Smart System Monitoring for High Dimensional Multistage Manufacturing Processes

by

Mohammadhossein Amini

B.S., Khaje Nasir Toosi University of Technology, Iran, 2007
M.S., Kansas State University, 2015

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Industrial and Manufacturing Systems Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Abstract

This research studies a system-wide approach to monitor product quality in real time to avoid manufacturing defects in high dimensional multistage processes. Traditional control charts have been widely used in various manufacturing industries due to their simplicity. However, in today's complex manufacturing processes, these charts are not efficient anymore. A complex manufacturing process may include multiple stages with sensors embedded throughout the processes that generate a huge amount of data in high dimensions. Since the numbers of stages and parameters are usually very large, traditional control charts are incapable of handling a multistage high-dimensional problem mainly due to the problem of false alarm rates of simultaneous monitoring.

Industry 4.0 and Internet of things (IOT) provides opportunities to achieve better quality products toward a zero defect system. Currently, data is either thrown away or stored in unused databases. In a inefficient approach called "fire-fighting", when there is a decline in the quality, process engineers go back to archived process data to figure out the problem. However, due to various reasons such as messy and unclean data, outdated data, and common manufacturing data features such as complexity and dimensionality issues, this process may take a long time. In the best cases, researchers provide classification-based process monitoring techniques to use the manufacturing data. However, the state-of-the-art classification-based process monitoring techniques usually provide quality predictions at the end of the manufacturing process and provide no chance to fix the problem. In addition, knowing high dimensional, unbalanced, and newly released manufacturing data, the literature is largely silent on providing a comprehensive study addressing those issues. While addressing above mentioned challenges, the proposed research delivers a stage-wise process monitoring which provides plenty of time for engineers to fix the

process before the last point. Then, based on the results from the predictive models, adjustable process parameters can be altered to avoid potential defects. The proposed research relies on predictive models which are built on a series of classifiers.

The proposed research is implemented in two different manufacturing application – additive manufacturing (AM) and semiconductor production. The proposed research in the AM industry called the Multi-Layer Classification Process Monitoring (MLCPM) is applied in the Laser Powder Bed Fusion (LPBF) metal printing process. In the semiconductor manufacturing industry, we applied the proposed method on a very imbalanced high dimensional production data called SECOM (SEmiCOnductor Manufacturing) which is publicly available through the UCI repository lab [1]. In this study, we examined various classification models and sampling approaches to find the best results in terms of specific metrics chosen regarding the imbalanced nature of the problem. The applied case studies show the effectiveness of the proposed framework in terms of accurately predict the real-time production quality state. The chance of predicting the quality of the process before the last step provides chances to reduce the waste and save cost and time in the production systems.

Smart System Monitoring for High Dimensional Multistage Manufacturing Processes

by

Mohammadhossein Amini

B.S., Khaje Nasir Toosi University of Technology, Iran, 2007
M.S., Kansas State University, 2015

A DISSERTATION

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Industrial and Manufacturing Systems Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Approved by:

Major Professor
Dr. Shing I. Chang

Copyright

Abstract

This research studies a system-wide approach to monitor product quality in real time to avoid manufacturing defects in high dimensional multistage processes. Traditional control charts have been widely used in various manufacturing industries due to their simplicity. However, in today's complex manufacturing processes, these charts are not efficient anymore. A complex manufacturing process may include multiple stages with sensors embedded throughout the processes that generate a huge amount of data in high dimensions. Since the numbers of stages and parameters are usually very large, traditional control charts are incapable of handling a multistage high-dimensional problem mainly due to the problem of false alarm rates of simultaneous monitoring.

Industry 4.0 and Internet of things (IOT) provides opportunities to achieve better quality products toward a zero defect system. Currently, data is either thrown away or stored in unused databases. In an inefficient approach called "fire-fighting", when there is a decline in the quality, process engineers go back to archived process data to figure out the problem. However, due to various reasons such as messy and unclean data, outdated data, and common manufacturing data features such as complexity and dimensionality issues, this process may take a long time. In the best cases, researchers provide classification-based process monitoring techniques to use the manufacturing data. However, the state-of-the-art classification-based process monitoring techniques usually provide quality predictions at the end of the manufacturing process and provide no chance to fix the problem. In addition, knowing high dimensional, unbalanced, and newly released manufacturing data, the literature is largely silent on providing a comprehensive study addressing those issues. While addressing above mentioned challenges, the proposed research delivers a stage-wise process monitoring which provides plenty of time for engineers to fix the

process before the last point. Then, based on the results from the predictive models, adjustable process parameters can be altered to avoid potential defects. The proposed research relies on predictive models which are built on a series of classifiers.

The proposed research is implemented in two different manufacturing application – additive manufacturing (AM) and semiconductor production. The proposed research in the AM industry called the Multi-Layer Classification Process Monitoring (MLCPM) is applied in the Laser Powder Bed Fusion (LPBF) metal printing process. In the semiconductor manufacturing industry, we applied the proposed method on a very imbalanced high dimensional production data called SECOM (SEmiCOnductor Manufacturing) which is publicly available through the UCI repository lab [1]. In this study, we examined various classification models and sampling approaches to find the best results in terms of specific metrics chosen regarding the imbalanced nature of the problem. The applied case studies show the effectiveness of the proposed framework in terms of accurately predict the real-time production quality state. The chance of predicting the quality of the process before the last step provides chances to reduce the waste and save cost and time in the production systems.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Chang for his valuable guidance, supports, and feedbacks. I am thankful to Dr. Chae for his dedication, organization, enthusiasm and hard work. I would also like to thank my committee members, Dr. Easton, Dr. Wu, and Dr. Robby for serving as my committee. I would especially like to thank Dr. Jaberi for giving me the chance to serve as a data scientist and collaborate with great researchers in 1Data group. Moreover, I am thankful to my beloved wife for her support, encouragement, and love.

Dedication

The dissertation is dedicated to my mother, my wife, and my beloved daughter.

Chapter 1 - Introduction

This research studies a system-wide multi-stage, real-time process monitoring approach for high dimensional multistage processes using predictive classification models. A multistage system contains several steps needed to produce a product or perform a service. Examples of multistage systems include semiconductor manufacturing, assembly lines, and additive manufacturing [2]. In multistage systems, each stage may have multiple characteristics. This kind of general multistage process may constitute a high dimensional vector in which each element contains either the status or measures of a process parameter or quality characteristics at the time of measurement. Note that the timestamps for each measurement may be different in different stages but can be strung together. This high-dimensional vector can be used directly for process monitoring or diagnosis during production and post-production.

While manufacturing processes have seen much improvement, process monitoring techniques such as control charting have not experienced a transformative improvement since Shewhart [3] introduced X-bar and R charts in the 1920s. For example, in a car assembly line, the body dimension inspection is an important stage where coordinate measuring machines generate multiple data points [4]. Although all dimensions fit into their tolerance and no control chart indicates out of control, however, door fitting in a later assembly stage may leave large gaps in some areas. Existing process control methods such as control charts do not pass the dimensions information into later stages due to the sheer data volume or dimensional constraints in the data set itself.

Since 1920, many studies have been published to incrementally improve process monitoring techniques. Examples include cumulative sum (CUSUM), exponential weighted moving average (EWMA), and Multivariate techniques such as Hotelling T^2 [5], principal

component analysis (PCA), and generalized likelihood ratio test (GLRT) [6]. However, these process monitoring methods fail to answer the challenges posed by future manufacturing environments where abundant sensor data on process parameters and semi-finished parts are readily available for system-wide monitoring.

For example, one of the visions of Industry 4.0 calls for a smart quality management system leveraging the real-time use of process data to monitor product quality [7]. Nowadays, machine learning techniques or loosely called *AI* (Artificial Intelligence) have been adopted for decision making in pockets of automation. Specifically, process data has still been used in an isolated manner regarding process monitoring practices. Control charts are implemented only for critical quality characteristics rather than on process parameters, of which data is either thrown away or stored in huge databases. This phenomenon is dubbed "dark data" in that most data has never been used for any purpose. Some manufacturers only use process data in a "fire-fighting" mode when data is dug out for root-cause analysis when there is a decline in product quality. To diagnose what parameters, which stage, and when such a discrepancy took place, process engineers have to examine archived process data, which may take a long time due to various reasons such as messy and unclean data, outdated data, complexity, and dimensionality issues [6].

Answering these challenges, researchers have provided classification-based process monitoring techniques for manufacturing data [6]. However, these methods usually provide quality predictions at the end of the manufacturing process and therefore provide no chance to fix the problem during production. Moreover, data-driven approaches generally face different challenges such as high dimensionality, updating process, and rare faulty samples. Addressing these challenges, we propose a stage-wise process monitoring model which provides prognosis information related to future product quality following the stage of the current production before

the last stage is reached. The next section provides a background study regarding quality engineering and process monitoring studies for high dimensional multistage systems.

1.1. Process Monitoring in High Dimensional Multistage Systems

Multistage systems are very common in practice in various industries. However, quality control of such systems is very complex since the variation of each stage does not solely depend on itself but may come from upstream stages. Figure 1 illustrates a diagram of the multistage system.

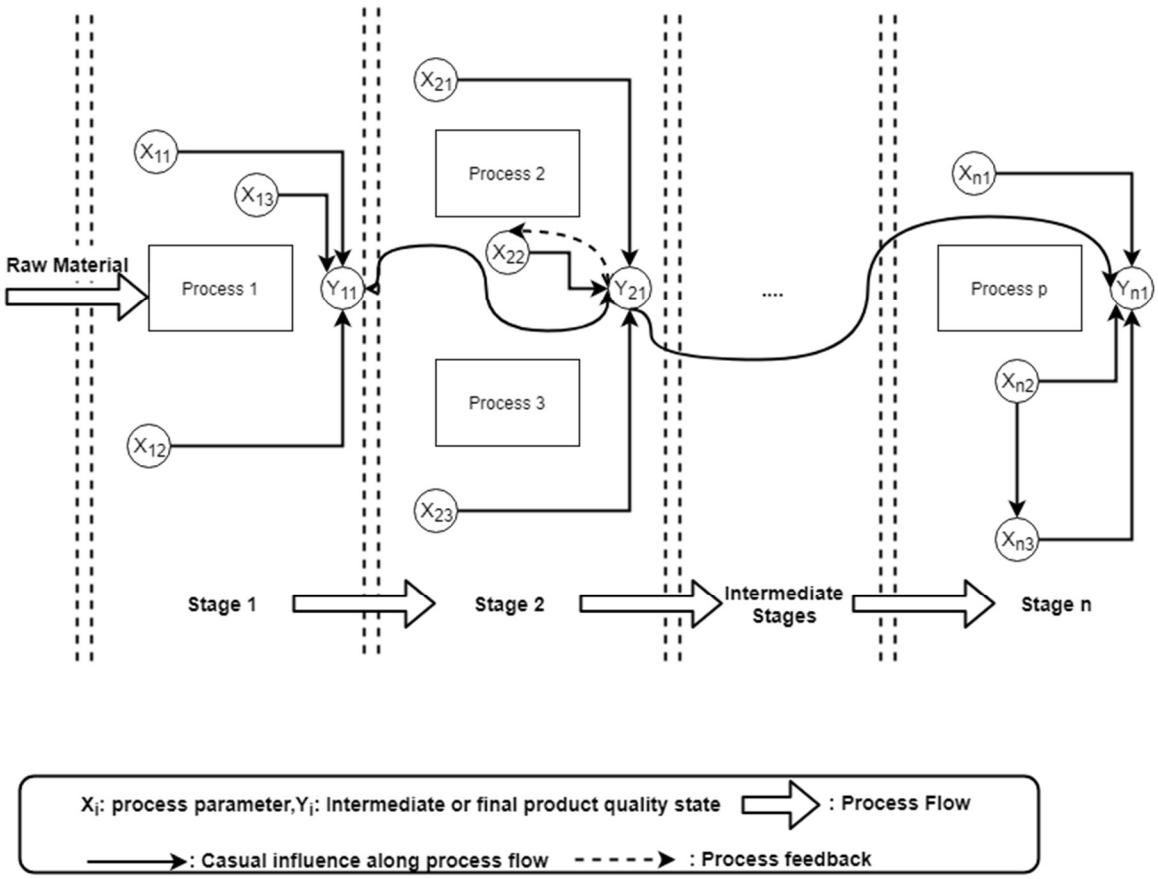

Figure 1- A diagram of the multistage system

Manufacturers have been using traditional control charts to monitor their product quality since the 1920s. Shewhart [3] converted a series of hypothesis testings into a graphic monitoring tool. Traditional statistical process control or monitoring (SPC or SPM) approaches are widely

used because of their simplicity and applicability. However, in the area of high-tech manufacturing products, traditional methods of quality control are not effective due to the "curse of dimensionality" [8]. Unlike the traditional methods where measurement is restricted to physical products or work in progress, process parameters offer ample opportunities for process monitoring and defect prevention. Since the number of parameters is usually very large, a high-dimensional problem often renders traditional control charts ineffective. For example, the production of a CPU includes hundreds of processes and thousands of process parameters. To study multiple quality characteristics, multiple techniques such as Hotelling T^2 [5], PCA, and GLRT are proposed [6]. Hotelling T^2 chart developed in 1947 are used where a $p \times 1$ sample vector with mean μ and covariance matrix Σ are known or can be estimated.

$$X^2 = (x - \mu)'\Sigma^{-1}(x - \mu) = \chi_p^2 \qquad \textbf{Equation 1}$$

A constant c can be determined according to the desired type I and type II error to define the boundaries of the normal process when $X^2 < c$ the process of interest is in control. PCA is often used to reduce the dimension of the sample vector and then, a monitoring technique is applied to the reduced dimension. Finally, GLRT is another method to detect the changes in multivariate problems. It also can be used to incorporate time information into the change detection models. Suppose X_t as a p-dimensioanl sample obtained at time unit t for a process. Two following hypotheses are testing the process to detect the change that happens in time τ.

$$H_0: X_1, X_2, \dots, X_t \sim f_0(x) \qquad \textbf{Equation 2}$$

$$H_1: X_1, X_2, \dots, X_\tau \sim f_0(x), \qquad X_{\tau+1}, X_{\tau+2}, \dots, X_t \sim f_1(x) \qquad \textbf{Equation 3}$$

The likelihood ratio is defined as

$$L = \frac{\prod_{i=1}^{\tau} \boldsymbol{f_0}(\boldsymbol{x_i}) \prod_{i=\tau+1}^{t} \boldsymbol{f_1}(\boldsymbol{x_i})}{\prod_{i=1}^{t} \boldsymbol{f_0}(\boldsymbol{x_i})} = \frac{\prod_{i=\tau+1}^{t} \boldsymbol{f_1}(\boldsymbol{x_i})}{\prod_{i=\tau+1}^{t} \boldsymbol{f_0}(\boldsymbol{x_i})} \qquad \textbf{Equation 4}$$

where f_0, f_1 are identified as unknown probability densities for the in-control (IC) and out-of-control (OC) process points. To find out the unknown τ, the generalized ratio can be defined to maximize the likelihood ratio in Equation 4. The log of the generalized likelihood ratio is

$$L_t = \max_{\tau} \sum_{i=\tau+1}^{t} ln \frac{f_1(x_i)}{f_0(x_i)} \quad \textbf{Equation 5}$$

The signal is triggered when the decision parameter L_t exceeds a certain limit. This signal indicates that a change has taken place.

Despite the effectiveness of these methods, however, the traditional multivariable methods cannot be effectively applied in complex processes because they were designed to detect mean shifts of a moderate number of quality characteristics usually less than 10 [6]. Another major SPC innovation since its inception was to detect the small process changes faster. Univariate control charts for this purpose include cumulative sum (CUSUM) and exponentially weighted moving average (EWMA) control charts. The main concept is to involve historical observations leading up to the current observation to expedite mean shifts or variance changes. These univariate control charts like the X-bar and R charts cannot be implemented effectively in cases where multiple quality characteristics or process parameters exist. In the multivariate environment, multivariate exponentially weighted moving averages (MEWMA) [9] and multivariate cumulative sum (MCUSUM) [10] control charts are appropriate. However, these methods either cannot detect off-target (*AKA* OC) signals fast enough or cause unacceptable false alarm rates as the number of variables increases [6]. Therefore, more efficient models are needed to tackle high dimensional process monitoring problems. Also, most SPC methods do not perform well in multistage applications since data from a multistage process is often considered as a whole without timestamps and, therefore, traditional multivariate SPC methods could not discriminate which

stage where a change occurs [2]. Hence, the need for in-process monitoring for multistage systems cannot be greater in the context of SPC.

Jin and Shi [11] first considered the modeling of a general multistage system where the key quality characteristics of the product at stage k is represented by $x_k$as the following equation:

$$x_k = A_{k-1}x_{k-1} + B_k u_k + w_k \text{ and } y_k = C_k x_k + v_k \quad \textbf{Equation 6}$$

where $u_k, w_k,$ and v_k represent process error source, unmodeled error, and sensor noise respectively. $A_{k-1}x_{k-1}$ represents the transformation of product quality deviations from station $k-1$ to station k, $B_k u_k$ represents the product deviations resulting from process errors at stage k and C_k maps the product quality states to quality measurements. The model has been used in many applications such as rigid-part assembly processes, compliant-part assembly processes, machining processes, and sheet stretch forming processes [2]. However, the physics of the process needs to be thoroughly studied to construct the process model.

Cause-selecting charts are other tools for monitoring the quality of the process in multistage systems. These charts have shown to be effective in finding the responsible stage in a faulty condition. Cause-selecting charts generally use univariate techniques, hence, cannot handle high dimensional problems [11]. Another commonly used technique in a special multistage system called multistream production is the group charts which can be used to detect quality changes in identical, individual streams. However, this technique cannot perform diagnosis within stages because it only tracks the worst performance in a stage to analyze the process [12].

Data-driven approaches emerged as promising framework classes for SPC [6]. These approaches include the use of classification-based models to group historical data into two: IC or OC. After learning from known patterns, trained models can predict the class IC or OC of a new

dataset. Several studies have been proposed to monitor the process in the multivariate environment under two main categories [6].

The first category utilized a method called the artificial contrast method [13-18] which artificially produced OC points to balance the sample set. Tuv and Runger [13] first introduced the artificial contrast data to represent the OC points. The artificially generated data were random numbers generated by a uniform distribution. In this study, the range of IC points has been used to generate random numbers. Then, the generation of contrast data has been repeated for each parameter independently. A gradient boosting machine has been used in this study to classify the IC and OC points. In addition, in case of high dimensionality, the authors recommended the reduction of the number of parameters using by a feature selection classifier.

Hwang, et. al., [18] followed the previous work by Tuv and Runger [13] where the generation of artificial contrast data was limited to one standard deviation of the target point. In addition, the selected classifiers by Hwang, et. al., [18] include Random Forest (RF) [19] and regularized least square classifier (RLSC) [20].

Hu, *et al.,* [14] then introduced the concept of fine-tuning the artificial contrast data by incorporating prior knowledge of the manufacturing process [13]. While Tuv and Runger [13] used a uniform distribution to generated artificial contrast data, Hu, et. al., [14] instead generated the contrast data using the artificial contrast data in an intentional pre-defined direction. The tuned direction comes from the pre-knowledge of the process. The classifier used by this method is RF. Hu, *et. al.,* [14] claimed to obtain more precise results in terms of accuracy and false alarms.

Li, *et. al.,* [15] applied the artificial contrast concept in a change point detection problem where a vector of data points considered as features to capture the time when there was any change in the pattern. By using the likelihood ratio function (Equation. 5), Hu and Runger [16]

incorporated the time element in the artificial contrast concept. The probability of each class by RF in each time unit is used to represent the functions in the likelihood ratio. Then, the EWMA chart using the obtained likelihood ratio is used to monitor the process.

Using the real-time data and the artificial contrast concept, Deng, *et. al.*, [17] introduced the real-time contrast concept to monitor a process. This proposed study used fixed-size new real-time observations to contrast the reference data (training set). By having a new observation window, a new classifier is trained for process monitoring. Like the traditional process monitoring studies, Deng, *et al.* [17] did not alter the reference data (points labeled y =0) while the new observations were defined as OC points. Hence, in normal process condition, the error of the proposed method is expected to be high as both IC and OC points are following the same pattern. Once a shift in the process occurs, the error reduces. To identify the important features, Deng, et. al., [17] used RF as the classifier.

The second category applies feature selection methods [21-24] to reduce the dimensionality of high-dimensional process monitoring problems. The developed T^2 statistic by Jian and Tsui [21] enables the identification of the responsible variables for OC points.

VS-MSPC is a variable selection based multivariate SPC control chart developed by Wang and Jiang [22]. The variable selection in the VS-MSPC is based on a penalized likelihood function. Then, the VS-MSPC method only monitors the selected variables. Wang and Jiang [22] assumed that the simultaneous shift in multivariate problems usually happens in a limited number of variables. Based on the assumption, monitoring a small fraction of variables is then possible by the multivariate SPC methods. One of the limitations of the proposed method, however, is not being sensitive to small shifts.

Zou and Qiu [23] used Lasso [25] as the variable selection based model. Then, the MEWMA chart is proposed as the monitoring tool in the reduced problem. Like the previous studies, Zou and Qiu's approach [23] comes with strong assumptions such as a limited number of variables to shift and normal and independent observations.

To incorporate cascade information, Jin, *et al.* [24] extended the previous work by Zou and Qiu [23]. In the cascade process, a process leads to a number of succeeding processes. Hence, when a shift in the process occurs, besides the root cause parameter, the subsequent process parameters might be considered as responsible elements as well where this might not be always the case. Jin, et. al., [24] incorporated the cascade information using a Bayesian Network. Hence, the proposed method is called Lasso-BN where BN stands for Bayesian Network. After identifying the truly responsible variables, a T^2 chart is used to monitor the process. The availability of the cascade relationship between parameters is assumed to be available and represented by a BN.

The above-mentioned studies generally follow the traditional SPC approaches where only product quality characteristics are considered in process monitoring and quality assessment. Using data-driven techniques, Wuest, *et al.,* [26] incorporated both product state and process state data for quality monitoring. The proposed study benefits from a wide range of supervised and unsupervised machine learning techniques. In the multistage production system illustrated by Wuest, *et al.,* [26], changes in the product physical shape are defined as the checkpoints.

Table 1- A literature review of machine learning approaches for process monitoring[6].

Authors	Approach	Method used	Assumption	Application
Tuv and Runger [13]	Artificial Contrast Data	Gradient boosting machine	Uniform contrast data by 3 standard deviations from the reference data. Enough amount of data is available to train the model. Easy to implement.	General high-dimensional problem
Hwang, *et. al.,* [18]	Artificial Contrast Data	RF, RLSC	Uniform contrast data by one standard deviation from the reference data. Enough amount of data is available to train the model. Easy to implement.	General high-dimensional problem
Hu, *et. al.,* [14]	Artificial Contrast Data	RF	Prior knowledge of the process is available. Enough amount of data is available to train the model. Easy to implement	Prior knowledge is available
Hu and Runger [16]	Artificial Contrast Data	RF	Prior knowledge of the process is available. Enough amount of data is available to train the model. Moderate to implement	Time-based monitoring
Deng, *et. al.,* [17]	Artificial Contrast Data	RF	Prior knowledge of the process is available. Enough amount of data is available to train the model. Hard to implement	Real-data is used to contrast the reference data

Table 1-continued- A literature review of machine learning approaches for process monitoring [6].

Authors	Approach	method used	Assumption	Application
Jian and Tsui [21]	Dimension Reduction	Extended T^2 statistic	Independent, normal data. Complex implementation	General high-dimensional problem
Wang and Jiang [22]	Dimension Reduction	VS-MSPC	Independent, normal data. Complex implementation	General high-dimensional problem
Zou and Qiu [23]	Dimension Reduction	Lasso	Independent, normal data. Moderate to implement.	General high-dimensional problem
Jin, *et. al.,* [24]	Dimension Reduction	Lasso, Bayesian Network	The cascade relation between elements is available and can be modeled by a Bayesian network. Independent, normal data. Complex implementation.	Process monitoring of cascade processes.
Wuest, *et. al.,* [26]	Process monitoring using product/process data	Support vector machine, Agglomerative hierarchical clustering	The relation between the processes could be very complex which in this study a simple process is illustrated.	Process monitoring using product/process states.
Uhlmann *et al.* [27]	Pattern recognition in the 3D printing process	Support vector machine, Neural networks, Bayesian classifier, Nearest neighbors	A limited number of variables are selected (17 parameters). A predefined category is assumed for product quality. The number of samples is limited; hence the accuracy of the model is not satisfactory.	Process monitoring in selective laser melting process
Kao *et al.* [28]	Semiconductor manufacturing	Decision tree, Support vector machine, Naïve Bayes	Majority of the features have been dropped (550 out of 590). Stages have been combined and only one model has been trained on all feature set.	Process monitoring in semiconductor manufacturing systems

Data-driven techniques have also been used by Uhlmann *et al.* [27] to perform quality monitoring in a metal 3D printing process called selective laser melting. Several machine learning models such as support vector machine, neural networks, Bayesian classifier, and nearest neighbors were applied to perform the monitoring task. Kao *et al.* [28] also applied. several machine learning techniques on a semiconductor manufacturing dataset called SECOM (SEmiCOnductor Manufacturing) [1]. Table 1 provides a summary of the above-mentioned studies.

Despite all these improvements in process monitoring of multistage systems, there are still ample opportunities and challenges remained including problem dimension, new unseen fault behaviors, and unbalanced classes of training data. These challenges are discussed by details in the next section.

1.2. Challenges

In developing classification-based process monitoring techniques for high dimensional multistage systems, researchers usually face three main challenges: high dimensionality, updating process, and rare OC points as discussed in the following subsections.

1.2.1. High Dimensionality

Today's production machines are often equipped with multiple sensors generating a large amount of process data at a torrential pace. The widening use of the internet of things (IoT) has contributed to this trend [29]. Machine learning (ML) techniques, a subset of "artificial intelligence," has also contributed to the possibility of solving problems with high dimensionality because algorithms are used to autonomously learn from data [30]. The production machines usually generate a huge amount of data with high dimension. As discussed earlier, coordinate measurement machines produce great measurements in a high dimension. Dealing with high

dimensional data usually asks for more efforts and computations. Besides the computation time, data-driven models with high dimensions tend to overfit. Overfitting is a very common problem, especially in predictive models where they perform very well in the training sets. However, they fail to generalize meaning that they cannot predict unseen datasets well [19]. Hence, researchers usually avoid overfitting by reducing the dimension of problems. Dropping correlated features, dimension reduction techniques such as PCA, linear discrimination analysis (LDA), and penalized learners such as Lasso [25] and Ridge Regression are examples of the dimension-reduction techniques to avoid overfitting.

1.2.2. Updating Process (Cover Unseen Data Patterns)

The training phase is a crucial part that makes a data-driven based model more accurate. In traditional SPC practice, the training data set usually does not change. This practice is often referred to as the Phase I SPC in which data from a processing period deemed in control constitutes a training set. However, most data-driven approaches cannot perform accurately facing unseen patterns. Hence, the training phase must be updated periodically to enable the model to cover new patterns which should include both IC and OC patterns. In general ML models benefit from more training samples for better accuracy.

1.2.3. Rare OC points (Unbalance classes)

In traditional manufacturing processes, the reference data consisted of only IC points and the process was monitored against the reference data to pinpoint faulty spots. However, data-driven approaches such as classification algorithms require both IC and OC observations in the training set. However, a healthy manufacturing process contains occasional OC conditions. Hence, a historical data set consists of much more IC than OC data. This phenomenon causes the unbalance classification issue, especially in binary classification problems. Two approaches including

undersampling and oversampling have been proposed in the literature [31] to tackle this issue. In the undersampling method, the number of training data from the majority class is reduced to the level of a minority class. On the other hand, the oversampling method generates more samples from the minority class to those of the majority class. Artificial contrast data is one of the oversampling methods applied to the process monitoring studies [13]. Despite the existence of several studies that use the artificial contrast data in process monitoring problems, a comprehensive study exploring several machine learning approaches facing unbalance samples is lacking. It is not certain whether the undersampling approach is better than the oversampling methods to tackle unbalance sampling problems in the content of process monitoring. Hence, both approaches should be investigated.

1.3. The Proposed Data-Driven Multistage Process Monitoring Model

In this part, a short summary of the proposed framework is presented. The first step of the proposed framework is the organization of the collected data for each manufacturing stage. Then at each stage, a clustering method (K-means) reduces readings of each parameter into a few discrete categories. Specifically, the parameters data in each production stage is reduced to an assigned cluster to reduce dimension. In other words, each cluster represents a production recipe in a production stage. In the next step, a classification algorithm performs feature selection to further reduce the dimension by selecting only significant stages in predicting the final quality status, i.e. either a good product or bad product. Multiple classification models then are built upon the significant stages to perform process monitoring. The most crucial outcome of this proposed framework is the identification of significant process parameters at critical stages potentially affecting the final production quality. This knowledge would enable a viable process control strategy to be developed. The developed model has been successfully implemented in two cases:

metal additive manufacturing (AM) and semiconductor manufacturing. The cases where the proposed framework has been implemented are different in nature. Therefore, although both models share the same basic framework, there are major differences in term of implementation strategies. A summary of two case studies will be provided in the next session.

1.4. Applied Industries

1.4.1. Case 1- Metal AM

We have applied a multi-layer classification process monitoring model (MLCPM) [32] to the metal 3D printing industry, which a multistage process considering its layer-by-layer nature of production. We adopt supervised and unsupervised models in MLCPM to control the quality of the print process before the print process reaches its final layer. MLCPM provides solutions toward the high dimensionality of metal 3D print data using clustering and feature selection methods. MLCPM model will be discussed in detail in Chapter 2.

1.4.2. Case 2- Semiconductor Manufacturing

In this research, we extended the previous study [32] by addressing multiple challenges such as imbalanced classes, high dimensionality, and covering unseen patterns. A semiconductor manufacturing repository dataset is used to demonstrate how the proposed framework can be implemented. This framework will be discussed in detail in Chapter 3.

1.5. Conclusion

In this chapter, we outline the need for new studies toward process monitoring for complex multistage production systems. This chapter explained why the traditional process monitoring studies fail in complex high dimensional systems. In addition, we summarized several challenges for implementing data-driven process monitoring methods. Then, the proposed process monitoring framework applied to two applications were briefly explained.

This study contains the following chapters. Chapter 2 provides detailed information regarding the MLCPM model applied in the laser-powered, power-bed, 3D metal AM processes. Then, in Chapter 3, the proposed framework will be applied to a semiconductor manufacturing industry. Finally, future studies and conclusions are presented in Chapter 4.

Chapter 2 - MLCPM: A Process Monitoring Framework for 3D Metal Printing in Industrial Scale

***Chapter 2**, in full, is a reprint of the material as it appears in Computers & Industrial Engineering, 2018, Mohammadhossein Amini and Shing. I. Chang. doi: 10.1016/j.cie.2018.07.041*

Abstract

Metal 3D printing is one of the fastest growing additive manufacturing (AM) technologies in recent years. Despite such improvements in its technical capabilities, reliable metal printing is still not well understood. One of the barriers of industrialization of metal AM is process monitoring and quality assurance of the printed product. These barriers are especially much highlighted in aerospace and medical device manufacturing industries where the highly reliable and quality products are needed. Selective Laser Melting (SLM) is one of the main metal 3D printing methods where more than 50 parameters may affect the quality of the print. However, current SLM printing processes only utilize a fraction of the collected data for quality related tasks. This study proposes a process monitoring framework named MLCPM (Multi-Layer Classifier for Process Monitoring) to predict the likelihood of successful printing at critical printing stages based on collective data provided by identical 3D printing machines producing the same part. The proposed framework provides a blueprint for control strategies during a printing process and aims to prevent defects using data-driven techniques. A numerical study using simulated data is provided to demonstrate how the proposed method can be implemented.

2.1. Introduction

Complex and flexible production technologies such as additive manufacturing (AM) widely known as 3D printing [33] is increasingly in demand for building sophisticated products. The term "AM" defines the production process by adding material layer by layer rather than

removing material from a block. Metal printing is one of the applications of AM that has been widely studied in recent years. The overall AM has seen 34.9% growth while the metal AM segment has experienced growth of over 75% in 2013 [34]. Industries such as aviation, healthcare, and automotive that use complex metallic parts have contributed to this growth of metal AM. Although there are various categories of metal printing, powder bed printing is one of the most prominent types widely used in the industry. Under the category of powder bed printing, Selective Laser Melting (SLM) is the most promising method that has drawn much attention in recent years [35]. The focus of this research is mainly on the SLM method application in the industrial scale [36]. By industrial scale, we mean that multiple identical 3D printing machines scattered in different 3D printing farms.

Even though AM technologies are improving, there are still several main challenges including process reliability and quality assurance of the process and finished product [37]. Quality assurance and reproducibility are two key factors to bring AM into the industrial scale. The first challenge is the amount of sensor data collected during a SLM printing process. For example, the coaxial visible-wavelength camera in Concept Laser's machines is capable of capturing pictures at the rate of 4000 frames per second [37]. At this rate, the size of the data is huge which is cumbersome for the traditional relational databases to handle. Some studies have provided closed-loop feedback control systems to adjust the parameters affecting the quality of the final product [38]. However, the limited knowledge of significant parameters leads to inefficient monitoring models [35]. In addition, more than 50 process parameters have impacts on print quality but only a small set of these variables are actively used for process control purposes [35]. Existing process monitoring and control methods often based on the first principle focus on single SLM machine. These methods ignore the possibility of process data generated from other SLM machines. In the

situation where the same part is printed in multiple SLM machines in various printing farms, it provides the opportunity for a data-driven approach to learn the relationship between print quality and process parameters. A system-wide monitoring framework may provide a way to ensure printing reliability and reproductivity by pooling production and quality data from various printers.

This study proposes a system-wide process monitoring system using data from all machines printing the same part to predict printing quality at critical layers. The proposed method is called MLCPM which stands for Multi-Layer Classifier for Process Monitoring. We assume process and quality data from multiple printers located in various places are shared through secured servers in the cloud. All analyses take place in the cloud so that big data analytics is possible [39]. Unlike the current techniques in the metal AM process monitoring, MLCPM is capable of monitoring all measurable process parameters in the printing process. The predicted quality at certain critical layers provides operators a chance to change process settings before a part is fully printed. The proposed MLCPM overcomes high-dimension issues by adopting multiple techniques such as clustering and feature selection. Equipped with various predictive models, MLCPM is designed to predict defective possibilities while a part is still printing.

This chapter contains the following sections. The next section provides a brief literature review of metal 3D printing methods and studies related to quality monitoring. Then MLCPM is introduced for process monitoring. A simulated numerical example is further provided to show the operation of the proposed MLCPM. Finally, future studies and conclusions are presented in the last section.

2.2. BACKGROUND

2.2.1. Additive Manufacturing and Metal Additive Manufacturing

AM has provided the ability to make complex parts such as honeycomb structures, and intricate internal features right from a 3D CAD model [40]. Due to its capability of manufacturing complex parts and easy and fast process, many industries are moving toward it. 3D printers have built plane engine parts, medical parts, houses, toys, and even the development of printing human organs are on the horizon. 3D printers are capable of printing polymer or metallic parts. Polymer printing has been around for 40 years while the metal printing has been emerged around 20 years [41]. Since the industry has shown huge interests toward the metal segment of AM [34], the focus of this paper is on the metal 3D printing.

American Society for Testing and Materials (ASTM) technical committee F42 [42] has provided a list of AM terminologies where five out of seven categories are capable of printing the metallic material [36]. The metal printing capable methods include the following main categories: directed energy deposition (DED), powder bed fusion (PBF), material jetting, binder jetting, and sheet lamination processes. Among these methods, PBF and DED are the two major technologies contributing to the AM industrial revolution. In DED, the part is shaped by melted material as the metal is deposited. Metal DED is similar to the other deposition methods involving polymer and ceramic. PBF techniques use either a laser or electron beam to melt and fuse the metal powder. In this paper, we focus on the SLM method that is a technique under the PBF category. SLM uses laser power to melt the metal powder by selectively melting a thin layer of the powder to shape the part. The powder is evenly distributed on the surface layer by a roller. Figure 2 shows the mechanism of an SLM process.

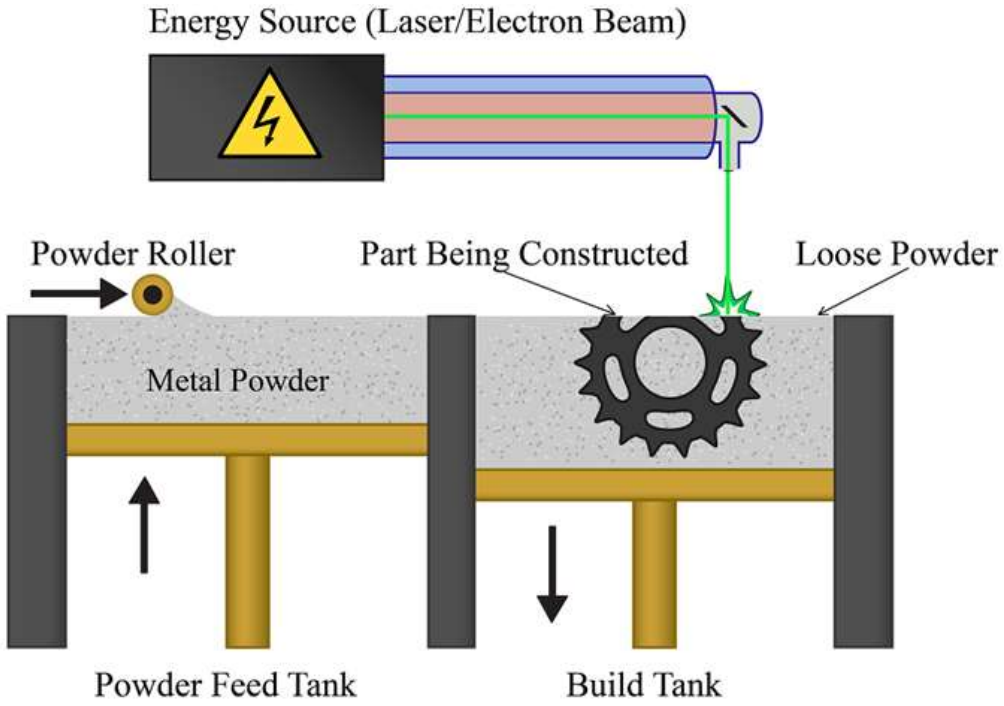

Figure 2-Mechanism of SLM process [43]

2.2.2. Process Monitoring in AM

Process monitoring techniques should be applied to ensure quality production. Due to the complex nature of the SLM, the process monitoring, and controlling is very difficult [44, 45]. In some mission-critical industries, destructive inspections are used to check the quality of the final product. To overcome this issue, several studies have been conducted to assure the quality of the final product [27, 35, 36, 38, 46-56]. The first step of monitoring a printing process is to identify the variables affecting the quality of the printed products. Leaving these variables uncontrolled will generate defects such as balling effects or cracks. Variables affecting printing quality includes powder, laser beam, printing chamber environment, and so on. Figure 3 illustrates some of these parameters in a diagram. Variables are further categorized into predefined variables and controllable variables which can be used to adjust a process in closed-loop feedback control systems [51]. In the process of SLM, more than 50 variables may affect printing quality, which

makes the understanding of the process physics and process monitoring more challenging [35]. Understanding these variables will drastically help control the process.

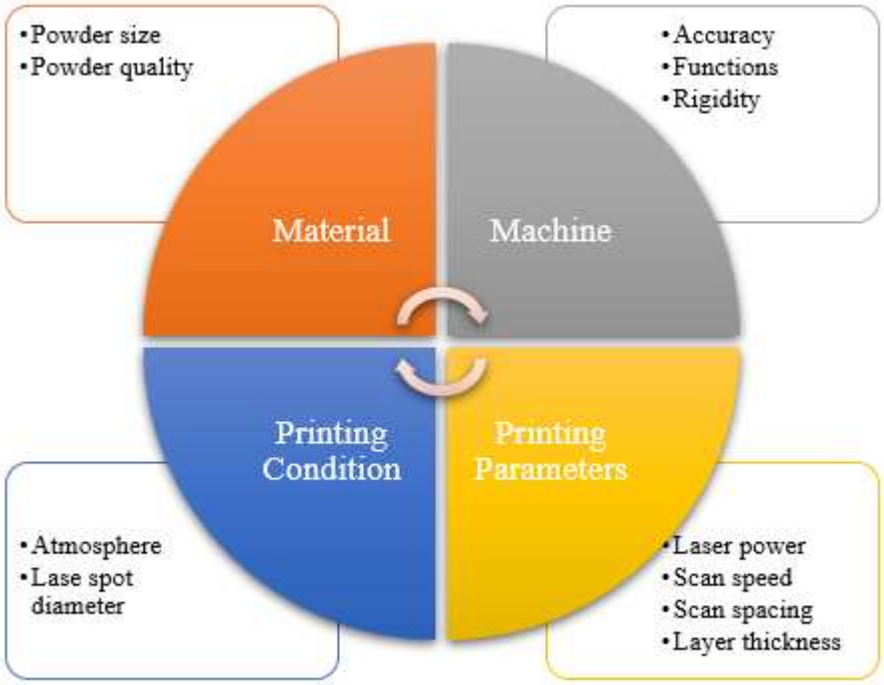

Figure 3- Diagram of affecting parameters in the SLM process

Malekpour and El-Mounayri [51] provided a list of variables affecting SLM printing quality. Their studies in the context of process control in metal 3D printing mainly focused on process signatures such as the characteristics of the melt-pool, which is believed to be the most influential to the quality of the printed product [38]. Based on the laser power and scan speed, the melt-pool grows or shrinks. Figure 4 shows a melt pool on a workpiece under a laser beam.

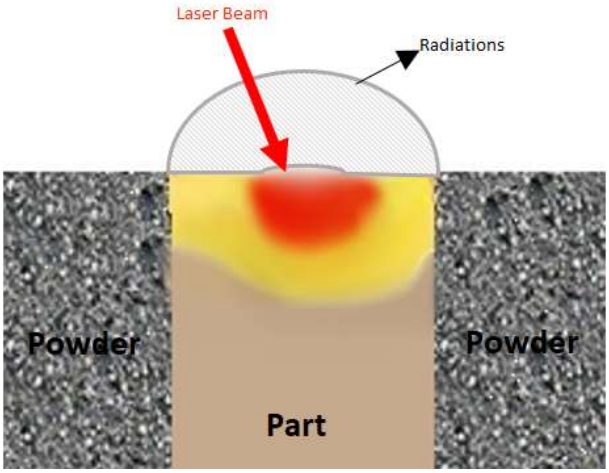

Figure 4- Melt-pool and emitted process signatures under the laser beam

Two main categories of monitoring a laser printing process involve acoustic and optic methods where the latter is more popular [46]. In acoustic methods, surface-contact and non-contact sensors are used to convert sound waves into electrical outputs [46]. Optic methods contain two approaches: Lagrangian and Eulerian. In the Lagrangian method, the camera moves with the laser beam to follow the melting area wherein the Eulerian method the sensor focuses only on a fixed position. The Lagrangian method provides the opportunity to follow and control the melt-pool area but it lacks the record of a historical thermal characteristic at the assessment point. On the contrary, the historical thermal data of points can be achieved in the Eulerian method, but the melt-pool analysis cannot be accomplished. Many studies have focused on the shape and temperature distribution of the melt-pool. Kruth and Mercelis [48] patented a feedback control system where temperature distribution in melt-pool was used to adjust the laser power. In the setup, a photodiode was installed to record the light intensity reflected by the melt-pool. Yadroitsev *et al.* [49] introduced a temperature monitoring system using a charge-coupled device (CCD) camera. To monitor the temperature distribution, a camera was coaxially aligned to the laser beam. The focus of the study was to monitor and detect built-in microstructures during a heating process. Imani *et al.,* [52] proposed a layer-based monitoring based on X-ray computed tomography (XCT)

images of titanium alloy (Ti-6Al-4V) printed cylinder. Yao *et al.* [53] utilized the image profiles acquired in a powder bed fusion process and studied the fractal patterns for the purpose of process monitoring, quality assessment, and control. Chivel and Smurov [50] proposed the closed loop control system for selective laser sintering (SLS) systems where the goal was to monitor the temperature distribution and record the maximum surface temperature. Infrared (IR) cameras are used widely as well to monitor the temperature distribution. Chivel and Smurov [50] applied IR cameras to monitor a temperature distribution over an entire workpiece as oppose to photodiodes which were designed to focus on a single spot only. In the monitoring process, the location of the camera is extremely challenging especially in electron beam melting (EBM) printing due to the lack of space in the housing for electron beam gun in EBM process [38]. While most of the studies have focused on melt-pool effects on print quality, studies such as Montazeri *et al*., [54], Imani *et al*., [52], Morsali *et al.* [55], Montazeri and Rao [56], and Malekipour and El-Mounayri [51] have focused on other affecting parameters such as material quality, nozzle diameter, and hatching space.

While most of the studies rely only on the melt-pool area to control the process, Uhlmann *et al*. [27] proposed a machine learning (ML) algorithm considering 16 different variables such as platform temperature, process oxygen, and process chamber temperature. This study aimed to provide a pattern recognition tool by using ML algorithms. In addition, the authors provided a variable selection tool to identify the most effective parameters. A condition monitoring tool is used for analysis. First, three different states for the final product were defined as finished perfectly, finished with errors, and not finished. Using a section of the dataset (totally 271 samples), they trained the proposed classifier and obtained its accuracy. In general, an accurate model should not provide too many false positives or false negatives. The highest accuracy

obtained by Uhlmann *et al*. [27] however is less than 60%. ML methods heavily rely on the given samples. In general, the larger the dataset containing all the categories (product states), the more prediction accuracy can be achieved. Our proposed study leverages a networked 3D printing machines and the data generated to provide amble training data set for better model accuracy.

In terms of industry's effort toward quality metal printing, Simufact [57] released a simulation software which could predict the failures of the printed job based on the given parameters. In addition, ESI also provided simulation tools for AM industries. ESI's platform is called unified integrated computational material engineering (ICME) that provides tools to predict workpiece behavior during the printing process. These tools focus on powder interaction and thermal distribution issues to eliminate potential defects [58]. Other developments in the metal printing industry are mostly led by aerospace and healthcare industries [59].

Despite these improvements in process monitoring in the AM industry, current methods use local data and provide a closed-loop feedback system to adjust a small portion of independent variables such as laser power in one AM machine. To improve prediction accuracy, more samples and data are needed to train a predictive model. It may involve other patterns that may have been faced in different machines. We propose a data-driven framework called MLCPM that utilizes data from multiple printing machines and machine learning technologies to identify opportunities for process control during a printing process. Details of the proposed MLCPM framework is described in the next section.

2.3. MLCPM: The Proposed Data-Driven Approach for Metal 3D Printing Process Monitoring

The nature of the laser melting process makes the understanding of the physics of printing process challenging. In addition, a large number of affecting parameters make the process

monitoring more complex. Current studies usually focus on a certain area (melt-pool) which is deemed the most influence on the quality of the finished part. Further, ML models such as the proposed model by Uhlmann *et al*. [27] have shown to be promising in 3D print process monitoring. ML models have been widely used in the industry and academic studies. ML can be categorized into supervised and unsupervised methods. The main difference between them is that supervised learning methods need labeled targets whereas unsupervised models do not require labeling. One of the most important features of ML methods is relaxing the assumption of known data distributions. Regression and classification models are the most well-known supervised models while clustering is the main technique of the unsupervised category. Regression and classification models have been used for prediction based on a set of independent parameters. Uhlmann *et al*. [27] used four powerful classification methods as support vector machine, neural networks, Bayesian classifier, and nearest neighbors toward 3D printing problems. However, one of the drawbacks of ML models is the demand for a large sample of data to learn patterns. Therefore, when a limited amount of data is given, the accuracy of the model decreases. This limitation causes the model developed by Uhlmann *et al.* [27] to be less satisfactory as its model accuracy is less than 60%. In addition, the proposed method by Uhlmann *et al*. [27] does not perform the diagnosis and adjustment task. The main purposes of the method were to evaluate the pattern recognition and parameter selection tasks. Further, it applies locally on a single printing machine.

In this study, we propose a system-wide process monitoring method that incorporates processing data from all printing machines in a printing farm. The proposed method monitors process parameter readings layer by layer and provides warning signals when a defective pattern emerges before a printing part is fully printed. In the proposed method, we assume that data comes

from multiple machines but prints the same job, Analyses of the proposed model are performed in the cloud-based servers. Therefore, enough amount of samples are available to apply ML techniques.

This section provides a proposed framework to tackle the process monitoring task in SLM process using ML techniques to provide an adequate amount of samples for training. MLCPM consists of several clustering and classification models to analyze the data during the printing process. The inclusion of a large amount of data is designed to increase prediction accuracy.

In metal printing methods such as SLM, ML techniques could effectively be applied to predict the products state in term of printing quality. To tackle the complexity of layer-by-layer printing process involving data with large dimensions, we propose a multi-layer model called MLCPM using multiple clustering, classification, and prediction models. MLCPM is built and performed in two separate phases. Phase I is the training phase while phase II is the monitoring phase.

Phase I – Data Pooling and Model Building

Assume the training data set is composed of n printed parts collected from identical printing machines available in the farm where each print is called a job. This assumption helps to train the model with multiple patterns identified on as many as possible machines. Next, job quality and the process data generated from all printing layers are collected. Job quality is based on a final inspection of printing quality at the end of printing. A binary decision should be made regarding job quality: good (0) or defective (1), which is denoted as a target value (Y). The printing process for each job is composed of M layers of printing. During each layer of printing, production data from k process parameters are collected as shown in Table 1. Printing parameters are either controllable or pre-defined where controllable parameters are used for adjustment [35]. Usually,

the speed of data collection is high and therefore the amount of data collected is tremendous. This application is a typical big data problem often encountered in modern manufacturing facilities.

For each job, a matrix listed in Table 1 is formed by the collected data from various sensors where X_{ijk} represents the collected data for job i and parameter k in layer j. In Table 2, rows represent the printing layers for each job i and columns represent the data for each measurement by a parameter. However, the last column represents the target value for job i. After collecting data for all printing parts, a matrix for each printing layer is formed. The new matrix contains data from all jobs within a layer. Table 2 illustrates the layers matrix.

Table 2- Data collected for job i

Layer	Affecting Parameters				Target value (Y)
	Parameter 1	Parameter 2	…	Parameter k	
Layer 1	X_{i11}	X_{i12}	…	X_{i1k}	
Layer 2	X_{i21}	X_{i22}	…	X_{i2k}	Y_i
…	…	…	…	…	
Layer M	X_{iM1}	X_{iM2}	…	X_{iMk}	

An unsupervised clustering algorithm is then applied to each layer of data shown in Table 3. These clusters help identify hidden patterns inside the collected data. The resulted cluster is a representation or snapshot of this printed layer. The clustering algorithm used in this study is the embedded K-means method in Scikit-Learn [60] package available for Python 2.7. In addition, an "elbow method" [61, 62] is used to determine the efficient number of clusters.

Table 3- Data collected for layer *i*

Job	Affecting Parameters			
	Parameter 1	Parameter 2	…	Parameter k
Job 1	X_{1i1}	X_{1i2}	…	X_{1ik}
Job 2	X_{2i1}	X_{2i2}	…	X_{2ik}
…	…	…	…	…
Job n	X_{ni1}	X_{ni2}	…	X_{nik}

An elbow method uses the distortion within clusters to determine the efficient number of clusters. Generally, in clustering a dataset, the distortion of clusters is zero when each point is assigned to one cluster and is in maximum when all points are grouped in one cluster. The first cluster illustrates a lot of variation, but at some point, the marginal gain will drop, giving an angle in the graph. The number of clusters is chosen at this point, hence the "elbow criterion".

Several distance methods such as Euclidean, Manhattan, or Chebychev [63] can be applied in clustering task. MLCPM uses the Euclidean distance to measure the similarities between points. The K-means++ method embedded in Scikit-Learn [60] package was chosen for initial centroids in K-means clustering. K-means++ initializes the centroids to be distant from each other, leading to better results than random initialization [60].

The K-means algorithm assigns each printed part in a given layer to a cluster based on Equation 7. This process is repeated for all parts and all layers to form the classification matrix illustrated in Table 4. The outcome is that each printed part has one identified cluster for each layer. These assigned clusters then import into the classification models to be discussed.

$$C_{ij} = g(X_{ijk}), C_{ij} \in \left\{C_{1j}, C_{2j}, \dots, C_{p_j j}\right\} \quad \textbf{Equation 7}$$

where C_{ij} is the assigned cluster for part i within layer j and p_j is the efficient number of clusters for layer j obtained by the elbow method.

Table 4- Classification matrix

Job	Printing layers				Target value (Y)
	Layer 1	Layer 2	…	Layer M	
Job 1	C_{11}	C_{12}	…	C_{1M}	Y_1
Job 2	C_{21}	C_{22}	…	C_{2M}	Y_2
…	…	…	…	…	…
Job n	C_{n1}	C_{n2}	…	C_{nM}	Y_n

In Table 4, C_{ij} is the identified cluster for job i in layer j generated from the K-means algorithm and Y_i is the target value (e.g. 0 for good and 1 for defective) for job i. Using Table 4, additional classification models need to be built to monitor the printing process and predict the target value. In the proposed model, features of the classification model are the clusters of layers. We propose the use of Random Forest (RF) model because the number of layers is usually very high. In addition, RF provides the probability to the predicted classes. These probabilities can be used further by quality technicians to define the level of product quality. Random Forest model uses multiple decision trees to perform the classification model. Therefore, the prediction result is more accurate as more classifiers are generated [6]. RF provides the importance of features (i.e. layers) where a cut off number (significance value) can be chosen to select only the most important ones. For example, in a printing process containing 1000 layers of printing, RF can sort the importance of the layers regarding the print quality. Then, the limited number of those layers (for example, layer 20, layer 50, and layer 900) can be chosen to further predict the print quality. Next, three classification models will be generated where the first classifier includes layers (features) up to 20 but with layer 20 as the most important layer. The second classifier includes layers up to 50

but with the most significant layers 20 and 50. The last classifier includes all printer layers up to 900. In general, assuming *m* significant layers in *M* layers of printing, *m* classification models can be built as shown in the following set of equations:

$$\text{Prediction Models} \begin{cases} \text{Model 1: } \widehat{Y}_i = f_1(C_{i1})) \\ \text{Model 2: } \widehat{Y}_i = f_2(C_{i1}, C_{i2}) \\ \dots \\ \text{Model m: } \widehat{Y}_i = f_m(C_{i1}, C_{i2}, \dots, C_{im}) \end{cases} \quad \textbf{Equation 8}$$

where C_{ij} is the identified cluster for job i in layer j generated from the K-means algorithm and Y_i is the target value (e.g. 0 for good and 1 for defective) for job i. Note that not all layers are significant although they are considered in the classification models. For example, the classification models that significantly affect the outcome of Y are at layers 20, 50, and 900 (m=3) as follow:

$$\text{Prediction Models} \begin{cases} \text{Model 1: } \widehat{Y}_i = f_1(C_{i1}, C_{i2} \dots, C_{i20}) \\ \text{Model 2: } \widehat{Y}_i = f_2(C_{i1}, C_{i2} \dots, C_{i20}, \dots, C_{i50}) \\ \text{Model 3: } \widehat{Y}_i = f_2(C_{i1}, C_{i2} \dots, C_{i20}, \dots, C_{i50}, \dots, C_{i900}) \end{cases} \quad \textbf{Equation 9}$$

Phase II – Process Monitoring

When a new part is being printed, a K-means algorithm is used to assign each layer into clusters determined by the training data set. When the first significant layer (e.g. layer 20) is reached, the first classification model can then be used to predict the target value. If the prediction result is 0 (i.e. the good part), the process continues until it reaches the next significant layer (e.g. 50). If the prediction result is 1 (i.e. the defective part), then the process engineers should adjust parameters for the rest of the process. For example, the operator may adjust the controllable parameters such as the laser power in the consequent layers to reduce the chance of printing the

defective part. This is a warning to the process engineers to adjust parameters to prevent the possible deficiency in the printed part. The same procedure applies to the rest of the layers until the printing process finishes. The proposed framework may also provide a blueprint for a close-loop process control of SLM printing.

MLCPM can be summarized as follow:

Phase I:

1. Data collection. The data generated by sensors from multiple printers for all printing parts in each printing layer are collected (Table 2).
2. Clustering. The K-means is applied to the data collected in each printing layer (Table 3). A K-means algorithm generates clustering for each layer of printing. The number of clusters is determined by the elbow method [59, 60].
3. Classification matrix generation. A matrix consist of the assigned cluster for each layer of printing along with the target value (Y) is formed for all printed parts (Table 4). This matrix is the input for building predictive models.
4. Layer selection. RF is applied to the matrix generated in the previous step to find the most significant layers. Only the significant layers out of *M* layers will be used for further use.
5. Building the classifiers. Based on the significant layers found in the previous step, multiple prediction models using RF are built.

Phase II

6. Clustering the new data layer by layer. The trained K-means is used to assign clusters for printing layer of a new job.

7. Process monitoring. The classifiers built in step 5 are used to predict the target value for the assigned clusters in step 6 when a significant layer identified in Phase I is reached. If the prediction result is a good part ($Y = 0$), then the printing process continues, and a cluster is obtained for all layers up to the next significant layer. However, if the prediction result is a defective part ($Y = 1$), then engineers should adjust the process parameters in subsequent layers to avoid the defective part.

One of the advantages of MLPCM over current data driven process monitoring studies is multi-layer prediction. Built on multiple layer-based predictive models, MLCPM can predict the overall print quality in multiple stages. While the current data driven process monitoring techniques in AM, provide a single predictive model. Due to this limitation and high dimensionality of the problem, the applied feature selection techniques in current studies shrink the model by dropping good amount of data.

MLCPM includes several contributions to the study of process monitoring in 3D metal printing. The clustering task in MLCPM reduces a high-dimension ML problem dramatically. For example, consider a printing process with 1000 of layers and 50 parameters. The MLCPM clustering task reduces the matrix of 1000 x 50 to 1000 x 1 (98% reduction). In addition, the significant layer selection further reduces the computation by not focusing on other layers. It should be noted that in current literature, since only one predictive model is built feature selection models drop a significant amount of data. However, MLCPM uses feature selection to select checkpoints on significant layers to make efficient numbers of predictive models. As data is precious, MLCPM does not ignore any printing data. MLCPM predicts the defects during a printing process while the current quality-monitoring practice has to wait until a printing job is

finished. Printing farm operators can benefit using proposed MLCPM to prevent printing defective parts or prevent unnecessary printing when printing adjustments are not feasible.

The MLCPM framework, however, comes with a set of assumptions. First, we assume that samples are printed using identical printing machines in the farm. This assumption helps to populate the training set with patterns identified on as many as machines as available. Second, samples are using the same material and same design. Printing a single design means mass producing the same part. Third, MLCPM assumes that the data from the same parameters are collected from each printing machine. Fourth, the printing parameters can be controllable or pre-defined. But there should be at least one controllable parameter to be used for adjustment purpose. Fifth, the initial value for controllable parameters are set based on the pre-knowledge of the process. Finally, MLCPM assumes that once the printing job is finished, the data is instantly available on the cloud to be analyzed. A numerical example using the simulated values is provided in the following subsection to illustrate the use of the proposed framework.

2.3.1. A Numerical Example

In this section, a simulated case is used to demonstrate how the proposed MLCPM functions using the information gleaned from the literature [27, 57-59, 64]. The training data set contains n=1000 printed parts where 100 layers of printing are considered. The same procedure illustrated here can be extended for parts with much more layers. Four process parameters selected for demonstration are oxygen level, optical bank temperature, pump temperature, and laser power where the first three are non-controllable parameters but very crucial for print quality [27] while the last controllable parameter is used to adjust machines' parameters during printing.

Python 2.7 was used to code MLCPM. RF and the K-means embedded in Scikit-Learn [60] package are used to perform the classification and clustering tasks. A uniform random number

generator embedded in scipy package for python 2.7 [65] was used to generate simulated values for the first three parameters and random integer values [65] were assigned for laser power. The steps of the algorithm are implemented as follow:

Step 1: Collect the data for all layers of printing (M=100) for 1000 printing jobs from identical printing machines. The training data can be collected from a smaller number of printing machine, however, a higher number of machines can potentially cover more patterns result in more accurate predictive models. Table 5 illustrates the data for the first printing layer for all 1000 samples collected from multiple SLM printers.

Table 5- Data collected from the first layer of printing

Samples	Parameters			
	oxygen level	optical bank temperature	pump temperature	laser power
Sample 1	0.0153	996.62	4995.053	48
Sample 2	0.0083	993.98	5002.389	42
…	…	…	…	…
Sample 1000	0.0792	994.72	4998.753	46

Step 2: An elbow method is applied to the data of all 1000 samples/jobs from every printing layer to select the best number of clusters for each printing layer. After obtaining the number of clusters, a K-means algorithm is applied to each printing layer. The input to the K-means for each layer is the data collected for the specific layer (e.g. the first layer shown in Table 5) with the intended number of clusters (obtained by the elbow method). Figure 4 shows the elbow chart by applying the K-means algorithm to the first layer data in Table 3. Figure 5 shows that the selection of more than 2 clusters does not introduce that much distortion. Therefore, we chose two clusters for the first layer. Similarly, the numbers of clusters for the rest of the layers have been selected.

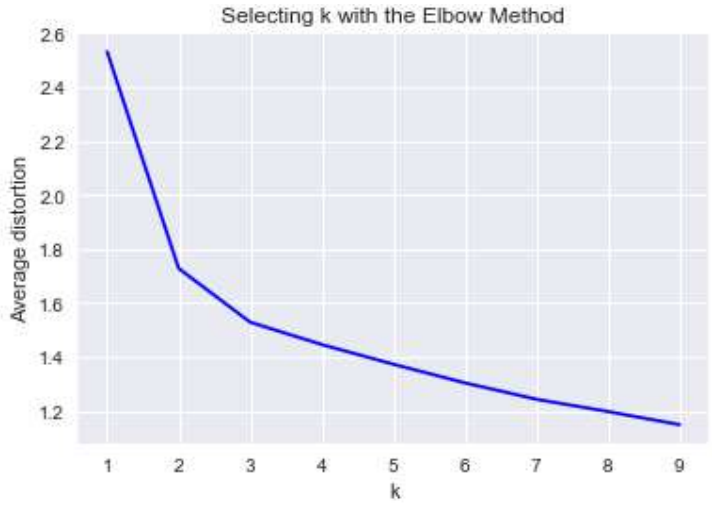

Figure 5- Elbow chart for the first layer of the print

Step 3: A 1000 x 1001 matrix is generated with the first 100 columns representing layers and the last column hosting the target value (Y). The number of rows of the newly generated matrix is the number of samples (1000). Note that the first 100 columns of this matrix contain numbers representing their corresponding cluster members while the last column consists of numbers of either 0 or 1 representing printing quality. The newly generated matrix is as follow.

$$\begin{pmatrix} 1 & 1 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \end{pmatrix}$$

Step 4: A random forest algorithm is applied to the matrix generated in Step 3. Using the ranking tool provided in the random forest model, the layers can be sorted based on their significance value. In this example, 12 important layers are selected at the significance level at 0.013. Using random forest importance function embedded in Scikit-Learn [60] package, the significance value of five first printing layers are shown in Table 6.

Table 6- Importance factor for printing layers from 10 to 14

Layers	…	layer 10	layer 11	layer 12	layer 13	layer 14	…
Significance	…	0.0075	0.0349	0.0060	0.0345	0.0065	…

A significance level is chosen based on the respected importance values for all layers. By sorting the importance values for all layers, significant layers can be easily distinguished. We are interested in having a handful of classifiers to be able to predict the faulty prints early. However, a higher number of classifiers means more computation and more false alarms. In addition, the limitation of computation time will prevent real-time monitoring from taking place. Thus, there is a tradeoff between time and accuracy in the selection of the best value of the significance level. Table 6 shows that between layers 10 and 14, layers 11 and 13 are the most significant layers while layers 10, 12, and 14 are not significant based on weights, which add to 1. In this case, users can choose a significance level or critical value of 0.013. Then layers 11, 13, 29, 31, 36, 40, 45, 48, 52, 57, 89, and 97 can be selected as the most significant layers according to this criterion.

Step 5: By using the results from step 3 and 4, 12 (m=12) classifiers are trained. The first classifier uses up to layer 11 and the second classifier uses layers up to 13 to predict the target value Y**,** i.e. $Y = 0$ means good and $Y = 1$ means defect and the rest classifiers are based on the other significant layers detected in step 4. The classifiers were trained based on 90% of the data (900 samples). The rest of the 100 samples were used to validate the prediction result. Based on the testing sets, the first and the second classifiers obtained 75% and 79% of accuracy respectively. The same computation has been done for the rest of the classifiers. Once the classifiers are generated, we are ready for process monitoring. The classification models then are as follow:

$$\text{Prediction} \begin{cases} \text{Model 1: } \widehat{Y}_i = f_1(C_{i1}, C_{i2} \dots, C_{i11}) \\ \text{Model 2: } \widehat{Y}_i = f_2(C_{i1}, C_{i2} \dots, C_{i13}) \\ \dots \\ \text{Model 12: } \widehat{Y}_i = f_{12}(C_{i1}, C_{i2} \dots, C_{i97}) \end{cases} \quad \textbf{Equation 10}$$

Phase II:

Step 6: A new printing job is proceeding. For example, job number 1001 is forwarded to one of the printing machines and process monitoring is underway to predict the printing quality (i.e. target value) during the printing process. Once the data from up to the first significant layer of printing (layer 11) is realized, the K-means algorithm in Step 2 would generate a cluster for layers 1 to 11. Suppose the first layer contains the printing data for the job 1001 as (oxygen level, optical bank temperature, pump temperature, and laser power) = (0.104, 999.48, 4999.56, 50), the assigned cluster is zero. Similarly, the K-means assigns the proper cluster to layers 2 to 11.

Step 7: Using the first classifier (Model 1) built in Step 5, a prediction on printing quality can be obtained. If the result is that the printing quality is good (i.e. $Y = 0$), the printing process continues to print. Otherwise, engineers are asked to adjust the controllable parameter (laser power) in the next layers to prevent the production of a defective part. In this example, the prediction of the print quality by the first classifier is good i.e. $Y = 0$. Therefore, we move to the next significant layer for quality prediction. The second classifier imports clusters up to layer 13 and predicts the print quality as $Y = 0$. This shows that the process is controlled, and no adjustment is necessary. The same procedure continues to up the last significant layer.

Building and training the classification and clustering model contribute to phase I of the process monitoring technique. The trained models are capable of detecting the defects and prevent them by warning operators to adjust the parameters in the rest of the process. Then, phase II is the

deployment of the trained models on the new coming data where the target value is unknown. This step performs the tasks that are defined in phase II of the process monitoring technique. However, MLCPM inherits the same features as data-driven approaches where they need as many samples as possible to achieve good accuracy [17, 38, 40, 44, 45]. In today's industries, AM techniques are used generally to produce prototypes or very complex parts in which the number of productions is not large. Therefore, the number of samples to train a model may not be adequate. However, having a 3D printing farm with many printers producing the same part can potentially generate many samples, which may enable the use of ML techniques to be effective.

MLCPM is capable of analyzing the printing process at significant layers. It benefits from several ML techniques to control the printing process. The number of layers of a printing part depends on layer thickness. It may start with 50 layers to thousands of layers. Therefore, an automated process monitoring method such as MLCPM may provide a plausible way for close-loop feedback process control.

2.3.2. Validation Based on Simulated Cases

The simulated dataset includes 276 defective parts ($Y = 1$) and 724 good parts ($Y = 0$). The models were trained using 90% randomly chosen of the total dataset. Therefore, 100 samples were used for testing purpose. The first predictive trained model (model 1) acquired 75% accuracy using a randomly selected dataset contains 26 and 74 defective and good parts respectively. However, the confusion matrix shows that model 1 has misclassified 25 parts where 15 parts are misclassified as good where they were defective parts (i.e. false positive or FP) and 10 parts are misclassified as defective where they were good parts (i.e. false negative or FN). The second model is improved since it benefits from more layers. The random selection of testing dataset is repeated for model 2 where it includes 26 and 74 defective and good parts respectively. Model 2 acquired

79% accuracy while it misclassified 21 parts including 16 FPs and 5 FNs. The statistics for the rest of the models can be acquired in the same way.

2.4. CONCLUSION

The laser melting processes such as SLM are very complex. This complexity makes process monitoring and printing reliability very challenging. There may be more than 50 parameters impacting the printing quality in the SLM process. The correlation of these parameters is still unknown and is ignored in the current state of the art. The proposed framework illustrates a plausible approach that melt-pool parameters along with other parameters may provide a way to adjust the printing process during the melting process for defect reduction. Therefore, feedback control systems for the metal printing process is achievable if the prediction models are highly accurate. The proposed framework trains various ML models based on the data accumulated from multiple printing machines. Theoretically, prediction accuracy is greatly enhanced with large enough data sets that include most if not all defective cases.

We propose a MLCPM framework based on ML techniques to monitor the printing process. ML techniques such as classification can effectively help the AM industry to build predictive models and control the process using multiple process variables. In addition, ML techniques such as feature selection methods could reduce the amount of data drastically. Data reduction helps reduce the amount of time needed to process the data and provides a possibility to monitor the printing process in real time. The proposed framework techniques should foster efforts towards the ultimate goal of 100% yield [66].

For future research, the proposed MLCPM framework should be validated through real datasets from various printing parts. The proposed framework has been tested on simulation data only. We plan to compare its performance against the current process monitoring technologies in

terms of a number of defective parts and model accuracy rates. Since a good training phase is the main part of making any predictive model more accurate, we are seeking more effective ways to improve training. In the current state of the art of process monitoring, a training set often remains fixed. However, we believe that the training set must be updated periodically to enable MLCPM to cover the new possible patterns during the printing process. It is still an open-ended research question of how the proposed classification and prediction models are updated and how frequent they should be updated.

The proposed MLCPM framework warns printing operators about possible defective part production before finishing the process. Therefore, they have time to adjust the parameters to avoid the production of defective parts. In the future, we will explore possible approaches toward making adjustment process automatically. The layer-based nature of the proposed MLCPM may enable the feedback control algorithms possible by collecting big data on controllable parameters during all printing processes.

In manufacturing, training sets generally do not contain many bad samples (defective parts). Therefore, the accuracy of the classifier may suffer due to the omission of adequate bad samples in a training data set. To populate the training set with more defective samples, the method called artificial contrast data may be used to further improve the proposed models. The concept of artificial contrast data was first proposed by Tuv and Runger [13]. The idea enables classification techniques as a possible tool for process monitoring. In a nutshell, it simulates out-of-control cases to expose any ML model to patterns of OC sample observations. To make the artificial data, Tuv and Runger [13] used a uniform random number generator. The random numbers were generated based on the range of out of control points. The contrast data generation was repeated for each

parameter independently. It is not clear how the use of artificial contrast can help improve the proposed MLCPM. Future studies are in order.

In addition, the challenges and solutions provided in this study have a common major element that is the data. The data is the input into the analysis systems and predictive models. Recently, many important decisions are data-driven, which makes data quality (DQ) at the center of trustworthiness for an organization's business intelligence [67]. With the large amount and wide varieties of data generated by sensors from machines daily at high speed, engineers have always assumed that data quality is perfect without proper verification [68]. When this assumption is no longer valid, one of the approaches to tackle the uncertainty of the data is simulating multiple scenarios. Simulation helps to consider the unseen scenarios into the analysis. This is very crucial since ML techniques use sample data to make predictive models. If a pattern generated from uncertainty is not covered in the training set, the model is not able to fully cover possible patterns and therefore, the accuracy drops. Thus, the quality of the data must be verified and assessed. MLCPM can benefit from studies with the consideration of data quality assessment such as that in [67] to guard against data uncertainty.

Chapter 3 - A Data-Driven Monitoring Model for High Dimensional Semiconductor Manufacturing Systems

3.1. Introduction

This chapter aims to study a system-wide process monitoring system based on predictive models. The proposed platform intends to monitor the manufacturing process and trigger necessary alarms while any process is heading into a detrimental quality outcome. The proposed framework is capable of process monitoring for high dimension multistage systems. The proposed system can be applied to various multi-stage systems such as assembly lines and layer-by-layer additive manufacturing [32]. The proposed model is an extension of our previously published work (MLCPM) for additive manufacturing in that each layer of print is considered as a production stage. Specifically, MLCPM provides a multi-layer predictive model process monitoring tool for the metal 3D print industry. MLCPM benefits from several supervised and unsupervised machine learning methods to tackle the prediction and high dimensionality problems. MLCPM is the base of this work. The proposed framework includes multiple stage-wise, predictive models that incorporate process parameters in a semiconductor production system. Since the dimension of this kind of process parameters is enormous, several data reduction techniques are applied to make the problem less complex to save the time of computation. For example, the SECOM dataset used as the case study in this study includes around 600 parameters. Hence, some techniques are necessary to reduce computation costs and avoid overfitting.

Also, any machine learning model including the proposed ones requires updated training data for prediction accuracy. Moreover, the unbalanced nature of the process monitoring problem, that is more IC observations available than OC ones may cause too many false alarms (i.e., type I errors) or miss out defective prone processes (i.e., type II errors). Hence, we have investigated

ways to mitigate this issue and explored multiple machine learning techniques to implement the proposed system. The ensemble of machine learning modeling and computations were implemented by python 2.7 using Scikit-Learn [60] package.

3.2. Building Predictive Models to Detect Faults

This section provides the initial steps toward building the intelligent stage-wise process monitoring in high dimensional multistage systems. Predictive models have been widely used in the industry. Especially in the field of process monitoring, many researchers have proposed the use of predictive models to classify product quality. Most of the studies have considered product quality as either good or defective (binary classification). The predictive models are generally classifiers based on supervised models such as Bayesian network, random forest [19], and support vector machine (SVM) [69] where a set of training data (includes both good and defective classes) are given to train a classifier. Then, the classifier can be used to classify a new observation into either the good or defective category. In general, assuming a problem with n samples and m parameters, a classification model can be modeled as:

$$Y = f(x) \; where \; (x, Y) = (x_{i1}, x_{i2}, \dots, x_{im}, Y_i), \quad i = 1, \dots, n \qquad \textbf{Equation 11}$$

where Y is the class set of data (target value of a quality characteristic) and x is the feature set (AKA process parameter or variable).

In a manufacturing process, x represents the setting of a process parameter such as temperature and Y is the quality state class (which could be 0 as good or 1 as defective). In some cases, the quality state can be more than two classes. In that situation, decision tree-based classifiers can work without modification. However, classifiers such as SVM need to be modified for multilabel classification. Two commonly used methods include one-vs-rest and one-vs-one. Assuming N different classes, the one-vs-rest method trains one binary model for each class where

classes are based on one class versus the rest (seen as a single class). In other words, for $j \in \{1, \dots, N\}$, single classifier will be trained where the class of j will be seen as positive and the rest as negative. In this approach, a total of N classifiers are trained for all classes. On the other hand, the one-vs-one approach makes a binary classifier for each pair of classes. Therefore, total $N(N-1)/2$ classifiers need to be trained. Either method has its own advantage and disadvantage. One-vs-one is computationally expensive but does not cause imbalance problems where one-vs-rest method encounter imbalance problem and hence, cannot be solved with general classifiers such as generic SVM. The quality parameter in this study is a binary variable and hence, we do not face a multilabel classification problem.

Process parameters could be either numerical or categorical. In a multistage system, the result of each stage is the input to the next stage in the system. Hence, each stage contributes to the final quality state. Depending on the applications, production time varies from seconds to days or even weeks. Hence, predicting the faulty process before the last step can save plenty of time and avoid costs. The first step to build predictive models is data collection of production parameters affecting the final quality state in all stages. Assuming a manufacturing process with k process parameters scattered in various production stages, a training data set that contains n produced products can be listed in Table 7.

Table 7- A Training Dataset

Part	Production Parameters				Target Value
	Parameter 1	Parameter 2	…	Parameter k	
Part 1	x_{11}	x_{12}	…	x_{1k}	Y_1
Part 2	x_{21}	x_{21}	…	x_{2k}	Y_2
…	…	…	…	…	…
Part n	x_{n1}	x_{n2}	…	x_{nk}	Y_n

where x_{ij} is the measured value of the j^{th}process parameter for part i and Y_i is the quality characteristic for part i. Using the training data as Table 7, we propose to establish an ensemble predictive platform based on k predictive models as follow:

$$\text{Predictive Models} \begin{cases} \text{Model 1: } Y = f_1(x) \text{ where } (x, Y) = (x_{i1}, Y_i) \\ \text{Model 2: } Y = f_2(x) \text{ where } (x, Y) = (x_{i1}, x_{i2}, Y_i) \\ \dots \\ \text{Model k: } Y = f_k(x) \text{ where } (x, Y) = (x_{i1}, x_{i2}, \dots, Y_i) \end{cases} \quad \textbf{Equation 12}$$

The proposed platform enables online process monitoring during the production process where the data up to each point, can be used in the set of models to predict the final quality state. Therefore, k different assessments on the final quality state are performed to ensure the quality process.

Several classification models are promising candidates for the function *f*(x) in equation (12). Based on [70], K-Nearest Neighbors (KNN), Naïve Bayes (NB), Neural Network (NN), Decision Tree (DT), and SVM are effective classifiers for anomaly detection problems. Also, Logistic Regression (LR) and Random Forest (RF) are among the list of classifiers. In unbalance classification problems, accuracy is not the best evaluation measurement where specificity, sensitivity, and area under the curve (AUC) are more effective. Hence, we propose to compare the classifiers based on the specificity, sensitivity, and AUC evaluation. The specificity and sensitivity equations are as follow:

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad \textbf{Equation 13}$$

$$Sensitivity = \frac{True\ Positive}{True\ Positive + Fa\quad Negative} \quad \textbf{Equation 14}$$

where true negative refers to the correctly classified good parts, true positive refers to correctly classified bad parts, false negative (type II error) refers to the parts that are truly bad while the model has misclassified them as good, and the false positive (type I error) refers to the parts that are truly good, however, the model has mistakenly classified them as bad. These measurements are usually placed in a confusion matrix for the evaluation of any classifier. For more details about the confusion matrix, the reader can refer to [71]. AUC calculates the area under the Receiver Operating Characteristic (ROC) curve, which is a figure resulted by plotting true positive rate (TPR) and false positive rate (FPR) of classifiers where TPR is equal to sensitivity and $\mathrm{FPR} = 1 - Specificity$. Higher the AUC score, the better the quality of prediction.

3.3. Tackle Dimensionality Problem

The proposed system can detect faulty products while the product is still in a production process but may generate many false alarms and disrupt normal production operation. This issue is more pronounced in a high dimensional problem because a huge number of parameters (k) are included. Hence, we propose the use of dimension reduction techniques to reduce the time and computation. The complexity reduction can be applied in two different stages.

3.3.1. Stage Clustering

We assume that the production process in a stage follows certain patterns. Therefore, instead of feeding the original raw data for modeling, clustering methods are proposed to identify these patterns. Model building time will be greatly reduced when the dimension reduction is accomplished. Toward this goal, the data from each stage should be gleaned from Table 7. Note that curating data into the format shown in Table 7 is nontrivial because process parameter data comes in different timestamp at different stages. Most experts agree that data wrangling usually takes about 80% of a data analytics project [72].

Assuming p parameters in stage j, Table 8 illustrates the production data for stage j where X_{ijk} represents the collected data for part i and parameter k in stage j.

Table 8- Data collected for stage *j*

Part	Production Parameters in stage j			
	Parameter 1	Parameter 2	…	Parameter p
Part 1	X_{1j1}	X_{1j2}	…	X_{1jp}
Part 2	X_{2j1}	X_{2j2}	…	X_{2jp}
…	…	…	…	…
Part n	X_{nj1}	X_{nj2}	…	X_{njp}

To perform the clustering analysis we propose to use the K-means algorithm in Scikit-Learn [58] package available for Python 2.7. K-means is the widest clustering method that has been used in many applications. After applying the K-means in Table 8, a cluster will be assigned for each part's data in a stage. It should be noted that the K-means ++ method embedded in Scikit-Learn, and Euclidean method are chosen as initial centroid and distance method for K-means. Since the K-means algorithm needs the knowledge of the number of clusters, "elbow method" [61, 62] is used to determine an efficient number of clusters. The elbow method uses the distortion within clusters to determine the efficient number of clusters. The elbow method is a graphical tool to detect an efficient number of clusters.

Most stages may have many possible production setting combinations. However, we assume that there are only small limited numbers of combinations that are used in production.

The K-means algorithm used first assigns each processed part in a given stage to a cluster based on Equation 15. Then it is repeated for all parts and all stages to form the matrix illustrated in Table 9. The outcome of this step is that each part has one identified cluster for each stage.

$$C_{ij} = g(X_{ijk}), C_{ij} \in \left\{C_{1j}, C_{2j}, \dots, C_{l_j j}\right\}$$ **Equation 15**

where g is the clustering function, X_{ijk} is the measurement of parameter k for part i in stage j, C_{ij} is the assigned cluster for part i within stage j and l_j is the number of clusters for stage j obtained by the elbow method.

Table 9- Classification matrix

Part (i)	Production Stage (j)				Target value (Y)
	Stage 1	Stage 2	…	Stage M	
Part 1	C_{11}	C_{12}	...	C_{1M}	Y_1
Part 2	C_{21}	C_{22}	...	C_{2M}	Y_2
…	...	...	...	...	...
Part n	C_{n1}	C_{n2}	...	C_{nM}	Y_n

Assuming total M production stages in Table 9, Y_i is the product quality characteristic for part i as it can be simply defined as 0 (for good) and 1 (for defective) part. Table 9 is then used as the training and testing set for predictive models (Equation 12). Hence, the total number of required predictive models drops from K to only M models. This step reduces computation time as the manufacturing processes usually include many processes parameters (K) while the number of stages (M) are limited.

Clustering reduces the parameters per stage into a limited number of classes (i.e. the assigned cluster). The training set then is used to perform the clustering model. After training, the trained clustering model will assign an appropriate cluster to each new data. This process has a huge impact on complexity reduction. Then, instead of using a large number of process parameters, the assigned clusters can be used in the predictive models of Equation 12. However, the stages that have only one process parameter may not need to go through the clustering process.

3.3.2. Stage Selection

In multistage systems, not all the stages have the same impact on the final quality characteristic. Accordingly, we propose the use of a few selected highly important stages for building predictive models. Feature selection techniques can be applied to achieve this purpose. For example, RF provides the feature importance values in the predictive models. Then, a cut off number can be chosen to select only the most important stages. However, several different methods such as BN, LR, and KNN can also be applied and compared to the RF algorithm. In this study, we apply several classification models on all stages and then, using the best classification model in terms of AUC for the stage selection process (feature selection).

This step will reduce the number of predictive models when limited or the most important models are selected. For example, in a production process containing 100 stages, RF can sort the importance of the stages regarding the product quality. Then, the limited number of those stages (for example, stage 20, stage 50, and stage 90) can be chosen to predict the product quality. Then, three predictive models will be generated where the first predictive model includes stages up to 20 but with stage 20 as the most important stage. The second predictive model includes stages up to 50 but with the most significant stages 20 and 50. The last predictive model includes the most significant stages up to 90 but with the most significant stages 20, 50, and 90. In general, assuming m significant stages in a production system with a total of M stages, m predictive models can be built as follow:

$$\text{Predictive Models} \begin{cases} \text{Model 1: } \widehat{Y}_\iota = f_1(C_{i1}) \\ \text{Model 2: } \widehat{Y}_\iota = f_2(C_{i1}, C_{i2}) \\ \dots \\ \text{Model m: } \widehat{Y}_\iota = f_m(C_{i1}, C_{i2}, \dots, C_{im}) \end{cases} \qquad \textbf{Equation 16}$$

where C_{ij} is the predicted cluster for part i in stage j and $\widehat{Y}_i$ is the predicted quality for part i. Two proposed complexity reduction techniques will reduce the time and effort of performing the proposed process monitoring framework.

3.4. Training Update Process and Imbalance Classes

As stated in Chapter 1, the update process and Imbalanced classes are the two main challenges for the proposed framework. To address the update process, we propose to use the AUC score as a threshold scale. Once the predictive models are trained, new datasets in the deployment process can be evaluated. Then, this new dataset can be appended to the training set. So, the new training set grows as the new datasets join. Generally, more data improves the accuracy of data-driven approaches. However, the established models are based on the previous training sets and they need to be evaluated using the updated dataset. Since the AUC score can identify the misclassifications, a threshold should be set to trigger a warning when the AUC score goes below the threshold. At this point, clustering and prediction models should be trained again using the new training set. These updated models will reduce the chance of misclassification. However, while the models are under training, the process monitoring can still take place using the old models. As the training set grows, the training time will increase. But the model performance is expected to improve. A balance must be struck for how often to repeat the clustering and training process. The pseudo code for the training update procedure can be seen as follow:

Specify the training_evaluation time and the threshold

Trigger training procedures.

Deploy the trained models on new coming samples.

Evaluate the new sample and add it to a separate dataset called temp_training.

Update the current time.

If time= training_evaluation time, then

Deploy the trained models on the temp_training and obtain AUC.

If AUC>= threshold then

continue

Else

add temp_training to the training set, trigger the training procedures using the new training set

To address the imbalance classes, two general approaches considered include oversampling and undersampling. As the names suggest, the undersampling method aims to balance the classes by trimming the data in the majority group while the oversampling method aims to increase the numbers of samples in a minority group. In this study, a random undersampling method has been used as the under-sampling approach. In this approach, all data points from the minority group plus randomly selected points from the majority group have been considered for the training set. The number of selected data points from the majority group is equal to the total number of samples in the minority group. For oversampling approach, SMOTE [31] (Synthetic Minority Over-sampling Technique) has been selected. SMOTE does not simply randomly copy from the available points in the minority group, but also it creates synthetic minority class examples. The algorithm selects similar samples from the minority group (using

distance methods) and creates an instance using interpolation of the selected points. In this study, SMOTE method is the one used in Scikit-Learn [60] package for Python 2.7. These techniques will be applied to the SECOM [1] dataset in the numerical example section.

3.5. Model Deployment (Process Monitoring)

Once the clusters and predictive models are generated, new production data can be evaluated by the trained models. First, K-means is applied to data in each stage to create clusters. Then, when the first significant stage is reached, the first predictive model (Equation 16) will predict the initial result. This prediction provides the likelihood of the quality outcome while the semi-finished product is still in the manufacturing process. If the result is a success, the process continues up to the next significant stage where the second predictive model can be applied to predict the quality. Otherwise, engineers have enough time to change the process parameters so that this semi-finish part may have a better chance to be good. The same procedure is repeated until all of the crucial stages are reached.

The main difference between the proposed model and the other studies is that the proposed model provides an evaluation of the product while the product is still in the production process. The other machine learning methods reviewed in the literature all wait until the very end to generate a prediction. Generally, at the end production stage, product quality has been determined and there is no chance to go back and adjust process parameters to improve product quality. Details of the proposed method will be shown in a numerical example in the next section.

3.6. A Numerical Example

The proposed model is applied to SECOM dataset – a semiconductor manufacturing dataset extracted from the UCI repository lab [1]. Each row of the dataset contains production parameters and the final quality stage. SECOM consists of 1567 examples each with 591 features

and a label containing the classification of the quality characteristic. The classifications reported as +1 stands for a defective part while -1 is for a good part. In all, 104 parts were identified as defectives (+1) and 1463 parts were failed (-1). It is clear that this data set is very unbalanced in term of the classes presented.

3.6.1. Data Preprocessing

Among 591 features, 116 of them are fixed (meaning its row value does not change) and therefore do not contribute any information toward this classification problem. Hence, they were dropped from this study. According to [73], each data point is generated by a manufacturing sensor. Data columns representing process parameters can be divided into five groups each representing a manufacturing workstation [28]. In addition, there are missing data reflected as empty cells in the data set. In this study, all empty cells were filled with the most frequently occurred number within each feature.

The first step is to divide all columns of data based on five production stages as stated in [28]. Table 10 demonstrates the process parameter data within all stages. For example, the first stage contains parameter readings from parameter 1 to parameter 107. After splitting the data, K-means can be applied to assign clusters to parts within each stage.

Table 10- SECOM dataset

Part	Stage 1				…	Stage 5			
	Parameter 1	Parameter 2	…	Parameter 107	…	Parameter 493	Parameter 494	…	Parameter 590
Part 1	3030.93	2564.00	…	0	…	10.0167	2.9570	…	0
Part 2	3095.78	2465.14	…	0	…	10.0167	3.2029	…	208.2045
…	…	…	…	…	…	…	…	…	…
Part 1567	2944.92	2450.76	…	0.0009	…	10.0167	2.7756	…	137.7844

3.6.2. Clustering

Before applying K-means, the efficient number of clusters for each stage data should be identified. According to the elbow method, 4,3,3,4, and 4 are the most efficient number of clusters for stages 1 to 5, respectively. For example, according to elbow chart in Figure 6, stage 5 can be divided into 4 clusters.

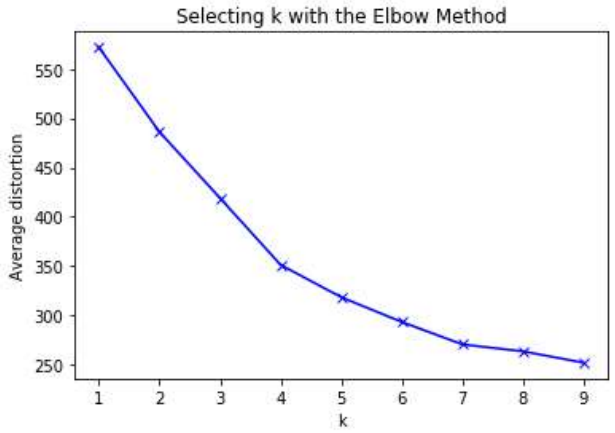

Figure 6- Elbow method for stage 5

After identifying the clusters, K-means can be applied on stages data to assign a cluster to each part within a stage. Then using identified clusters, a classification matrix can be formed as shown in Table 11 based on the SECOM dataset.

Table 11- Classification matrix

Part	Production Stage					Target value (Y)
	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	
Part 1	0	0	2	1	3	-1
Part 2	4	3	0	1	3	+1
…	…	…	…	…	…	…
Part 1567	1	0	0	3	4	-1

3.6.3. Classification

Table 11 is a sample of the input array into classification models. The classification type of this problem is binary and, hence, a handful number of classification models can be applied. We have applied KNN, NB, NN, DT, LR, RF, and SVM on the classification matrix. Figure 7 shows the evaluation of these models on the classification matrix from Table 11. Evaluation of models is based on accuracy, sensitivity, specificity, and AUC scores. The classifiers were trained on 70% of the data while 30% has been reserved for validation. It is noted that the training and testing sets were randomly selected.

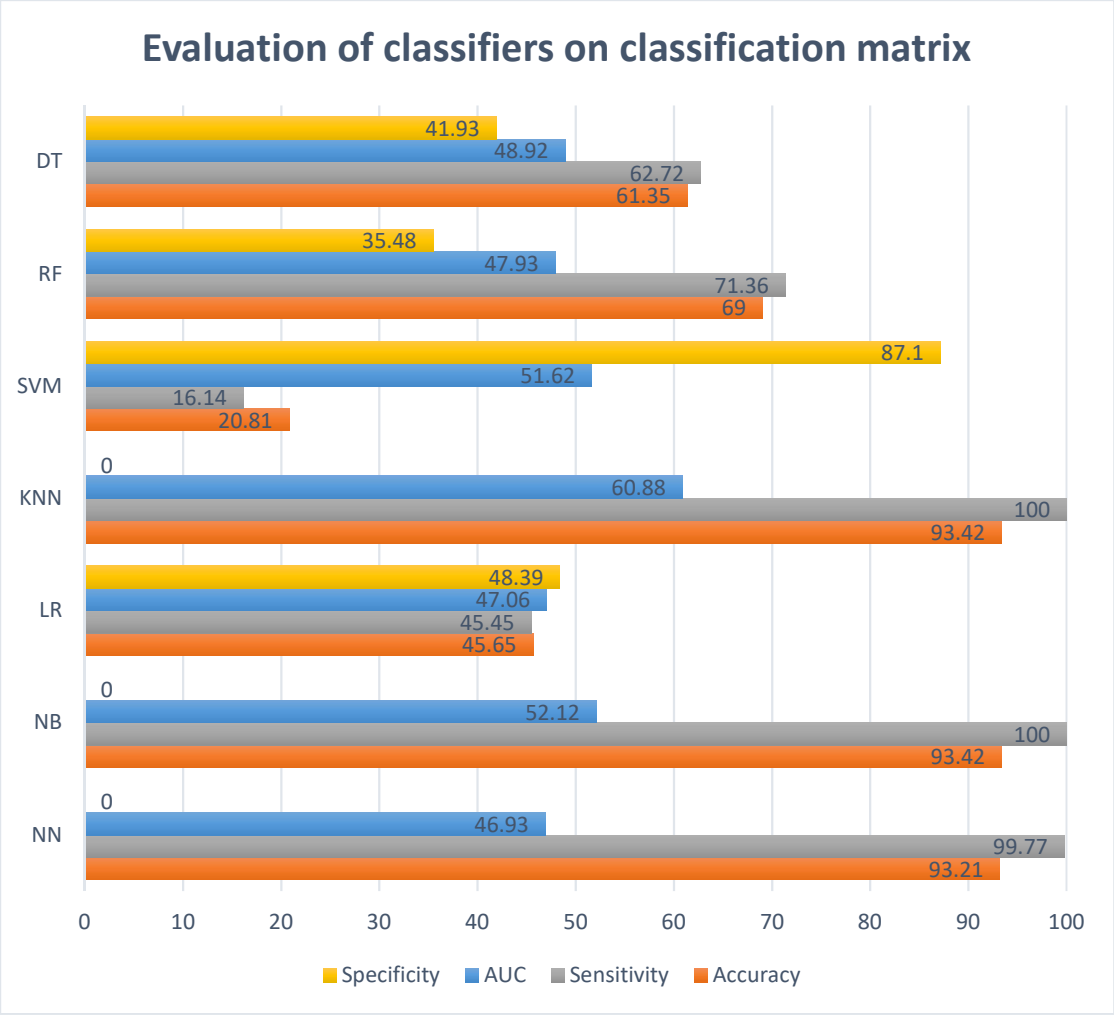

Figure 7- Evaluation of classifiers

Due to the imbalanced nature of the dataset, a balanced classification is required in all classifiers except for NN, KNN, and NB since these classifiers do not adjust weights based on a number of classes. In each set of evaluation (i.e. on original data, with undersampling, and with oversampling) the classifiers have been tuned to perform the best in terms of the AUC score. The tuning has been performed by H2O autoML platform [74].

Among all classifiers listed in Figure 7, RF, and DT perform the best in terms of overall evaluation scores. Although KNN, NN, and NB perform better in terms of accuracy, however, they have misclassified all the bad parts (zero specificity). In the imbalanced sampling problem, AUC, specificity, and sensitivity are better tools to evaluate a model than accuracy.

The next step is to tackle the imbalance problem of the dataset. As discussed before, two general approaches are oversampling and undersampling [31]. To solve the imbalance classification, we applied techniques using imblearn package [75] available for both python versions 2 and 3. It should be noted that in all of the discussed classification models, oversampling and undersampling methods were applied to the training set (70% of the data) but not on the testing set (30%).

Undersampling was applied by imblearn [75] set where data was bootstrapped from the majority class with the same size of the minority class. Using this technique, Figure 8 was generated by the evaluation of models with the newly trained model (trained with the undersampling method). Figure 8 shows clearly that undersampling method improves the training of the classifiers in terms of specificity and sensitivity scores. However, the accuracies of the models were dropped dramatically. For example, the NN had the accuracy of 93.21 before under sampling but the new rate is only 50.11. This phenomenon is due to the fact that by adjusting the samples, accuracy reflects a more realistic number than before.

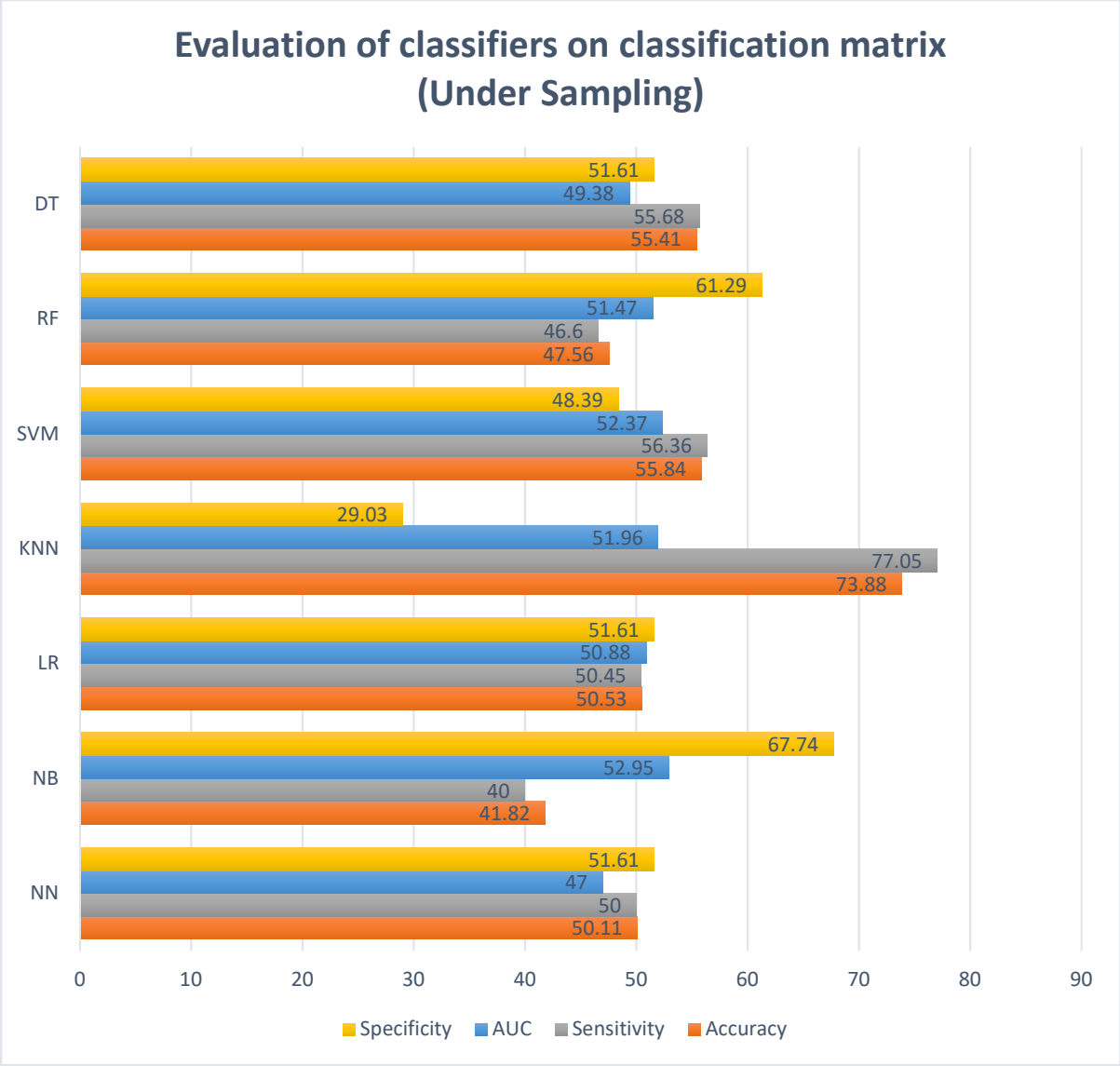

Figure 8- Evaluation of classifiers after applying the undersampling method

Next oversampling was applied to the original data using SMOTE [31] embedded in imblearn [75]. Training the classifiers with the oversampled dataset, Figure 9 was generated using the new training set.

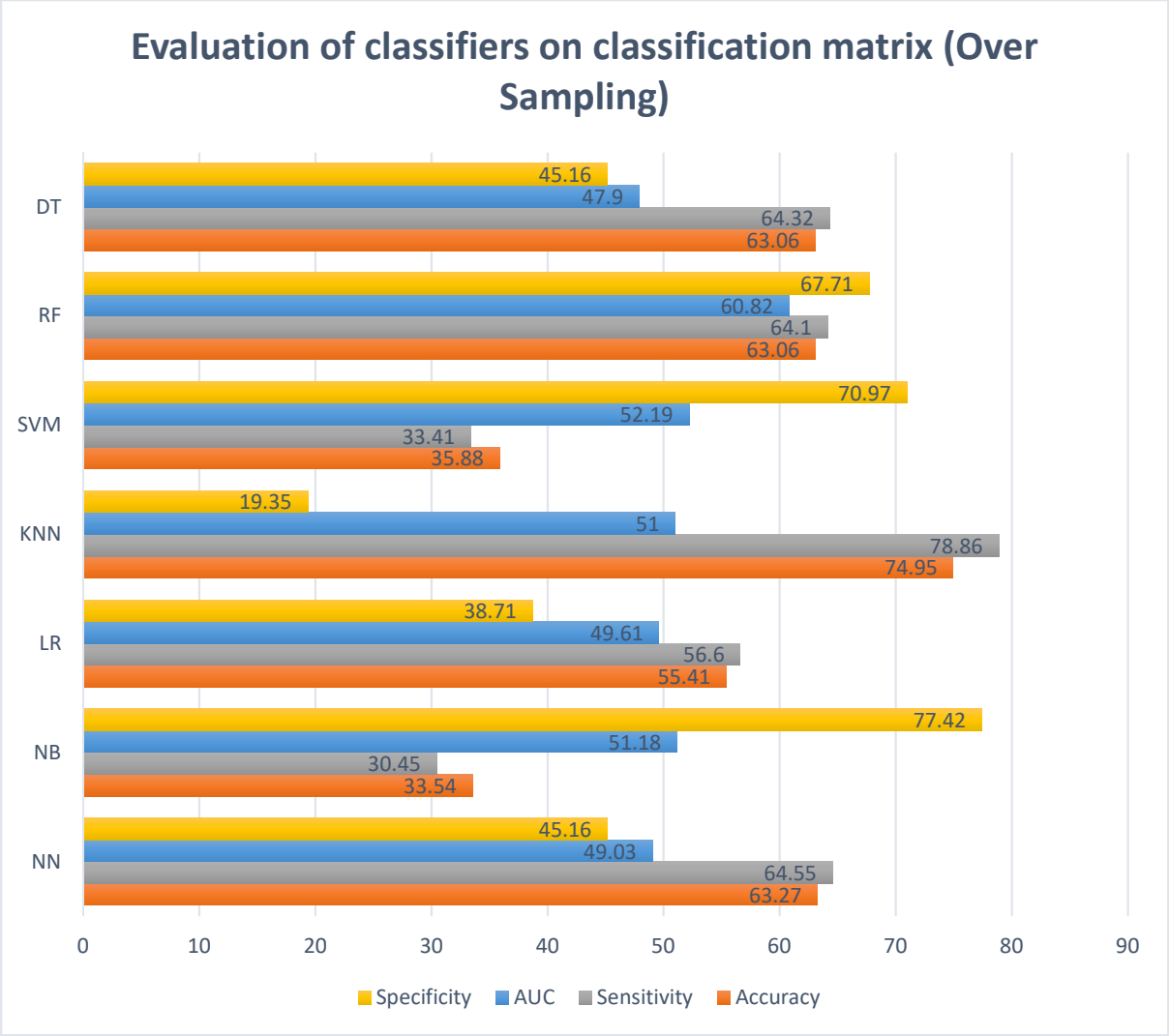

Figure 9- Evaluation of classifiers after applying the oversampling method

Like undersampling, the oversampling method by SMOTE [31] improves the training of the classifiers in term of specificity and sensitivity. But unlike undersampling, the accuracy values did not drop too much from the original analysis. As shown in Figure 9, we concluded that RF performs the best in terms of the overall consideration of accuracy, specificity, sensitivity, and AUC metrics. Therefore, oversampling by SMOTE has been selected as the imbalanced approach. Table 12 summarizes all different combinations.

Table 12- Evaluation of all models

Model	Original Data				Undersampling				Oversampling			
	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC	Accuracy	Sensitivity	Specificity	AUC
DT	61.35	62.72	41.93	48.92	55.41	55.68	51.61	49.38	63.06	64.32	45.16	47.9
RF	69	71.36	35.48	47.93	47.56	46.6	61.29	51.47	63.06	64.1	67.71	60.82
SVM	20.81	16.14	87.1	51.62	55.84	56.36	48.39	52.37	35.88	33.41	70.97	52.19
KNN	93.42	100	0	60.88	73.88	77.05	29.03	51.96	74.95	78.86	19.35	51
LR	45.65	45.45	48.39	47.06	50.53	50.45	51.61	50.88	55.41	56.6	38.71	49.61
NB	93.42	100	0	52.12	41.82	40	67.74	52.95	33.54	30.45	77.42	51.18
NN	93.21	99.77	0	46.93	50.11	50	51.61	47	63.27	64.55	45.16	49.03

After establishing the main predictive model, the next step is to select the most important stages to reduce the number of predictive models. RF has an inherent feature property that can weigh the features based on the amount of information they provide toward detecting the patterns of classes. Table 13 shows the importance of stages in the selected classifier.

Table 13- Importance factor for production stages

Stages	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
Importance	0.1101	0.3580	0.108	0.1305	0.2934

A significance level of 0.25 is chosen based on the respected importance values for all stages in Table 13. Then, stages 2 and 5 are selected as the most important stages toward predicting the final quality. The stage selection helps to reduce the number of predictive models. This feature selection stage may be trivial in this example. However, it would be significant for the other applications such as 3D prints where thousands of layers are required to finish a product. In addition, more predictive models will provide more chance to catch the faulty process. On the other hand, it may also increase false alarm or false negative rates and time computation. Therefore, there is a tradeoff between these considerations. In this example, two classifiers can be built as shown in Equation (17).

$$\text{Predictive Models} \begin{cases} \text{Model 1: } \widehat{Y}_i = f_1(C_{i1}, C_{i2}) \\ \text{Model 2: } \widehat{Y}_i = f_2(C_{i1}, C_{i2}, \dots, C_{i5}) \end{cases} \quad \textbf{Equation 17}$$

The first model includes data from the first two stages where the second model includes all production stages data. Model 2 was trained previously by applying RF using SMOTE method on 70% of the data while 30% was reserved for validation. The same procedure was done for model 1 while only two stages were considered. Figure 10 shows the metrics for both models.

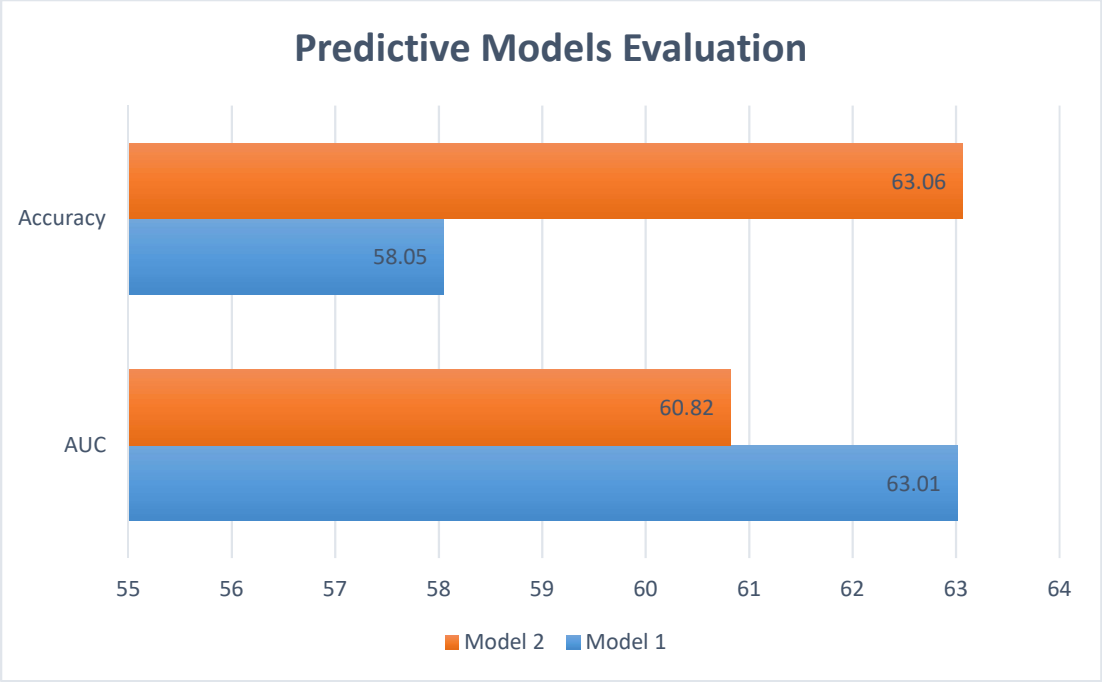

Figure 10- Evaluation of stage-based predictive models-based predictive models

After establishing the predictive models, process monitoring phase or Phase II of SPC can be initiated. Specifically, a new part proceeds to the production line. When this new product reaches stages 1 and 2, the K-means models associated with each stage will assign an appropriate cluster to each. Then, the first predictive model would provide a prediction on the final quality state (-1 for a good and +1 for a defective product). If the prediction results as good (-1) then the process continues up to the last stage to provide a prediction using model 2. Otherwise, process engineers have plenty of time to control the process to prevent producing a faulty product by examining historical data of those parts having the same pattern as this new part in the first two stages but still ending to be good part at the end of stage 5. Process engineers can then use the machine settings in stages 3, 4, and 5 of the good parts for this new part.

Chapter 4 - Conclusions and Future Research

4.1. Summary

Multistage systems are a major part of the general production of goods and service. Today's manufacturing processes are much more complex. Sensors embedded throughout multiple stages of the production processes generate a huge amount of data in high dimensions. Traditional quality engineering methods cannot be implemented effectively in this modern-day production environment. Most process parameters are often not used for decision making. Control charting is usually implemented independently throughout the production stages. A recent development in machine learning methods may provide solution strategies. Although various existing classification-based process monitoring techniques have been implemented for multi-stage processes, these methods either focus on quality characteristics on the project or just provide quality predictions at the end of the manufacturing process and offer no chance to fix potential problems during production. Also, the literature to date is largely silent on comprehensive models addressing issues related to high dimensions, new unseen fault behaviors, and unbalanced nature of training data set.

This research proposes a process monitoring framework for high-dimensional, multistage processes. The proposed framework offers an opportunity to provide a prognosis of product quality and mitigation strategies before the production of a product is finished. Hence, in a costly production system unlike the state of the art studies that do not provide real control over the manufacturing process, the proposed framework can save time, effort, and cost by warning about the defective procedure while the part is still in production.

The proposed framework has been successfully applied in the additive manufacturing industry and semiconductor manufacturing industry.

To address the high dimensionality challenge, the proposed framework benefits from two complexity-reduction steps as clustering and stage selection (feature selection). In the clustering step, we reduced the within stage parameters into a limited number of production recipes representing the behavior of the manufacturing in that stage. The stage selection step further selects the highly impactful stages affecting the final quality. The proposed complexity reduction techniques successfully reduced the computation of the proposed framework applied to two datasets with a dimension of 400 and 600 variables in the additive manufacturing and semiconductor manufacturing, respectively.

As discussed, updating the training set is one of the challenges in classification-based process monitoring techniques. The current practices in process monitoring only required an IC dataset as the reference data (training set) while the training of machine learning models requires both the IC data set as well as the data sets with different OC patterns of data. Due to the unbalanced nature of the manufacturing datasets, we proposed to use an AUC metric in order to monitor the updating process of the training set. A threshold is proposed to set to trigger the alarm to update the training set. Once the AUC metric results below the threshold, the training set should be updated with the newly identified samples.

Rare OC points is another challenge for the classification-based monitoring models. In manufacturing datasets, there is a lack of OC conditions where the product quality is usually at a satisfactory level. Hence, the historical data is full of healthy condition data while OC data consists of a very small fraction of the entire database. Unbalance data sets lead to poor classification accuracy. Hence, we conducted a comparative study using two different oversampling and undersampling methods using the SECOM dataset. The comparative study shows that the oversampling method results in better evaluation metrics than the undersampling method. The

limited number of available data to train the model can be the reason for better oversampling results.

The effectiveness of the proposed process monitoring framework has been shown by the evaluation of the results in two different case studies. In both cases, the proposed framework successfully warns about prone failure in the system while the manufacturing process is not finished. Then, process engineers have the opportunity to alter process parameters in order to save the identified part.

A recent development in Industry 4.0 [7] and the Internet of things (IoT) [29] have enabled the environment in which the proposed framework may become a reality. We expect this research to have a significant impact on product or service quality in various multistage systems not limited to manufacturing. Using cloud computing technologies, the proposed research may handle the process monitoring of a supply chain. The implementation of the proposed research could lead to smart firms where processes can be adjusted automatically in real time. This can help to bring the ideal system of a zero-defect production system closer to reality.

4.2. Future Studies

For future research, the training set should be periodically updated with new data points. Hence, in addition to the proposed strategy, the current developed practices in the field of continues learning can be studied to maintain the trained models at a satisfactory level. The continuous learning is still in the early stages; however, several optimized developed models are currently available. For example, Watson ML framework developed by IBM has already a continuous learning feature that can help to maintain the training models updated.

Also, a prognosis model such as a Bayesian Network can be used to establish stage-based predictive models. Since the proposed model generates limited clusters for multiple stage-based

modeling, a prognosis model can benefit from tremendous complexity reduction. This prognosis model can be used to suggest process parameter settings in unfinished stages to prevent defective production. Once a part is identified to be potentially faulty, then a search procedure is necessary to find the best possible production settings in the unfinished stages to guide the faulty part to back into healthy conditions. Since the number of parameters is large, an efficient approach is necessary to perform a fast and efficient computation and search among all possible production settings. A prognosis model or algorithm is required to provide potential impactful process parameters and proper settings for control purposes.

The effectiveness of a prognosis model can be evaluated using an accessible manufacturing process such as a metal 3D printing or semiconductor manufacturing processes. The prognosis part of the proposed framework may also be applied to non-manufacturing applications. A prognosis model is under development for a drug recommendation system. The proposed model can be based on classification-based models. The recommender system called 1DrugAssist currently is currently designed for two different diseases: breast cancer, and type 2 diabetes. 1DrugAssist utilizes machine-learning based predictive models to recommend a drug to reduce the patient's risk. The risk is calculated based on five different possible outcomes such as death, hospitalization, disability, life-threatening, and other serious outcomes. The risk level can be measured by the probability of the target classes. For example, a response variable as death indicates the death record for a patient. Then, a drug which has the least probability toward the negative class (death class) will be favorable against the rest of the candidates. The underlying predictive models have been trained on public FDA Adverse Events Reporting System (FAERS) [76]. FAERS dataset includes patients' visit considering the disease, drug information, demographic information, and outcomes. To enrich the dataset, drug characteristics such as drug class, target, and pathway have

been extracted from the DrugBank database [77]. Then, based on the trained models, a recommender system has been generated to recommend a drug with least risk level based on the patient's information such as gender, age, weight, country, and limiting the drug target. A user-friendly web-based system [78] has been developed and is publicly available. In addition, since the risk is associated with multiple outcomes, a simulation-based weighing system has been established to glean the relative importance of outcomes in users' point of view while aiming to reduce the sensitivity of results. However, the reaction associated with the recommended drug has not been completely understood yet. One of the challenges in this study is the large categorical variables in the dataset where the reaction set includes more than 1500 unique values and different techniques have yet to be applied on the dataset to check the association between the recommended drug, and reactions.

References

1. Dheeru, D. and E.K. Taniskidou, *UCI machine learning repository (2017).* URL http://archive. ics. uci. edu/ml, 2017.
2. Shi, J. and S. Zhou, *Quality control and improvement for multistage systems: A survey.* IIE Transactions, 2009. 41(9): p. 744-753.
3. Shewhart, W., *Economic quality control of manufactured product 1.* Bell System Technical Journal, 1930. 9(2): p. 364-389.
4. Ceglarek, D. and J. Shi, *Dimensional variation reduction for automotive body assembly.* Manufacturing Review, 1995. 8(2).
5. Hotelling, H., *Multivariate quality control illustrated by the air testing of sample bombsites*. 1947, New York: McGraw-Hill. p. 111.
6. Amini, M. and S. Chang. *A review of machine learning approaches for high dimensional process monitoring*. in *Proceedings of the 2018 Industrial and Systems Engineering Research Conference, Orlando, FL*. 2018.
7. Foidl, H. and M. Felderer. *Research challenges of industry 4.0 for quality management*. in *International Conference on Enterprise Resource Planning Systems*. 2015. Springer.
8. Friedman, J., T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Vol. 1. 2001: Springer series in statistics New York.
9. Lowry, C.A., W.H. Woodall, C.W. Champ, and S.E. Rigdon, *A multivariate exponentially weighted moving average control chart.* Technometrics, 1992. 34(1): p. 46-53.
10. Runger, G.C. and M.C. Testik, *Multivariate extensions to cumulative sum control charts.* Quality and Reliability Engineering International, 2004. 20(6): p. 587-606.
11. Jin, J. and J. Shi, *State space modeling of sheet metal assembly for dimensional control.* Journal of Manufacturing Science and Engineering, 1999. 121(4): p. 756-762.
12. Pan, J.-N., C.-I. Li, and J.-J. Wu, *A new approach to detecting the process changes for multistage systems.* Expert Systems with Applications, 2016. 62: p. 293-301.
13. Tuv, E. and G. Runger, *Learning patterns through artificial contrasts with application to process control.* WIT Transactions on Information and Communication Technologies, 2003. 29.

14. Hu, J., G. Runger, and E. Tuv, *Tuned artificial contrasts to detect signals.* International Journal of Production Research, 2007. 45(23): p. 5527-5534.
15. Li, F., G.C. Runger, and E. Tuv, *Supervised learning for change-point detection.* International Journal of Production Research, 2006. 44(14): p. 2853-2868.
16. Hu, J. and G. Runger, *Time-based detection of changes to multivariate patterns.* Annals of Operations Research, 2010. 174(1): p. 67-81.
17. Deng, H., G. Runger, and E. Tuv, *System monitoring with real-time contrasts.* Journal of Quality Technology, 2012. 44(1): p. 9-27.
18. Hwang, W., G. Runger, and E. Tuv, *Multivariate statistical process control with artificial contrasts.* IIE transactions, 2007. 39(6): p. 659-669.
19. Breiman, L., *Random forests.* Machine learning, 2001. 45(1): p. 5-32.
20. Poggio, T. and S. Smale. *The mathematics of learning: Dealing with data*. in *2005 International Conference on Neural Networks and Brain*. 2005. IEEE.
21. Jiang, W. and K.-L. Tsui, *A theoretical framework and efficiency study of multivariate statistical process control charts.* IIE Transactions, 2008. 40(7): p. 650-663.
22. Wang, K. and W. Jiang, *High-dimensional process monitoring and fault isolation via variable selection.* Journal of Quality Technology, 2009. 41(3): p. 247-258.
23. Zou, C. and P. Qiu, *Multivariate statistical process control using LASSO.* Journal of the American Statistical Association, 2009. 104(488): p. 1586-1596.
24. Jin, Y., S. Huang, G. Wang, and H. Deng, *Diagnostic monitoring of high-dimensional networked systems via a LASSO-BN formulation.* IISE Transactions, 2017. 49(9): p. 874-884.
25. Tibshirani, R., *Regression selection and shrinkage via the lasso.* Journal of the Royal Statistical Society Series B, 1996. 58(1): p. 267-288.
26. Wuest, T., C. Irgens, and K.-D. Thoben, *An approach to monitoring quality in manufacturing using supervised machine learning on product state data.* Journal of Intelligent Manufacturing, 2014. 25(5): p. 1167-1180.
27. Uhlmann, E., R.P. Pontes, A. Laghmouchi, and A. Bergmann, *Intelligent pattern recognition of a SLM machine process and sensor data.* Procedia Cirp, 2017. 62: p. 464-469.

28. Kao, H.-A., Y.-S. Hsieh, C.-H. Chen, and J. Lee. *Quality prediction modeling for multistage manufacturing based on classification and association rule mining*. in *MATEC Web of Conferences*. 2017. EDP Sciences.
29. Morgan, J. *A simple explanation of 'The Internet of Things'*. Leadership 2014 [cited 2018 12/1/2018]; Available from: http://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#475fab4e6828.
30. Marr, B., *A Short History of Machine Learning*, Forbes, Editor. 2016: Forbes. p. 1-2.
31. Chawla, N.V., K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer, *SMOTE: Synthetic minority over-sampling technique.* Journal of Artificial Intelligence Research, 2002. 16: p. 321-357.
32. Amini, M. and S.I. Chang, *MLCPM: A process monitoring framework for 3D metal printing in industrial scale.* Computers & Industrial Engineering, 2018. 124: p. 322-330.
33. Wohlers, T., *3D Printing and Additive Manufacturing State of the Industry*, W.R. 2016, Editor. 2016.
34. Wohlers, T., *3D Printing and Additive Manufacturing State of the Industry Annual Worldwide Progress Report*, in *Wohlers Report 2015*, T. Wohlers, Editor. 2014.
35. Spears, T.G. and S.A. Gold, *In-process sensing in selective laser melting (SLM) additive manufacturing.* Integrating Materials and Manufacturing Innovation, 2016. 5: p. 2.
36. Tapia, G. and A. Elwany, *A Review on Process Monitoring and Control in Metal-Based Additive Manufacturing.* Journal of Manufacturing Science and Engineering-Transactions of the Asme, 2014. 136(6).
37. Dunsky, C., *Process monitoring in laser additive manufacturing.* Industrial laser solutions, 2014. 29(5): p. 14-18.
38. Chua, C.K., C.H. Wong, and W.Y. Yeong, *Standards, quality control, and measurement sciences in 3D printing and additive manufacturing*. 2017: Academic Press.
39. Smith, K.T., *Big data security: The evolution of hadoop's security model.* August 14, 2013, Link; http://www. infog. com/articles/HadoopSecurityModel, 2013.
40. Gibson, I., D.W. Rosen, and B. Stucker, *Additive manufacturing technologies*. Vol. 17. 2014: Springer.
41. Evans, J., *DMLS: A Bumpy Road in History.* Design & Motion, 2014. 10.

42. Materials, A.S.f.T.a., *Committee F42 on Additive Manufacturing Technologies - Scope*, ASTM, Editor. 2009.
43. 3DEO. *3DEO*. 2018; Available from: http://www.3DEO.com/.
44. Park, H., N. Tran, and D. Nguyen. *Development of a predictive system for SLM product quality*. in *IOP Conference Series: Materials Science and Engineering*. 2017. IOP Publishing.
45. Patterson, A.E., S.L. Messimer, and P.A. Farrington, *Overhanging features and the SLM/DMLS residual stresses problem: Review and future research need.* Technologies, 2017. 5(2): p. 15.
46. Amini, M. and S. Chang. *Process Monitoring of 3D Metal Printing in Industrial Scale*. in *ASME 2018 13th International Manufacturing Science and Engineering Conference*. 2018. American Society of Mechanical Engineers.
47. Purtonen, T., A. Kalliosaari, and A. Salminen, *Monitoring and adaptive control of laser processes.* Physics Procedia, 2014. 56: p. 1218-1231.
48. Kruth, J.-P., P. Mercelis, J. Van Vaerenbergh, and T. Craeghs. *Feedback control of selective laser melting*. in *Proceedings of the 3rd international conference on advanced research in virtual and rapid prototyping*. 2007. Taylor & Francis Ltd.
49. Yadroitsev, I., P. Krakhmalev, and I. Yadroitsava, *Selective laser melting of Ti6Al4V alloy for biomedical applications: Temperature monitoring and microstructural evolution.* Journal of Alloys and Compounds, 2014. 583: p. 404-409.
50. Chivel, Y. and I. Smurov, *On-line temperature monitoring in selective laser sintering/melting.* Physics Procedia, 2010. 5: p. 515-521.
51. Malekipour, E. and H. El-Mounayri, *Defects, process parameters and signatures for online monitoring and control in powder-based additive manufacturing*, in *Mechanics of Additive and Advanced Manufacturing, Volume 9*. 2018, Springer. p. 83-90.
52. Imani, F., A. Gaikwad, M. Montazeri, P. Rao, H. Yang, and E. Reutzel. *Layerwise in-process quality monitoring in laser powder bed fusion*. in *ASME 2018 13th International Manufacturing Science and Engineering Conference*. 2018. American Society of Mechanical Engineers.
53. Yao, B., F. Imani, A.S. Sakpal, E.W. Reutzel, and H. Yang, *Multifractal analysis of image profiles for the characterization and detection of defects in additive*

manufacturing. Journal of Manufacturing Science and Engineering, 2018. 140(3): p. 031014.

54. Montazeri, M., R. Yavari, P. Rao, and P. Boulware, *In-process monitoring of material cross-contamination defects in laser powder bed fusion.* Journal of Manufacturing Science and Engineering, 2018. 140(11): p. 111001.
55. Morsali, S., S. Daryadel, Z. Zhou, A. Behroozfar, M. Baniasadi, S. Moreno, D. Qian, and M. Minary-Jolandan, *Multi-physics simulation of metal printing at micro/nanoscale using meniscus-confined electrodeposition: Effect of nozzle speed and diameter.* Journal of Applied Physics, 2017. 121(21): p. 214305.
56. Montazeri, M. and P. Rao, *Sensor-Based Build Condition Monitoring in Laser Powder Bed Fusion Additive Manufacturing Process Using a Spectral Graph Theoretic Approach.* Journal of Manufacturing Science and Engineering, 2018. 140(9): p. 091002.
57. Metal-AM. *Simufact to launch process simulation software solution for metal Additive Manufacturing*. 2016 1/1/2016]; Available from: http://www.metal-am.com/simufact-launch-process-simulation-software-solution-metal-additive-manufacturing.
58. Esi-Group. *Metallic Additive Manufacturing Process Simulations*. 2016 1/1/2016]; Available from: https://www.esi-group.com/software-solutions/virtual-manufacturing/additive-manufacturing.
59. Grasso, M. and B.M. Colosimo. *Statistical process monitoring of Powder Bed Fusion processes via in-situ video imaging*. in *FACAM 2018*. 2018.
60. Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, *Scikit-learn: Machine learning in Python.* Journal of machine learning research, 2011. 12(Oct): p. 2825-2830.
61. Ketchen, D.J. and C.L. Shook, *The application of cluster analysis in strategic management research: an analysis and critique.* Strategic management journal, 1996. 17(6): p. 441-458.
62. Hackeling, G., *Mastering Machine Learning with scikit-learn*. 2017: Packt Publishing Ltd.
63. Singh, A., A. Yadav, and A. Rana, *K-means with Three different Distance Metrics.* International Journal of Computer Applications, 2013. 67(10).

64. Campanelli, S.L., N. Contuzzi, A. Angelastro, and A.D. Ludovico, *Capabilities and performances of the selective laser melting process*, in *New Trends in Technologies: Devices, Computer, Communication and Industrial Systems*. 2010, IntechOpen.
65. Jones, E., T. Oliphant, and P. Peterson, *{SciPy}: Open source scientific tools for {Python}*. 2014.
66. Goehrke, S.A. *Metal 3D Printing with Machine Learning: GE Tells Us About Smarter Additive Manufacturing*. 2017 [cited 2017 1/1/2017]; Available from: https://3dprint.com/191973/3d-printing-machine-learning-ge/.
67. Amini, M. and S. Chang. *Assessing Data Veracity for Data-Rich Manufacturing*. in *IIE Annual Conference. Proceedings*. 2017. Institute of Industrial and Systems Engineers (IISE).
68. Saha, B. and D. Srivastava. *Data quality: The other face of big data*. in *2014 IEEE 30th International Conference on Data Engineering*. 2014. IEEE.
69. Cortes, C. and V. Vapnik, *Support-vector networks.* Machine learning, 1995. 20(3): p. 273-297.
70. Omar, S., A. Ngadi, and H.H. Jebur, *Machine learning techniques for anomaly detection: an overview.* International Journal of Computer Applications, 2013. 79(2).
71. Lancaster, F. and V. Gale, *Pertinence and relevance.* Encyclopedia of Library and Information Science, 2003. 2: p. 2307-2316.
72. Dasu, T. and T. Johnson, *Exploratory data mining and data cleaning*. Vol. 479. 2003: John Wiley & Sons.
73. Arif, F., N. Suryana, and B. Hussin, *Cascade quality prediction method using multiple PCA+ ID3 for multi-stage manufacturing system.* Ieri Procedia, 2013. 4: p. 201-207.
74. H2O.ai, *Python Interface for H2O*. 2016, H2O.ai: H2O.ai.
75. Lemaître, G., F. Nogueira, and C.K. Aridas, *Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning.* The Journal of Machine Learning Research, 2017. 18(1): p. 559-563.
76. FDA, *Administration USF and D. FDA Adverse Event Reporting System (FAERS)* FDA, Editor. 2018.

77. Wishart, D.S., C. Knox, A.C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey, *DrugBank: a comprehensive resource for in silico drug discovery and exploration.* Nucleic acids research, 2006. 34(suppl_1): p. D668-D672.
78. 1DataGroup. *Intelligent Medicine Recommender System*. 2019 [cited 2019 4/23/2019]; Available from: http://1data.olathe.ksu.edu/drugassist/.