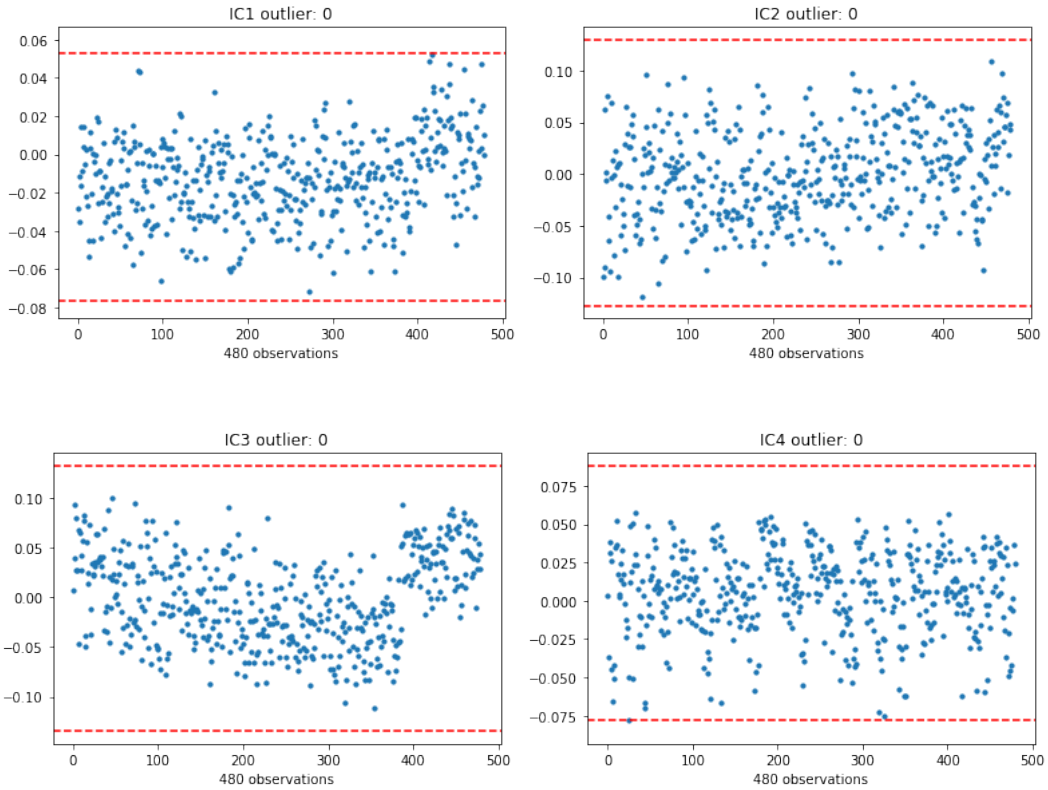
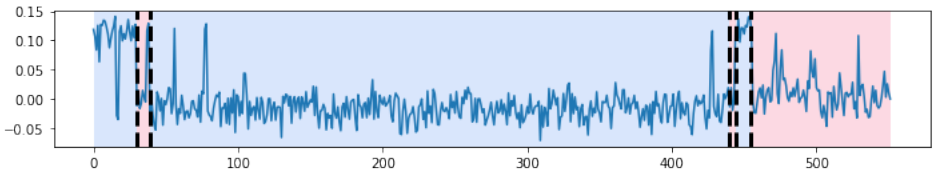
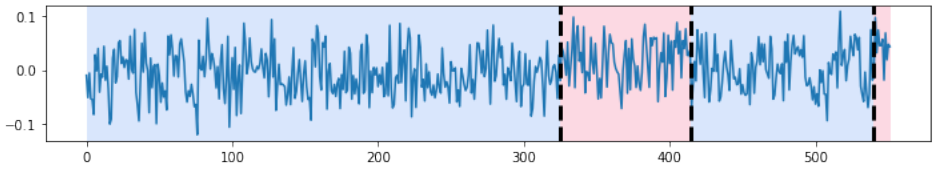


Figure 6. The final step of univariate mean chart

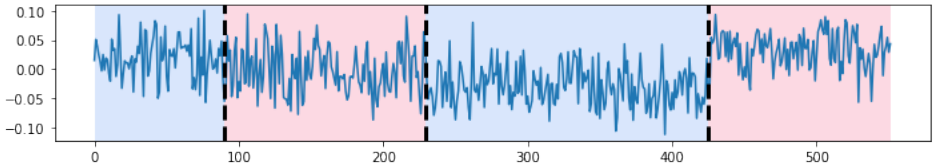
[30, 40, 440, 445, 455, 552]



[325, 415, 540, 552]



[90, 230, 425, 552]



[530, 540, 552]

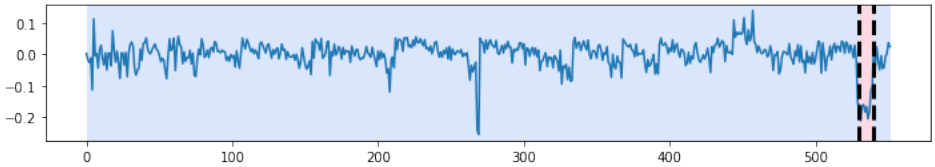


Figure 7. ICA with CPD, data lists above the figures are the change points

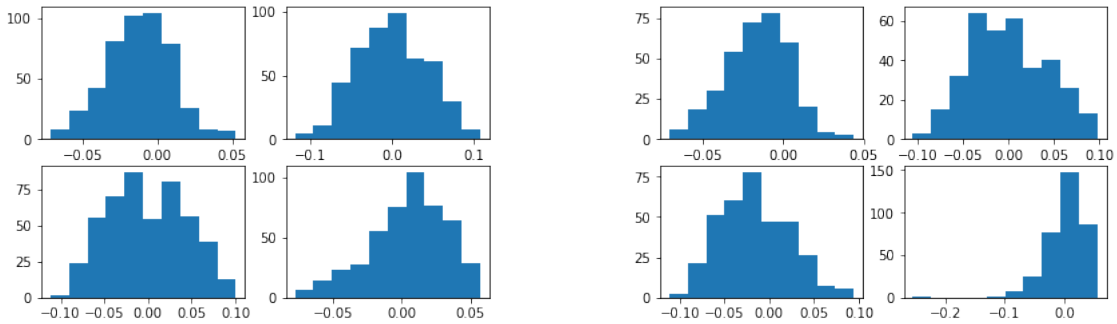


Figure 8. Left: histogram of in-control data obtained from mean chart. Right: histogram of in-control data obtained from CPD