Hybrid Statistical and Deep Learning Models for Diagnosis and Prognosis in Manufacturing Systems

Mohd Safwan Ahmad Mohd Ibrahim Ansari

A Thesis

in

The Department

of

Mechanical, Industrial and Aerospace Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Industrial Engineering) at

Concordia University

Montréal, Québec, Canada

November 2020

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mohd Safwan Ahmad Mohd Ibrahim Ansari**

Entitled: **Hybrid Statistical and Deep Learning Models for Diagnosis and Prognosis in Manufacturing Systems**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Industrial Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

______________________________ Chair
Dr. Hossein Hashemi Doulabi

______________________________ External Examiner
Dr. Amin Hammad

______________________________ Examiner
Dr. Hossein Hashemi Doulabi

______________________________ Supervisor
Dr. Farnoosh Naderkhani and Dr. Anjali Awasthi

Approved by ______________________________
Dr. Martin Pugh, Chair
Department of Mechanical, Industrial and Aerospace Engineering

____________ 2020

____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Hybrid Statistical and Deep Learning Models for Diagnosis and Prognosis in Manufacturing Systems

Mohd Safwan Ahmad Mohd Ibrahim Ansari

In today's highly competitive business environment, every company seeks to work at their full potential to keep up with competitors and stay in the market. Manager and engineers, therefore, constantly try to develop technologies to improve their product quality. Advancements in online sensing technologies and communication networks have reshaped the competitive landscape of manufacturing systems, leading to exponential growth of Condition Monitoring (CM) data. High-dimensional data sources can be particularly important in process monitoring and their efficient utilization can help systems reach high accuracy in fault diagnosis and prognosis. While researches in Statistical Process Control (SPC) tools and Condition-Based Maintenance (CBM) are tremendous, their applications considering high-dimensional data sets has received less attention due to the complexity and challenging nature of such data and its analysis. This thesis adds to this field by designing a Deep Learning (DL) based survival analysis model towards CBM in the prognostic context and a DL and SPC based hybrid model for process diagnosis, both using high dimensional data. In the first part, we a design support system for maintenance decision making by considering degradation signals obtained from CM data. The decision support system in place can predict system's failure probability in a smart way. In the second part, a Fast Region-based Convolutional Network (Fast R-CNN) model is applied to monitor the video input data. Then, specific statistical features are derived from the resulting bounding boxes and plotted on the multivariate Exponentially Weighted Moving Average (EWMA) control chart to monitor the process.

Acknowledgements

In the name of Allah, the Most Gracious and the Most Merciful.

All praises are to Allah and to Him alone. I thank Him for all the opportunities and trials that He put forward for me and the capability He granted to finish writing this thesis. I am also grateful to Him for all the learnings and experiences I had during the process. After that, I am grateful, indebted and pleased to acknowledge and convey my gratitude to a lot of people, for without their guidance, collaboration, and encouragement, this thesis would have hardly been possible.

First and foremost, I would like to sincerely thank my supervisor Professor Farnoosh Naderkhani for her constant guidance, understanding, patience and especially for putting up with me during the bad phases of both experimentation and writing process. Her constant encouragement and warm spirit along with assisting me with the correct resources throughout the program really helped me stay on the right track and finish this thesis within time. The door to her office was always open whenever I ran into trouble or had a question about my research or writing. I am grateful for all the encouragements during my degree and opportunities she provide for me.

Special appreciation goes to Professor Anjali Awasthi, for her supervision and constant support in the beginning when I ran into problems with finding a supervisor in a different department, and then for considering me for an amazing internship opportunity at Siemens, which was an enriching experience to say the least. She was there for any and every issue I faced.

This journey would have not been the same without Professor Daria Terekhov. Her suggestions during our meetings in the beginning of the program not only helped me take the right courses but she also directed me towards Dr. Naderkhani for my thesis, for which I am immensely grateful. Also, I would like to thank Professor Hossein Hashemi Doulabi for the friendly conversations which were filled with lessons that proved their significance over time.

I must express my gratitude to my parents and my siblings for their prayers and support throughout. Words cannot express how thankful I am for their wise counsel and sacrifices.

Sincere thanks to all my colleagues and friends especially Negar, Akhil, Ankit, Abdul Razaq, Talal and others moral support during my study. This accomplishment would not have been possible without them. Thank you.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
PDF	Probability Density Function
CM	Condition Monitoring
SPC	Statistical Process Control
CBM	Condition-Based Maintenance
DL	Deep Learning
PHM	Proportional Hazards Model
CNN	Convolutional Neural Network
DNN	Deep Neural Networks
RUL	Remaining Useful Life
OS	Operational Setting
SM	Sensor Measurement
FFNN	Feed Forward Neural Network
Fast R-CNN	Fast Region-based Convolutional Network method
EWMA	Exponentially Weighted Moving Average
MEWMA	Multivariate Exponentially Weighted Moving Average
PM	Preventive Maintenance
ML	Machine Learning
DL	Deep Neural Networks
ROI	Region of Interest
ANOVA	Analysis of variance
CI	Concordance Index
NB	Naive Bayes
BN	Bayesian Networks
ANN	Artificial Neural Networks
SVM	Support Vector Machines
SVR	Support Vector Regression

RVM	Relevance Vector Machine
SGD	Stochastic Gradient Descent
CQA	Critical Quality Attributes
CUSUM	Cumulative Sum

Chapter 1

Thesis Introduction and Overview

1.1 Motivation

Under today's aggressive business environment, it is critical and of paramount importance that manufacturing and industrial sectors operate at their full potential with maximum availability. In this regard, both manufacturing systems/equipment and processes should be monitored over time to detect any abnormality in the systems and processes.

All systems in manufacturing and industries deteriorate over time due to everyday use and duration and fail eventually. This can lead to additional operation cost and reduced availability of the system. Also, it is shown that degradation of equipment has a direct effect on quality of final product. In order to increase the efficiency of manufacturing systems and processes, proper maintenance and monitoring ssytems should be designed and developed. In particular, to avoid costly failures in manufacturing systems, preventive maintenance (PM) is commonly performed while the system is still operational. Traditional PM models (age-based replacement models, block replacement models and periodic maintenance models) do not take into the account the information obtained from CM and it results in inaccurate failure predictions and low maintenance cost reduction. Therefore, the state-of-the-art maintenance program referred to as CBM is introduced to take into considerations the CM data. With the abundance of CM data becoming readily available with an advancement in computing power together with advancements in Machine Learning (ML), special, Deep Neural

Networks (DNNs), have made it possible to intelligently and more efficiently monitor industrial systems and optimally perform maintenance decision making. Similar trend is seen in monitoring manufacturing processes. Several embedded sensors have led to exponential growth of data in different type such as waveform, i.e., timeseries signals and high-dimensional data, i.e., images and videos. High-dimensional data sources play an important role in process monitoring and control due to providing thousands of rich data points. Efficient utilization of such high-dimensional data sources leads to highly accurate results in fault diagnostics. While the researches on maintenance and process monitoring are tremendous, the application of SPC and CBM considering high-dimensional data sets has received less attention due to complex characteristics of such high-dimensional data including high variety, high-dimensionality, high velocity, and complex spatial/temporal structure. This motivate us to develop advanced maintenance and process monitoring techniques in present of high-dimensional CM data for fault diagnostic and prognostic via applying proper DL-based solutions.

1.2 Thesis Statement

With the growth of complex high-dimensional data combined with almost a trillion-fold increase in computing performance within the last several decades, a drastic revolution in DL research is seen, especially since last two decades. The benefits and variety of application potential of these researches have been proven time and again by the advantages reaped by the investors and stake holders and witnessed by observers in all case these researches were applied to. From self driving cars, advanced recommender systems and disease detection to effective and better prediction capability in almost anything worth predicting, the fields that have gained from DL is a topic worth exploring in itself. This premise would make it easy to understand the possibility of a drastic advancements in certain fields in terms of applying such methods compared to others.

One such interesting example is survival analysis. While the researches on survival analysis has seen a similar exponential progress, most of the works focus on application in the healthcare sector, that it was initially developed for, and the development of a framework to apply the state of the art researches in the maintenance context has received less attention, though objectively, both may be seen as fields with similar end goals and structures, with their own variations considered. As morbid as the analogy sounds, the approaches taken to accurately predict the time of failure of a machine which is subject to degradation over time, given a set of factors, is not too different than predicting time of death of a patient diagnosed with certain disease, given the factors affecting the subject are measured and recorded. Efficient utilization of high-dimensional data sources and state of the art researches to exploit such data can lead to highly accurate fault prognosis and diagnosis, paving a way for an advanced and efficient CBM policies. The main goal of this thesis is to review the advanced DL based survival analysis and object tracking methods and attempt to develop a framework for prognostic and diagnostic within quality control and CBM contexts by employing DL-based solutions for two separate case studies in quality control and CBM area.

1.3 Objectives and Contributions

The thesis objective is to develop and design DL-based solutions for fault diagnostic and prognostic in process quality control and CBM domains. The first research work is performed in the prognostic context in which the survival probabilities estimation is carried out on a popular NASA turbofan dataset wherein the engines deteriorate over time. We begin by analysing the data and understanding the patterns, distribution and correlation between covariates and transformation of data to suit the model formats. Two models are used on the dataset, the first being time-varying Cox's PHM, a variation of the popular Cox's PHM, and the second being Non-Proportional Cox-Time, a neural network based survival analysis model.

An important contribution in this part of the thesis is applying a DL-based model towards a maintenance dataset and a conclusion that a DL-based model will most likely give better performance compared to a statistical model unless the dataset is intentionally or unintentionally modelled on the statistical model itself.

The second part of the thesis deals with developing a DL-based solutions for process monitoring in diagnostic context in which a simulated visual dataset is used to develop a hybrid model that combines a state-of-the-art DL algorithm with SPC tool to monitor a manufacturing process represented by the visual data at hand. In particular, a Fast R-CNN is applied in order to monitor the image sequences over time. Then, some statistical features are derived and plotted on the EWMA control chart. This is a novel approach in the sense that no work has been seen that combines object tracking with an SPC tool in the way that output of the object tracking algorithm is used towards generating control charts in real time. Specifically, determining the coordinates of the bounding boxes for the region of interest (ROI) and determining the error vectors by comparison with the in-control parameters to develop the control chart is completely unseen far as our understanding goes.

In both case studies, the steps and requirements necessary to apply the models on the selected dataset are discussed in detail along with the challenges and limitations in their applications followed by interpretation and discussion on the results.

1.4 Outline

The rest of the thesis is organized as follows: Chapter 2 consists of a detailed literature review and background on the topic of interest of this thesis. It includes a detailed discussion on survival analysis, the latest trends in DL-based survival analysis research that sets the basis for chapter 3 which a DL-based survival analysis is performed for asset degradation modeling based on NASA data set. This is followed by a discussion on process quality control through real-time monitoring, description of conventional approaches and then the trends in the field associated with our work which is discussed in chapter 4 such that an advanced hybrid DL and SPC model to monitor the manufacturing process in presence of high-dimensional monitoring data is developed. Finally, the conclusion and potential future work are discussed in chapter 5.

Chapter 2

Background & Literature Review

As mentioned previously, the thesis is divided into two main parts, i.e., diagnostic and prognostic in quality control and maintenance context. In this chapter, the comprehensive review is provided, starting with the prognostic part in CBM domain followed by real-time process monitoring methods in quality control context. The chapter begins with a detailed discussion on survival analysis, including its definition and description of data, censoring, the functions and their equations and then review the popular Cox's PHM referred to as Cox's PHM to survival analysis. We also provide a detailed discussion on the latest trends in survival analysis research that are associated with our work which is discussed in Chapter 3. Then, we concentrate the attention on the diagnostic part, which includes a discussion on process quality control through online or real-time monitoring, describe the conventional approaches and then discuss the trends in the field associated with our work discussed in Chapter 4.

2.1 Survival Analysis

Survival analysis has been a very active research field for several decades, especially in healthcare domain. The application of survival analysis is not limited to healthcare and due to its universal applicability, it is widely applied in different research areas including but not limited to maintenance management, financial engineering and service industries. An important contribution that stimulated the entire field was the counting process formulation given by [1]. Since then a large number of fine text books have been written on survival analysis and counting processes, with some key references being [2], [3], [4] and [5]. Excellent texts aimed at the biostatistical community with biomedical application as the motivating factor include [6], [7], [8] and [9].

Survival analysis is a branch of statistics used to analyze the expected duration of time an event, for instance, death in biological organisms and failure in mechanical systems. The most common application appears in medicine, where common events of interest are population deaths, treatments etc. Based on a specific set of information used as input, it attempts to answer questions like ”What proportion of a population which will survive past a certain time?” or ”of those that survive, at what rate will they die or fail?” or ”can multiple causes of death or failure be taken into account?” or ”how do particular circumstances or characteristics increase or decrease the probability of survival?” In engineering applications, survival analysis is sometimes referred to as reliability theory.

In this section, detailed description on survival analysis through its definition and terms like censoring, survival and hazard functions and equations are provided. Then, a discussion on several modelling approaches to survival analysis with a special focus on the popular Cox's PHM is provided.

2.1.1 Introduction

Kaplan and Meier [10] defines survival analysis as a set of tools that model the time to an event from a common start. Time could mean years, months, weeks, or days from the beginning of follow-up of a system (individual or a machine) until the event occurs. Alternatively, time can also mean age of a system when an event occurs. The event could include disease incidence, relapse from remission, recovery, death, or any designated experience of interest that may happen to a system. The time variable is referred to as survival time, since it informs about the time that a subject has “survived” over some period of follow-up. The event is also referred to as failure, since the event of interest usually is death though not necessary. It does usually include disease incidence, or some other negative individual experience. It can also mean in some cases the “time to return to work after an elective surgical procedure,” in which case failure is a positive event.

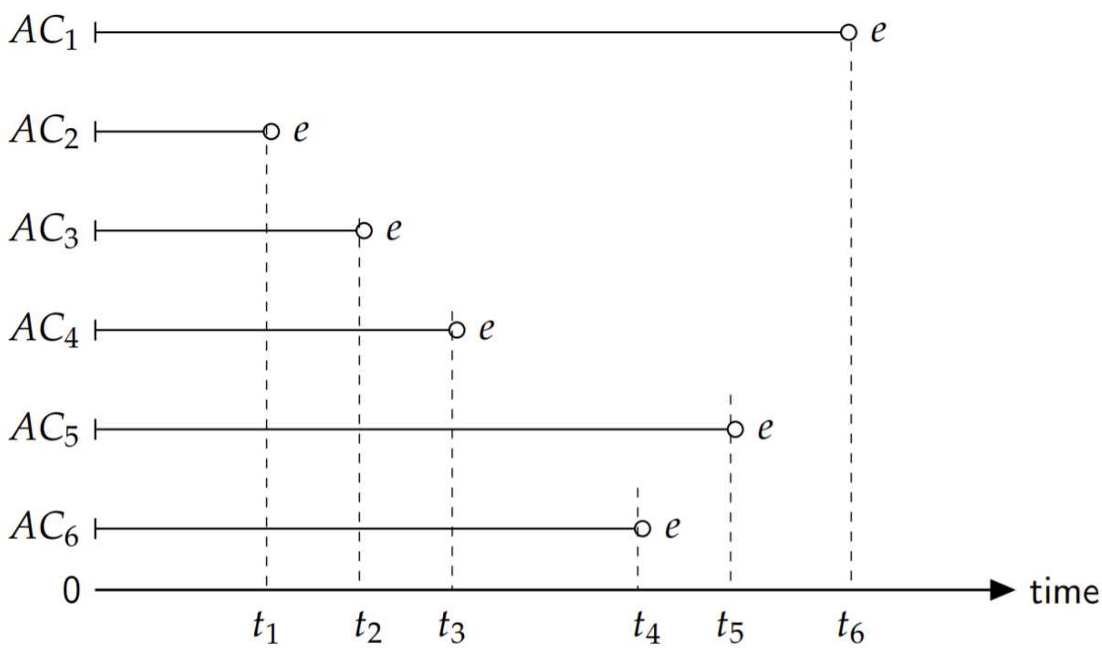

Figure 2.1: Censoring [32]

2.1.2 Survival Data and Censoring

A typical dataset in survival analysis consist of subjects, their survival time and other variables associated with the subjects depending on the objective of the study, the model and their availability. Fig 2.1 shows a small example of depicting a survival dataset, consisting of a few subjects and their times until the event of interest, e, $(t_1 - t_6)$. All the subject's observations are aligned in time and they begin at t = 0.

Many common statistical approaches cannot be used towards survival analysis because of characteristics of survival data, one of the problems being that some subjects will not experience the event, meaning some subjects will not experience the event till the end of the study. We therefore have two time classifications involved in a study of survival. A subject may experience the even in the course of the study and has an event time T. It is also possible that it survives until a time t, and is lost from the study, possibly having left the study. This insufficient information is called censoring. The three types of censoring are:

- **Right censoring** is when time until event is more than some known value. For example, the end of observation period without observing the event.

- **Left censoring** is when time until event is less than some familiar or known value.
- **Interval censoring** when when time until event is within a known interval but the exact value is not known.

Censoring is specified by an indicator feature i that is 1 for event occurrence and 0 for censoring. It is very important to handle the censored data. All types of censoring have been extensively studied in [11]. In the case of maintenance and our experiment, we deal mostly with right censoring. To summarize, regular regression models could be used in absence of censoring data but these may not be sufficient because of several reasons. For instance, the time to event is only positive (unlike regression). Furthermore, since no subject fails at T = 0, the distribution of the data is positively skewed, unlike normal.

2.1.3 Survival and Hazard Functions

A common method to approach survival analysis is the predicting hazards and obtaining survival times from it. We shall discuss the terms involved in Survival Analysis next. Time T indicates time to failure and it is a non-negative continuous random variable. The probability density function (PDF) and the cumulative density functions (CDF) are specified by $f(t)$ and $F(t)$, respectively.

$$F(t) = P(T < t) = \int_0^t p(z)\, dz. \tag{1}$$

The survival function $S(t)$, is the probability of an event occurring after time t, i.e. the probability that failure does not happen before time t,

$$S(t) = \int_t^\infty p(z)\, dz = 1 - F(t) = P(T \geq t). \tag{2}$$

It can be understood from the nature of the survival function that it is a non-increasing, usually decreasing function with respect to time and it always equals to 1 at the beginning of the observations. The graph of the survival function $S(t)$ is called the survival curve which begins at $S(t) = 1$ at time zero and as t increases to ∞, $S(t)$ decreases to 0.

Another metric is defined based on survival probability, referred to as hazard rate or hazard function. It is defined as the rate at which event is taking place, out of the surviving population at any given time t. Out of the people who survived at time t, hazard rate gives us the the rate of dying of those people. It is given by:

$$h(t) = \frac{f(t)}{S(t)}, \tag{3}$$

where $f(t)$ is defined as follows:

$$f(t) = lim_{dt \to 0} \ \frac{\Big(S(t) - S(t+dt)\Big)}{dt}. \tag{4}$$

Eq. 4 explains that the probability of a person who dies in a short interval $(t, t+dt)$ where the individual has already survived the time t. Naturally, the hazard function is always positive.

Because of the nature of data and challenges that comes with it, several parametric, non-parametric and semi-parametric approaches are employed to estimate survival time. Parametric approach assumes a distribution with particular type of parametric form. Non-Parametric approach does not assume any specific distribution and the method is quite simple to abridge the survival data and to make simple comparisons. It is difficult for these methods to deal with such situation for complex conditions though. Non-parametric methods are used more to analyse the survival data as it is less restricted then the parametric method. Semi-parametric method consists of models with both parametric and non-parametric parts to them and it focuses on effects of the covariates. We will briefly discuss Kaplan-Meier (a popular non-parametric model), parametric model and then popular Cox's PHM in details as it is the most broadly applied statistical survival analysis model for hazard predictions.

Kaplan-Meier estimate can be understood by looking at the evaluation of survival function using only uncensored observations for a moment. Recall that the survival function at time t can be evaluated as portion of subjects surviving longer than t. We can label the stated failure times in an increasing order, i.e., $t_1 \leq t_2 \leq t_3 \leq .... \leq t_n$.

The survival function at time t_i can be given by:

$$S(t_{\mathrm{i}}) = \frac{n-i}{n} = 1 - \frac{i}{n}, \tag{5}$$

where $(n - i)$ indicates the number of subjects not experiencing failure before t_{i}. Survival function is a step function with decreasing steps, starting from 1 and ending with 0 by the end. This formulation does not take into the account any of the censored data. If $t_1 \leq t_2 \leq t_3 \leq .... \leq t_{\mathrm{k}}$ are the times of k observed failures times in order, n_{i} are subjects that have not failed before time t_{i} and d_{i} indicating total observed failures at time t, the estimated survival function simply the portion of subjects surviving post time t and is evaluated as a product of conditional probabilities, given by:

$$S(t) = \prod_{i|t_{\mathrm{i}} \leq t} P(t_{\mathrm{i}} \leq T \leq t_{\mathrm{i+1}} | t_{\mathrm{i}} \leq T) = \prod_{i|t_{\mathrm{i}} \leq t} 1 - \frac{d_{\mathrm{i}}}{n_{\mathrm{i}}}. \tag{6}$$

The mean survival time can now be estimated as:

$$\hat{\mu} = \sum_{i=1}^{k} \hat{S}(t_{\mathrm{i\text{-}1}})(t_{\mathrm{i}} - t_{\mathrm{i\text{-}1}}) \qquad \text{where } t_0 = 0. \tag{7}$$

In parametric models, the hazard function is defined built on some particular distribution. One of the most common distribution used in literature is exponential distribution with parameter λ and constitute for the constant hazard function. A large λ means an inflated risk and dropped survival probability. On the other hand, small value for λ indicates lower risk and increased probability of survival.

The density and survival functions of T are then given as:

$$S(t) = \exp\,(-\lambda t), \tag{8}$$

$$f(t) = \lambda\ \exp\,(-\lambda t). \tag{9}$$

Hazard function:

$$h(t) = \frac{f(t)}{S(t)} = \lambda \quad \text{for } \ t \geq 0. \tag{10}$$

The correctness of the exponential model can be corroborated by plotting the $\log_e$ of survival function over t. If this results in a straight line through the origin, then an exponential distribution is indicated and λ can be approximated by the slope of the associated line. Further tests can be carried out to confirm if the exponential distribution is appropriate for the model.

Weibull distribution is another famously used distribution for parametric survival time modelling. It is a general form of exponential distribution and unlike former, it does not need a constant hazard rate. The hazard function is determined using two parameters of the Weibull distribution and is given as:

$$h(t) = \frac{\beta}{\eta^\beta} t^{\beta-1}, \tag{11}$$

where β is the shape parameter and η is the scale parameter. The resulting hazard function is decresing for $\beta < 1$ and vice versa otherwise. $\beta = 1$ gives a constant exponential distribution. Pdf for Weibull distribution is given by:

$$f(t) = \frac{\beta}{\eta^\beta} t^{\beta-1} e^{-(t/\eta)^\beta}. \tag{12}$$

The survival function is given by:

$$S(t) = e^{-(\frac{t}{\eta})^\beta}. \tag{13}$$

Just like the exponential distribution, there is an empirical test to confirm that the data follows a Weibull distribution.

2.1.4 Cox's PHM

The previously mentioned survival functions were based merely survival time only in the sense that they do not estimate relationships between survival time and the dataset features. What does that is a famous and commonly used semi- parametric model called Cox's PHM [12].

Assume a row vector of explanatory variable $Z = (x_1, x_2, ..., x_{\mathrm{k}})$ and let the hazard function at time t be indicated by $h(t; Z)$. The PHM is then given by:

$$h(t; Z) = h_0(t)\ \exp\,(\beta\,'Z), \tag{14}$$

where $h_0(t)$ is the baseline hazard function representing the effect of age. Second part is refereed to as hazard risk function which depends on the covariates. In other words, the elements of $\beta\,'Z$ will be of the form $\beta_1 x_1 + \beta_2 x_2 + ...\beta_{\mathrm{k}} x_{\mathrm{k}}$. It can be noted from the equation that the PHM assumes the variables are proportional to the underlying hazard function. The resulting conditional density function is given by:

$$f(t; Z) = h_0(t)\ \exp\,(\beta\,'Z)\ \exp\,\Big(-\exp\,(\beta\,'Z)\int_0^t h_0(x)\,dx\Big), \tag{15}$$

The corresponding survival function for t given Z is:

$$S(t; z) = S_0(t)\exp\,(\beta\,'Z), \tag{16}$$

where

$$S_0(t) = \exp\,\Big(-\int_0^t h_0(x)\,dx\Big). \tag{17}$$

Dividing both sides of the equation by $h_0(t)$ and taking $\log_{\mathrm{e}}$:

$$\ln\frac{h(t; Z)}{h_0(t)} = \beta\,'Z. \tag{18}$$

Maximum likelihood of β is estimated using Newton-Raphson methods and the determination of significant covariates is done by conducting likelihood ratio tests at each step of the model building process. After the significant variables have been identified, the consequent model defines the risk ratio, for covariates Z, as:

$$R = \exp\left(\ln \frac{h(t;Z)}{h_0(t)}\right) = \frac{h(t;Z)}{h_0(t)} = \exp\,(\beta\,'Z). \tag{19}$$

Based on 19, the risk ratio for any individual explanatory variable can be considered. The exponential of the estimated parameter coefficient, β_{i}, constitutes for the change in the hazard function with increment in variable x_{i} by one unit, given that the other variables do not change. For example, if $\exp(\beta_{\text{i}}) > 1$, then the hazard function increases, meaning that the survival probability decreases and vice versa.

In equation 14, two components need to be estimated, the regression coefficient β and the baseline hazard $h_0(t)$. β is be estimated by partial likelihood, which measures the goodness of fit of a statistical model for given values of unknown parameters to a sample of data. A standard likelihood function cannot be used for Cox's PHM since there is no knowledge about the baseline hazard $h_0(t)$, that is, it does not have any specific form. The censoring distribution is also not modelled and removed out of the formula. Cox's model likelihood function is therefore called partial likelihood function. Regression parameters β for the model is obtained by maximizing this likelihood function.

Assume that $t_{\text{i}} = t_1, t_2, ..., t_{\text{d}}$ are actual failure times with one failure at each time and $R(t_{(\text{i})})$ is a risk set consisting of subjects under observation. These subjects are not censored or have not failed by time $t_{\text{i}}, i = 1, 2, ..., d$. Then the full likelihood is given as:

$$L(\beta) = \prod_{i=1}^{k} L_i(\beta) = \prod_{k=1}^{n} \left\{ \frac{\exp(\beta\,'Z)}{\sum_{l \in R_{\text{i}}} \exp(\beta\,'Z)} \right\}^{\delta_{\text{i}}}, \tag{20}$$

where δ_{i} means that onely the contribution from death/failure times are considered and not from the right-censored times and l are all the individuals belonging to the risk set. Each term in the partial likelihood is the conditional probability of choosing a subject i to fail from the risk set, provided that the risk set at time t$_{\text{i}}$ and given one failure is to occur.

It is important to note that the baseline hazard depends on time t and not on the covariates Z in Cox's PHM. The hazard ratio $\exp(\beta\ 'Z)$ on the other hand depends on the covariates $Z =$

$(x_1, x_2, ..., x_\mathrm{k})$ and not on time t. But there are cases where if we measure some of the covariates over time, they may vary. For example, a patient's performance status, certain biomarkers, a operating conditions of a machine or sensory measurements of a system or process. This calls for considering a Cox model with time-dependent covariates as follows:

$$h(t; Z) = h_0(t)\ \exp\,(\beta\,' Z(t)). \tag{21}$$

The hazard at time t depends (only) on the value of the covariates at that time, i.e., $Z(t)$. The regression effect of Z is constant β over time. Note that each term in the partial likelihood is still the same and inference proceeds in the same manner as the conventional Cox model. The only difference is that the values of Z changes over time. A detailed implementation of time-varying Cox's PHM is presented in section 3.2.

The model assumes subject's log-risk of failure as a linear combination of the its covariates. This is the linear proportional hazards condition. It seems too primitive to assume that the log-risk function is linear and that might naturally not be the case with many cases. Therefore, a richer cluster of survival models is needed, the ones that are able to fit survival data with nonlinear log-risk functions. Some studies have argued that it is also difficult to used the Cox's PHM with high-dimensional and sparse data since it can not easily handle missing covariates. More flexible models can perhaps incorporate nonlinear relationships between the covariates and time-to-event along with their ability to handle high-dimensional data and work with missing information of covariates are becoming more necessary than before as we have advanced towards data explosion on one hand and increasing expansion of survival analysis in fields other than healthcare, especially in maintenance context. Machine Learning (ML)-based solutions pose themselves as promising techniques in addressing the above-mentioned shortcoming of statistical methods which is the main focus of this thesis. Prognostics and health management is among the most critical disciplines that employs the advancement of the great inter-dependency between data/signal processing and ML techniques to form a key enabling technology to provide an early warning of failure, in several domains ranging from manufacturing and industrial systems to transportation and aerospace. In the next section, the comprehensive review on ML-based survival analysis and especially Cox's PHM models that have dominated the trends in survival analysis research since last decade is provided in detail.

2.2 Review on ML-based Survival Analysis

Previously, a detailed discussion on several statistical approaches on survival analysis with a specific focus on Cox's PHM was provided. These approaches have been developed widely in the literature to model survival data and make predictions based on it. Due to limitations of statistical approaches in survival analysis, ML-based solutions are positioning themselves as the transformative set of technology of the century in industry 4.0 addressing key issues associated with conventional process monitoring in industries and physical asset maintenance management in dealing with multi-modal CM data. Due to great performance of ML-based solutions along with computer processing capabilities, there is a huge interest among researcher's to develop and apply different ML algorithms over the past few years. In this section, the focus is on ML-based solutions on survival model with special focus on Cox's PHM and provide a complete literature review. It gives a comprehensive review of advances in survival analysis dominated by ML and DL-based models. Before moving on, it is important to recall that the idea behind survival analysis as a different and important sub-field in statistics is because it provides an approach to deal with censored data. If not for this problem, conventional regression algorithms and developments in those would suffice since the problem at hand would be no different than a typical prediction problem. [13] provide a thorough review of commonly used statistical approaches along with ML approaches towards survival analysis. The next parts of this chapter follows the same structure and then discuss works that have been overlooked in the survey, especially the most recent DL-based models.

2.2.1 General Overview

To address some practical concerns that are faced when performing modelling with survival data, some works have adapted ML methods to solve survival analysis problems such that ML oriented researchers have developed a suite of sophisticated and effective algorithms that either complement or compete with the traditional statistical methods. Though the problem is important and relevant, its broad nature makes the research on this topic scattered across several different disciplines. Most of the reviews on the topic either focus solely on the statistical methods utilized and completely ignore the ML approaches or barely mention recent advances in ML in this field if they do. One of the earliest surveys [14] gives an overview of the statistical survival analysis methods and describes their application in criminology for predicting the time until recidivism. Most of the existing books on survival analysis [[15], [16]] introduce this topic from a traditional statistical perspective rather than a machine learning standpoint. Recently, however, this has begun to change and [18] and [19] both discussed applications in cancer prediction and included a comparison of several machine learning techniques.

Survival analysis methods can be broadly categorised into two main categories: (i) Statistical methods; and (ii) ML-based based methods. Both share a common goal, that is, making predictions of the survival time and estimate the survival probability at the estimated survival time. However, the former focus more on the distributions of time of event and the statistical properties of the parameter estimation, while the latter focuses more on the prediction of event happening by combining ML techniques with the power of traditional survival analysis methods. ML methods are usually applied to high-dimensional problems. ML methods offer more effective algorithms and approaches due to their ability to take advantage of recent developments in ML and optimization methods to learn the dependencies between covariates and survival times in different ways.

One of the first developments in survival analysis models began as extensions to Cox's PHM mainly due to its popularity and use in both academia and industry. With the continuing development of data collection techniques and detection methods, most real-world domains tend to accumulate high-dimensional data. In some cases, the number of variables in the given data might even exceed the number of instances. It is therefore challenging to build a good prediction model that takes into account all the features. It is possible that the model provides inaccurate results due to over-fitting problem as argued by [20]. This calls for using sparsity norms to select the most important features in high-dimensional data format given that many features would not be significant [21]. Identify relevant and significant features to the outcome variable then became a topic in itself and gave rise to a number of different penalty functions, including lasso and its different versions that were formulated for prediction models using sparse learning methods. Cox models include Lasso-Cox [22], Ridge-Cox ([23] and [24]), EN-Cox or Elastic net (EN) Cox ([26] and [27]) and Octagonal Shrinkage and Clustering Algorithm for Regression or OSCAR-Cox ([28] and [29]). While there are several algorithms that can be applied to empower sparse survival models to cope with high-dimensional data, none are applicable if mandatory covariates are to be necessarily required to be considered in the models. The CoxBoost [30] approach has therefore been proposed to incorporate mandatory covariates into the final model and it can also determine the coefficients like the Cox model [19].

One of the important extensions to Cox's Models to our context is the Time-varying Cox's Model which it is already discussed in the previous section. A time-dependent variable can typically be categorized into three types [15]: (i) Internal time-dependent variables, (ii) Ancillary time-dependent variables; and (iii) Defined time-dependent variables. It is important to note that the hazard ratio in the Time-varying Cox's model is a function of time, which implies that does not satisfy the PH assumption as in the Cox's PHM. The likelihood function, however, is constructed and optimized in the same way as Cox's PHM.

2.2.2 ML-Based Models

ML techniques can model non-linear relationships along with a great accuracy in predictions and this is evident from its applications and success in a extended practical domains. One of the main challenge facing ML methods however is handling censored data and the time evaluation. These techniques work great with huge dataset which might not be the case with for some problems [46]. Following part of this section provides a extensive discussion on some common ML methods and with a detailed review of the novel Nnet-survival model [37].

Survival Trees: Survival trees is a recursive partitioning method used for classification and regression and it is personalized to handle censored data. The idea centers around a splitting principle and the similar instances are placed in the same node based on event of interest. Different splitting criteria are used in different work. For instance, [34] measured the homogeneity and Hellinger distances between distribution functions for the splitting criterion, while [36] used exponential log-likelihood function based on the Cox model. Though similar in principle, a decision tree does not consider the interactions between the features for the splitting conditions [35]. and also its inability to consider censored data. Selection of the final tree is very important in building the model and methods like backward selection can be employed for selecting the best structure [42].

Bayesian Methods: Advances in Bayesian computation and their effectiveness has greatly developed its usage for survival analysis. The Bayes theorem provides a link between the posterior probability and the prior probability a change in probability values change before and after accounting for a certain event can be observed. Naive Bayes (NB) and Bayesian networks (BN) [43] are two common bayesian methods and both output the probability of the event. They are used for clinical prediction frequently ([44], and [45]). These methods have also have great interpretability and uncertainty reasoning [47]. Naıve Bayes is one of the most effective and simple prediction algorithms. One of the issues with the NB method it doesn't consider the possibility of a correlation between the features. Bayesian networks, however, can represent all the relationships between the variables, making it explicable. [50] proposed a Bayesian neural network framework to perform model selection for survival data using an automatic relevance determination procedure [51]. [52] proposed a Bayesian model averaging method for Cox's PHMs and it was also used to evaluate the Bayes factors.

Artificial Neural Networks: Artificial neural networks (ANN) have been extensively used for survival predictions. Three main methods are proposed in the literature for applying neural network methods to address survival analysis problems. Neural network survival analysis can be employed to predict survival time directly from the inputs without any feature extractions. Many extensions of Cox PHM were developed in the late 90s that preserved the characteristics of the Cox model. However, they were still not optimal since most of them still used linear output layer ([55], [62] and [62]). Many approaches like ([58], [59], [99], [61] and [50]) take the survival status as the outcome of the neural network as given by the survival and hazard functions. This is an indication of neural networks' capability to deal with censored data and it is for this reason that DL methods have gained huge attention to solve survival analysis problems in various fields. In healthcare for instance, deep correlational survival models and Convolutional Neural Networks (CNNs) are used to learn the compound relationships between features of patient data [63]. Several deep survival analysis approaches like [64] and [66] were proposed to assist with clinical decisions on the patients by estimating the disease risk and even offer custom treatment directions. Recurrent neural network (RNN) based approaches on the other hand have been used for survival analysis towards studying recurring events particularly in the context of user behavior modeling applications [68] and [69].

Support Vector Machines: Support Vector Machines (SVM), an efficient supervised learning method, typically used for classification and extended to regression, has proven its capability of adaption for survival analysis as well. One of the approaches is to apply SVM classification using constrained classification approach that imposes constraints on the model framing for two comparable instances so it goes on to maintain the required order [73]. It can be understood that this would be computationally expensive and not feasible for huge datasets. Another approach is using support vector regression (SVRs) since it adapts for censored cases by using an asymmetric loss function. [76] introduces an SVR based approach that combines the ranking and regression methods in the context of survival analysis. Another approach, Relevance Vector Machine (RVM) uses Bayesian approach by taking the prior over to a weight whose most probable value is iteratively estimated from the dataset.

Ensemble Learning: This method takes weighted votes from the results of different classifiers to predict the labels. It is possible to formulate an approach that can overcome the issues with methods like bagging [80] and random forests [81]

Many other ML-based methods like Bagging Survival Trees, Random Survival Forests, Boosting, Active Learning, Transfer Learning and Multitask Learning are extensively mentioned in DL-based survival analysis literature, some of them being [87], [88], [89] and [90]

DL-Based Models

Reference [91] propose strategies with Gaussian process models that makes it simple to visualize the corresponding effect of each input variable on the output prediction. It also argued that Neural Network based models for survival analysis might be difficult to interpret and though many gaussian approaches do not perform as good as the DL based models, they are more interpretable than the DL-based models. Deep survival analysis [66] was one of the first extensive work on DL in Survival Analysis. It proposes a hierarchical generative approach to survival analysis in Electronic Health Record EHR context. It models covariates and survival time in a Bayesian framework with a non-linear latent structure that captures relation between the covariates and the failure time. However, the model is limited to process with structured data only and it assumes a probability distribution.

Reference [64] proposed DeepSurv, a deep feed forward neural network model that predicts the effects of a patient's covariates on their hazard rate. This is basically a Cox's PHM with deep neural network for feature extraction upon the sample covariates, especially from image datasets. It replaces the exponential part βTx in the survival function with nonlinear deep fully connected network in traditional Cox model. This work was interesting especially due to carry out enhanced feature extraction method using DNN. However, it assumes a constant hazard rate over time and does not fully exploit the potential capacity of deep neural networks to learn complex representations of risk and in particular, to capture the time-varying influence of covariates on survival. Since the model uses batch gradient descent, it requires the entire training set to be used for each model update.

Reference [67] argues that the human designed features have limitations and proposed DeepConvSurv, a model that can represent more abstract information compared with hand-crafted features from the images, thus improving the survival prediction performance. It comprises of a CNN Model - replaces the risk function βTx in the Cox model with deep convolutional network. The model demonstrates advantages over DeepSurv. Reference [93] proposed DeepHit which avoids the problems inherent in assuming a form for the underlying stochastic process or a form for the relationship of covariates to the underlying stochastic process or any kind of time-invariance. The model make no assumptions about the underlying stochastic process and allows for the possibility that the relationship between covariates and risk(s) changes over time. It learns the (joint) distribution of survival times and events directly and handles competing risks smoothly; i.e. settings in which there is more than one possible event of interest. It employs a network architecture consisting of a single shared sub-network and a family of cause-specific sub-networks. However, the event probability estimation is regarded as a pointwise prediction problem and sequential patterns within neighboring time slices is ignored.

Reference [37] proposed an parametric approach that can organically deal with non-proportional hazards, called a discrete-time model. The model can be trained with minibatch gradient descent since only on the patients in current minibatch affect the likelihood/loss. The model also takes care of the time-varying nature of the input data. The approach proposes two primary advantages, for being taking care of the non-proportional hazards and second being its ability to fit huge datasets that do not fit in memory. Additionally, the model would also work effectively with image or text data due to its neural network based approach. However, since the follow up times are discretized, it results in an non-smooth survival curve as opposed to a non-discrete approach.

Reference [95] proposed Cox-nnet, which is a neural network whose output layer is a cox regression. This is basically a neural network extension of the Cox regression model. It is optimized for survival prediction from high throughput gene expression data. Additionally, they proposed usage of hidden node features as a new approach for dimensional reduction during survival data analysis. The model uses batch gradient descent and needs the whole training set for every update in the model. Reference [96] proposed a deep recurrent survival analysis model that combines DL for survival analysis to deal with censorship and for conditional probability prediction at a close data level. This was the first work that used auto-regressive model for survival. The model does not assumptions about the probability distribution of the event, thus flexibly modeling the survival probability function. It utilizes recurrent neural cell predicting the conditional rate of hazard and uses chain-rule to connect the predicted rate of hazard towards forecasting the probability of event time and survival rate estimation. The model estimates the true event ratio and survival rate through probability chain rule and demonstrates advantage over deephit and deepsurv.

Great end-to-end open source machine learning platforms like PyTorch, TensorFlow, Keras and Theano have made the application of neural networks to existing approaches simple and effective, resulting in development of a several approaches for time-to-event predictions since last decade. Several Cox partial likelihood based-models have been developed for time-to-event predictions, an important examples being [65] and discrete-time survival likelihood such as [93] and [37] amongst others. Next subsection will discuss Nnet-survival, a discrete-time approach outlined to be used with neural networks [37]. It is a state-of-the-art model due its flexibility in dealing with the non-proportional hazards, time-varying covariates and ease of training.

2.2.3 Nnet-survival

Nnet-survival is a discrete-time survival model formulated with neural networks and trained using maximum likelihood with mini-batch stochastic gradient descent (SGD). Using SGD enables rapid convergence and allows its implementation to huge datasets. The model is flexible to include time-varying co-variates. It has been executed in Keras. The authors first argue how adapting the Cox's PHM to neural networks as done is [55], [95] and [65] requires the whole datasets usage in each step of gradient descent. This is computationally expensive and could lead to getting stuck at loss function's local minima [97]. Nnet-survival on the other hand is a parametric model can be simply trained using SGD. Fixed intervals are assigned for follow-up times and the conditional hazard probability is estimated for each interval.

Model Formulation

Follow-up times are divided into n intervals such that $(t_1, t_2, \ldots, t_\mathrm{n})$ are upper limit times of the intervals. The probability of conditional hazard h_j is is simply the failure probability in interval j, provided that the subject survived until the time when the interval began. h_j can depends on the individual and neural network weights. The probability of an individual surviving till the end of interval j can be given as:

$$S_\mathrm{j} = \prod_{i=1}^{j}(1 - h_\mathrm{i}). \tag{22}$$

For an individual failing within the interval j (i.e., uncensored), likelihood is failure probability during interval j multiplied by the survival probability until $j - 1$:

$$lik = h_\mathrm{j} \prod_{i=1}^{j-1}(1 - h_\mathrm{i}), \tag{23}$$

or

$$loglik = \ln(h_\mathrm{j}) + \sum_{i=1}^{j-1}(1 - h_\mathrm{i}). \tag{24}$$

For an individual with time t_c (censored) that fails between the second half of $j-1$th interval and first half of jth interval (i.e., $\frac{1}{2}(t_{\text{j - 2}} + t_{\text{j - 1}}) \leq t_c < \frac{1}{2}(t_{\text{j - 1}} + t_j)$), the likelihood is the surviving probability in intervals 1 through $j-1$:

$$lik = \prod_{i=1}^{j-1}(1 - h_i), \tag{25}$$

or

$$\text{loglik} = \sum_{i=1}^{j-1}(1 - h_i). \tag{26}$$

A sum total of the log likelihoods for each individual gives the full log likelihood of the data. The objective is to maximize this likelihood, loss is set equal to the negative log likelihood and it is minimized either with SGD or mini-batch gradient descent. The conditional probability of surviving the interval i is given by:

$$1 - h_i = (1 - h_{\text{base}})^{\exp X\beta}, \tag{27}$$

where h_{base} represents the baseline hazard function whose values are evaluated by training a set of weights that forms the output layer with n dimensions (Eq. (27)). This approach is effective for smaller datasets and it is also easier to explain the predictions compared to other DL-based models.

The model is implemented in python using the Keras library with Tensorflow backend and a subject's survival probability is given by Eq. (22). An instance of neural network architecture using this model is shown in Fig. 2.2.

Since non-proportional hazards are considered, it can be used with SGD, trains faster and was consequently found to perform well with both simulated and real datasets. Since our experiment deals with time-varying dataset in maintenance, it should be interesting to explore the application of the model to such a dataset in maintenance context, which we shall see in the following chapter. This complete our discussion on DL-based solutions on survival analysis. Application of the DL-based solutions is not limited to the maintenance area. They have been well developed and widely used in quality domain for process monitoring which will be discussed next.

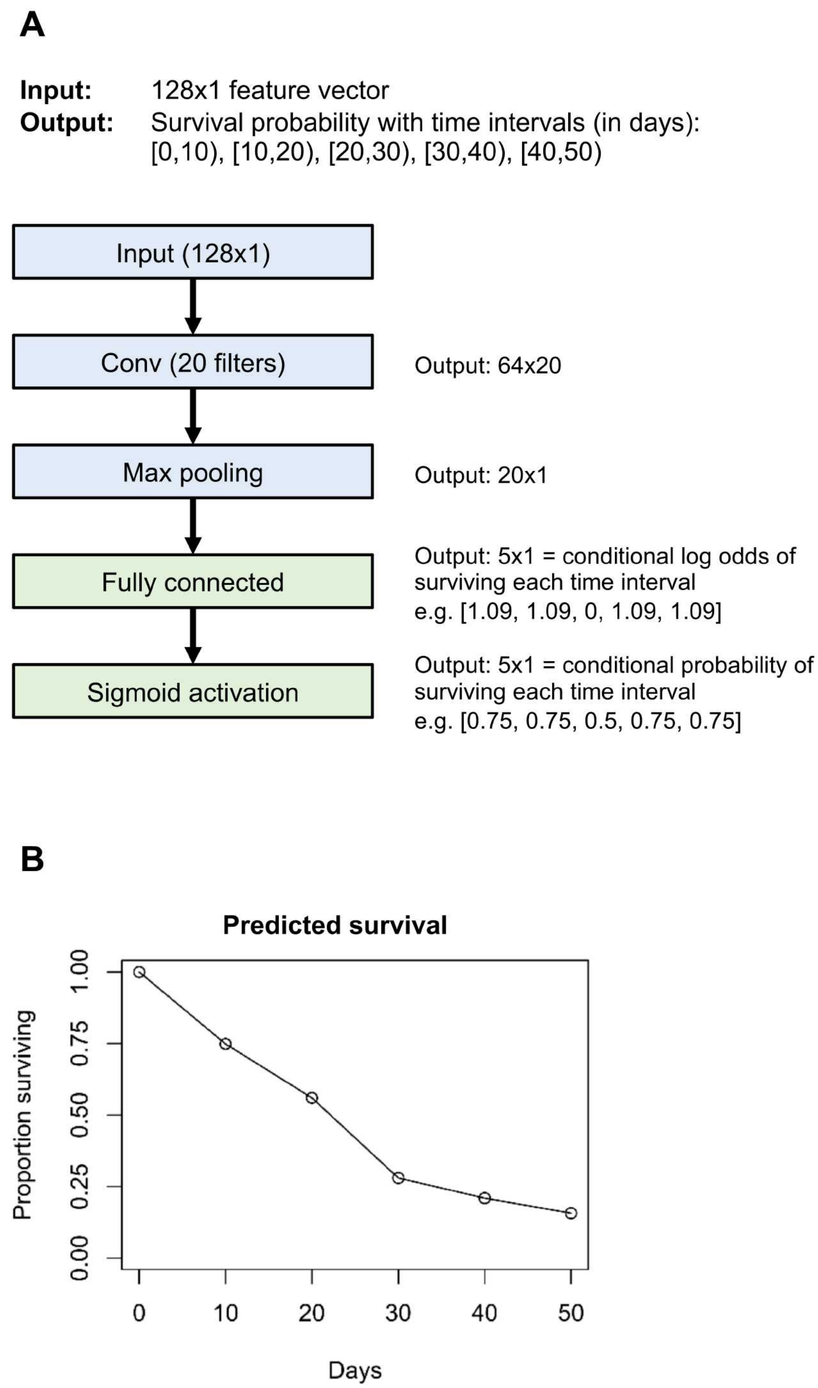

Figure 2.2: (A) Example neural network architecture and output for one individual (B). Layers in blue are unique to the example neural network; layers in green are common to all neural networks that use the "flexible" version of the survival model [37]

2.3 Process Monitoring Control Charts

It is of paramount importance that manufacturing systems operate at their maximum potential, especially in today's aggressive competitive business environment. Quality control is getting increasingly importance and directly affects customer satisfaction. Regular inspection and monitoring are therefore a vital part of any manufacturing process so that the final products quality is ensured.

There are several factors that needs to be monitored and controlled to ensure that no unexpected process occurrences take place. Without adequate and reliable process controls methods, such unexpected process occurrences can not be monitored, controlled, detected and eliminated. Process controls can range from simple manual actions to computer logic controllers, remote from the required action point, with supplemental instrumentation feedback systems [100]. Early process control breakthroughs came most frequently in the form of water control devices. Ktesibios of Alexandria is credited for inventing float valves to regulate water level of water clocks in the 3rd Century BC. In the 1st Century AD, Heron of Alexandria invented a water valve like the fill valve used in modern toilets [108]. The objective of process control is to keep key process-operating parameters within narrow bounds of the reference value(s) or setpoint(s). National Institute of Standards and Technology, an agency of the U.S. Department of Commerce, defines process control as the active changing of the process based on the results of process monitoring [109]. [101] describes process control as the organization of activities with the purpose of maintaining certain variables of a process (such as temperatures, pressures, concentrations, flow rates etc.) within specified limits and Changing said variables according to a preset program. Once the process monitoring tools have detected an out-of-control condition, an appropriate action will be conducted to bring the process back into the in-control condition. At a minimum, the process control strategy should address monitoring and control of critical process parameters. These are parameters whose variability have a direct impact on the physical, chemical, biological or microbiological property or characteristic of the product. These characteristics are more formally termed as critical quality attributes (CQAs).

There are several techniques used to monitor and control a production or a service process but one of the most important and relevant to us is SPC. It is a field in industrial statistics involves control charts, acceptance sampling, capability analysis and design of experiments. SPC plays a vital role in many industries. It can be defined as employment of several statistical tools towards process or production control. These tools and processes can help monitor the behaviour of a process, identify issues in them and find solutions for those issues. Control charts are one of the most important SPC tools applied typically to monitor the process over time and makes sure it is in within control limits.

American Society of Quality [110] describes control chart as a graph used to study how a process changes over time. Data are plotted in time order. A control chart contains lines determined from historical data: a central line for the average, and an upper and a lower line for the upper and lower control limit. The conclusions about whether the process variation is consistent (in control) or is unpredictable (out of control, affected by special causes of variation) can be drawn by comparing current data to mentioned lines. Controlling ongoing processes, predicting the expected range of outcomes, determining whether a process is stable, etc., are among the use cases of control chart. Originally developed by Walter Shewhart in the early 1920s, a control chart helps one record data and lets you see when an unusual event, such as a very high or low observation compared with ”typical” process performance, occurs. Control charts for variable data are used in pairs. The top chart monitors the average, or the centering of the distribution of data from the process. The bottom chart monitors the range, or the width of the distribution. If the data were shots in target practice, the average is where the shots are clustering, and the range is how tightly they are clustered. Control charts for attribute data are used singly.

Walter Shewhart proposed the fundamentals of the concepts in SPC and control charts in early 20th century but the researches in the area gained traction during the 1980s. This was primarily due to a sudden increased interest in quality due to the quality revolution caused by a global market that was now aggressively competitive. With the Shewhart chart control method, the state of control of the process at time t, is exclusively dependent on the most recent measurement from the process, in addition to the truthfulness of the estimates of the control from historical data. The Shewhart chart control method is activated when the most recent measurement has surpassed the control limit, while the EWMA method is responsive to smaller changes during the process. EWMA is a statistic that refers to the process of averaging data to commit less weight to data repeatedly in time. Shewhart charts are commonly used for the detection of shifts that are larger in size (i.e. $\geq 1.5\sigma$) with memory less property. To detect small-sized shifts ($\leq 1.5\sigma$) and to obtain benefit from memory of existing data, however, the EWMA and cumulative sum or CUSUM control charts are better. The state of control is dependent on the exponentially weighted average of all data, including the most recent measurement and all historical data. The EWMA statistic that is calculated as

$$EWMA_t = \lambda Y_t + (1 - \lambda) EWMA_{t-1}; \;\; for \;\; t = 1, 2, \ldots, n \, . \tag{28}$$

where

$EWMA_0$ is the mean of historical data (target)

Y_{t} is the observation at time t

n is the number of observations to be monitored including $EWMA_0$

$0 < \lambda \leq 1$ is a constant that determines the depth of memory of the $EWMA$

The parameter λ determines the rate at which "older" data enter into the calculation of the EWMA statistic. A value of $\lambda = 1$ implies that only the most recent measurement influences the EWMA (degrades to Shewhart chart). Thus, a large value of λ (closer to 1) gives more weight to recent data and less weight to older data; a small value of λ (closer to 0) gives more weight to older data. The value of λ is usually set between 0.2 and 0.3. The EWMA control charts are evaluated based on the average number of items sampled before the first out-of-control signal is reported, and developed based on non-transformed geometric, binomial and Bernoulli counts. By carefully selecting the smoothing constants, the EWMA control charts outperform the CUSUM control charts, the most efficient control charting techniques in existing literature for monitoring fraction non-conforming, in many cases. For the detection of small changes within population parameters, the EWMA control chart is considered to be more effective than the Shewhart chart. [102] provide a comprehensive review on control charts for high quality processes. For instance, a Bernoulli CUSUM chart is used for detecting a decrease in the proportion of nonconforming items. A markov chain model can be used to obtain the properties of the Bernoulli CUSUM chart.

Reference [106] presents a multivariate exponentially weighted moving average control chart that can be used to for detection of small shifts in multivariate SPC, whereas [107] explains the principles of the control chart based on the EWMA statistic, present some numerical examples to explore the properties and limitations of the EWMA chart. It also provides a comparison by computer simulation with Shewhart chart and the full Westgard multirule chart. Results show the EWMA chart is at least as good as the Westgard chart, atleast in terms of inaccuracy control. The ability of the EWMA chart to detect small shifts, however, recommends its use instead of the others for quality control. A further advantage of the EWMA chart is that all results are shown graphically and no additional rules for improved performance are needed (unlike the Shewhart chart). Thus, although the multirule system provides valuable data for quality-control assessment, the use of an EWMA graphical control chart provides the same or superior assessment data, not just a theoretically more satisfying concept. In general, the strength of EWMA charts is the detection of small increases in inaccuracy or imprecision. Due to great performance of EWMA chart, in this thheis we develop a hybrid model by incorporation EWMA control chart.

2.3.1 Control Charts for High-dimensional Data

Recent advancements in online sensing technologies and availability of several embedded sensors have led to exponential growth of data in different type such as waveform, i.e., time-series signals and high-dimensional data, i.e., images and videos. High-dimensional data sources play an important role in process monitoring and control due to providing thousands of rich data points. Efficient utilization of such high-dimensional data sources leads to highly accurate results in fault diagnostics. While the researches on control charts are tremendous, the application of SPC tools considering high-dimensional data sets has received less attention due to complex characteristics of such high-dimensional data including high variety, high-dimensionality, high velocity, and complex spatial/temporal structure. Application of control charts for detecting the change in the manufacturing process considering high-dimensional data is discussed in [122] among which the most popular control charts widely applied in machine vision systems are Hotelling T^2 chart [123], used for changes in the mean of multivariate Gaussian distributions and EWMA control chart, useful in detecting small shifts in the process mean based on samples taken from the process at given points of time ([124], [125]). Reference [105] present a survey on the control charts that have been proposed for use with image data in industry and in some medical-device applications and view image-based analysis and control charting as a very promising area of application within statistical quality control. Image monitoring is a natural extension of profile monitoring and though the work focuses more on image data from healthcare, a selective exploitation is definitely possible for maintenance application.

Reference [111] explored monitoring color transitions in plastic extrusion processes using a forecasting algorithm. They transformed the original RGB data into grayscale using the hue as the metric, where the hue was calculated as the average of the red, black, and blue intensities. Their methodology integrates traditional SPC tools for variable data with a forecasting system based on the EWMA statistic.

Similarly, Reference [112] integrated image processing technologies and multivariate SPC charts to design an automated visual inspection expert system, which could be used in mass production manufacturing systems as a part of the inspection process. As in many of the papers on this topic, their approach could be divided into a digital image processing step and a step where the control charts are applied. In the digital image processing stage, they suggested transforming the grayscale image into a binary image and then applying edge detection methods to further reduce the dimensionality of the data. Afterwards, the binary image was analyzed and the required dimensional quality characteristics were obtained and plotted on a multivariate control chart. They suggested the use of the chi-square control chart, Hotelling T2 [123] control chart, or a multivariate exponentially weighted moving average (MEWMA) control chart with the necessary control limits calculated based on Phase I data.

Existing process monitoring and control techniques in this context are usually consist of the following key steps: (i) Handcrafted feature design; (ii) Feature extraction/selection, and; (iii) Predictive classification/regression model construction. The right set of features are first needed, which are conventionally provided by some shallow algorithms such as support vector machines, naive Bayes, or logistic regression ([126], [127]). It is, however, difficult to determine the appropriate kind of features for a complex domain, at the same time, design of feature extraction selection methodologies is another tricky problem. The ability to automatically extract features in an end-to-end fashion proves indispensable in such cases. As a result, recently data-driven artificial intelligent (AI) solutions are being widely investigated as promising alternatives to hand-crafted (engineering) feature extraction for addressing the above-mentioned shortcomings and challenges. Specifically, DL-based solutions, as a breakthrough category of machine ML technologies, have attracted tremendous research interests in other research domains such as computer vision, natural language processing, signal processing and gaining more ground in quality control and maintenance domains leading to emergence of the AI. In particular, recently it has been shown that DL-based methods such as CNNs can achieve superior results when applied to the problem of classification on surface-quality control and inspection, which can be adapted to different products and industries ([128]- [134]). DL-based solution is positioning itself as the transformative technology of the century addressing key analytical challenges associated with conventional industrial process monitoring solutions in dealing with availability of such high-dimensional data. As competition is increasing in all industries and manufacturing sectors, industrial researchers, engineers, and the production managers are aiming to develop, design, and implement novel and innovative technologies to optimally utilize such rich high-dimensional data sources to further improve the quality of their products. In particular, AI is revolutionizing the business sector to better understand, control, and optimize the entire production process. In near future, the AI is expected to be the business critical.

Despite the fact that such DL-based methodologies have shown state-of-the-art performance beyond what can be achieved by previous state-of-the-art statistical methods, a key challenge for enabling the widespread deployment of deep neural networks (DNN) for embedded anomaly detection is the availability of large-scale training data sets. Consequently, high computational complexity requirement of DNNs for performing inference tasks and their large memory footprint cannot be overlooked. To tackle this identified problem, DL techniques coupled with statistical methods will be applied as a solution methodology. Generally speaking, in the quality control context, a sample of pre-determined size from a process is collected, a quality characteristic is measured based on the collected samples and plotted on a particular control chart. If any point exceeds the control limits, the process will be stopped, and investigation must be initiated to bring back the process into the in-control condition. In presence of the high-dimensional data, a similar path needs to be followed, however, instead of collecting few samples of the process, we have access to image sequences over time. Therefore, as a first step, we need to monitor these non-linear observations in order to detect quickly any change in the process. Therefore, in this thesis, we develop a hybrid model for process monitoring in presence of high-dimensional data. We formulate and view the aforementioned problem in the context of tracking a specified object or region of interest in the image or video sequence using Fast R-CNN along with multivariate EWMA control chart.

Chapter 3

Design of Maintenance Management System via DL-based Survival Analysis

Prognostics and health management is an important topic in industry for predicting state of assets to avoid downtime and failures. Within prognostics and health management context, Predictive Maintenance (PdM) is a great application of survival analysis since it consists in predicting when equipment failure will occur and therefore alerting the maintenance team to prevent that failure. Ability to model if and when a machine will break with greater accuracy is crucial for industrial and manufacturing businesses as it can help maintain a safe work environment by ensuring that machines are working properly, increase productivity by preventing unplanned reactive maintenance and minimizing downtime and optimize costs by removing the need for too many unnecessary checks or repairs of components, that is, preventative maintenance.

In the last few years, thanks to the use of Internet of Things (IoT) technologies, a plethora of data has been generated by various sensors on machines, mechanical and electrical components, such as temperature, vibration, voltage or pressure. This enables industries to generate high quality real time data that can be used to track and monitor the processes and use that information for prognosis. In this chapter, we apply different survival analysis models to a popular dataset for asset degradation modeling from NASA [[113], [115]]. It includes Run-to-Failure simulated data from turbo fan jet engines. Engine degradation simulation was carried out using C-MAPSS. Four different sets were simulated under different combinations of operational conditions and fault modes. The dataset was provided by the Prognostics CoE at NASA, Ames. This chapter begins with a detailed description of the dataset, its analysis and primary pre-processing. Two models will be then applied to the datasets, namely time-varying Cox's PHM and a feed forward neural network model (FFNN). The objective of both is to determine the survival curve of the engines with time. At the end of chapter, detailed discussion on the results are provided.

3.1 Description of dataset

The dataset includes multiple multivariate time series for different engines from a fleet of same engine types. The initial conditions of engines are unknown and the data also includes some sensor noise. The engines start with normal operation, developing faults over time and fail eventually (training set) whereas the series end before some time prior to failure in the test. The dataset was originally simulated to predict Remaining Useful Life (RUL) values, which is also provided for accuracy measurement but will not be used in our experiments. The dataset was used as challenge data for the 1st Prognostics and Health Management (PHM) data competition at PHM'08.

The dataset contains 26 different columns and every row is a data shot at every operational cycle. The columns correspond to Unit Number, Time in cycles, Operational Setting (OS1), Operational Setting (OS2), Operational Setting (OS3), Sensor Measurement SM1, Sensor Measurement (SM2), Sensor Measurement (SM3), ..., Sensor Measurement (SM25) and Sensor Measurement (SM26), respectively.

	FD001	FD002	FD003	FD004
Train trajectories	100	260	100	248
Test trajectories	100	259	100	249
Conditions	ONE (Sea Level)	6	ONE (Sea Level)	6
Fault Modes	ONE (HPC Deg)	ONE (HPC Deg)	TWO (HPC and Fan Deg)	TWO (HPC and Fan Deg)

Table 3.1: CMAPSS dataset [115]

	id	cycle	S1	S2	S3	SM1	SM2	SM3	SM4	SM5	...	SM12	SM13	SM14	SM15	SM16	SM17	SM18	SM19	SM20	SM21
42	32	122	0.0021	-0.0005	100.0	518.67	642.96	1597.20	1413.93	14.62	...	520.74	2388.17	8135.15	8.4707	0.03	393	2388	100.0	38.72	23.2185
39	82	162	0.0005	-0.0005	100.0	518.67	642.74	1590.54	1403.76	14.62	...	522.15	2388.01	8178.50	8.4336	0.03	395	2388	100.0	38.90	23.2789
64	61	23	0.0037	0.0004	100.0	518.67	641.81	1580.59	1404.09	14.62	...	523.13	2387.98	8145.57	8.3866	0.03	390	2388	100.0	38.93	23.4079
44	33	133	0.0018	0.0004	100.0	518.67	642.90	1587.62	1409.01	14.62	...	521.88	2388.07	8142.35	8.4537	0.03	394	2388	100.0	38.88	23.2703
40	59	202	-0.0007	0.0004	100.0	518.67	643.01	1599.80	1416.56	14.62	...	520.14	2388.14	8142.87	8.4696	0.03	395	2388	100.0	38.60	23.2109
03	38	188	-0.0005	-0.0003	100.0	518.67	643.33	1598.92	1425.12	14.62	...	519.77	2388.19	8204.76	8.5051	0.03	396	2388	100.0	38.67	23.0520
52	56	73	0.0011	0.0004	100.0	518.67	642.56	1591.76	1409.13	14.62	...	521.73	2388.15	8132.94	8.4544	0.03	394	2388	100.0	38.74	23.2034
33	84	49	0.0011	0.0003	100.0	518.67	642.63	1590.29	1410.48	14.62	...	521.24	2388.14	8130.37	8.4549	0.03	393	2388	100.0	38.83	23.1835
76	45	140	-0.0024	0.0002	100.0	518.67	642.94	1593.51	1416.68	14.62	...	520.62	2388.16	8161.29	8.5007	0.03	396	2388	100.0	38.63	23.1017
43	54	14	-0.0017	-0.0000	100.0	518.67	642.09	1576.44	1400.10	14.62	...	522.64	2388.02	8145.94	8.3683	0.03	391	2388	100.0	38.85	23.3487
47	53	113	0.0001	-0.0000	100.0	518.67	642.70	1585.47	1411.99	14.62	...	521.64	2388.07	8153.86	8.4713	0.03	393	2388	100.0	38.75	23.2904
09	71	180	0.0037	-0.0002	100.0	518.67	643.09	1598.48	1420.03	14.62	...	520.44	2388.08	8174.26	8.5302	0.03	397	2388	100.0	38.50	23.1638
45	68	14	-0.0004	-0.0004	100.0	518.67	642.85	1579.07	1406.61	14.62	...	522.04	2388.09	8142.12	8.4322	0.03	393	2388	100.0	38.97	23.2721
08	13	163	-0.0002	-0.0004	100.0	518.67	644.26	1610.89	1430.32	14.62	...	519.76	2388.16	8211.76	8.5477	0.03	397	2388	100.0	38.53	23.1291
88	21	21	0.0018	0.0002	100.0	518.67	643.19	1585.38	1402.15	14.62	...	521.69	2388.07	8128.46	8.4304	0.03	393	2388	100.0	38.89	23.3401

Figure 3.1: CMAPSS Dataset (FD001) [115]

3.1.1 Analysis and Data Pre-processing

Each of the pre-processing steps and experiments were conducted on all the four datasets mentioned in the Table 3.1. As evident, the S1, S2 and S3 correspond to the 3 operational settings (OS) and the SM1, SM2 until SM21 correspond to the 21 sensor measurements. A screenshot of the dataset FD001 converted into data frame is mentioned in Fig. 3.1.1. Similar patterns were found with insignificant differences between the four datasets in terms the covariate distributions as shown in Fig. 3.2, correlation between covariates as shown in Fig. 3.3, and the trends over time as shown in Fig. 3.4. For this reason, only FD001 dataset will be used to demonstrate the pre-processing steps and results. However, the experimental results are analyzed and documented for all the datasets separately.

Fig. 3.2 shows the distribution of the operations settings and sensor measurements. The spread in the distribution points towards the variance in the columns. It can be seen that some of the sensor measurements have a single value throughout. Also, from the Fig. 3.3 that describe the correlation between the values of different columns or covariates, it is found that there is a perfect correlation between sensor measurements 1 and 5, 1 and 10, 1 and 16, 5 and 10, 5 and 16 and 10 and 16. Also, sensor measurements 9 and 14 are found to have 0.963 correlation. It is also observed that these correlations hold for all engine together and a single-engine data as well. This is demonstrated by the similarity in two figures shown in 3.3. This was also found to be the case with all the 4 datasets.

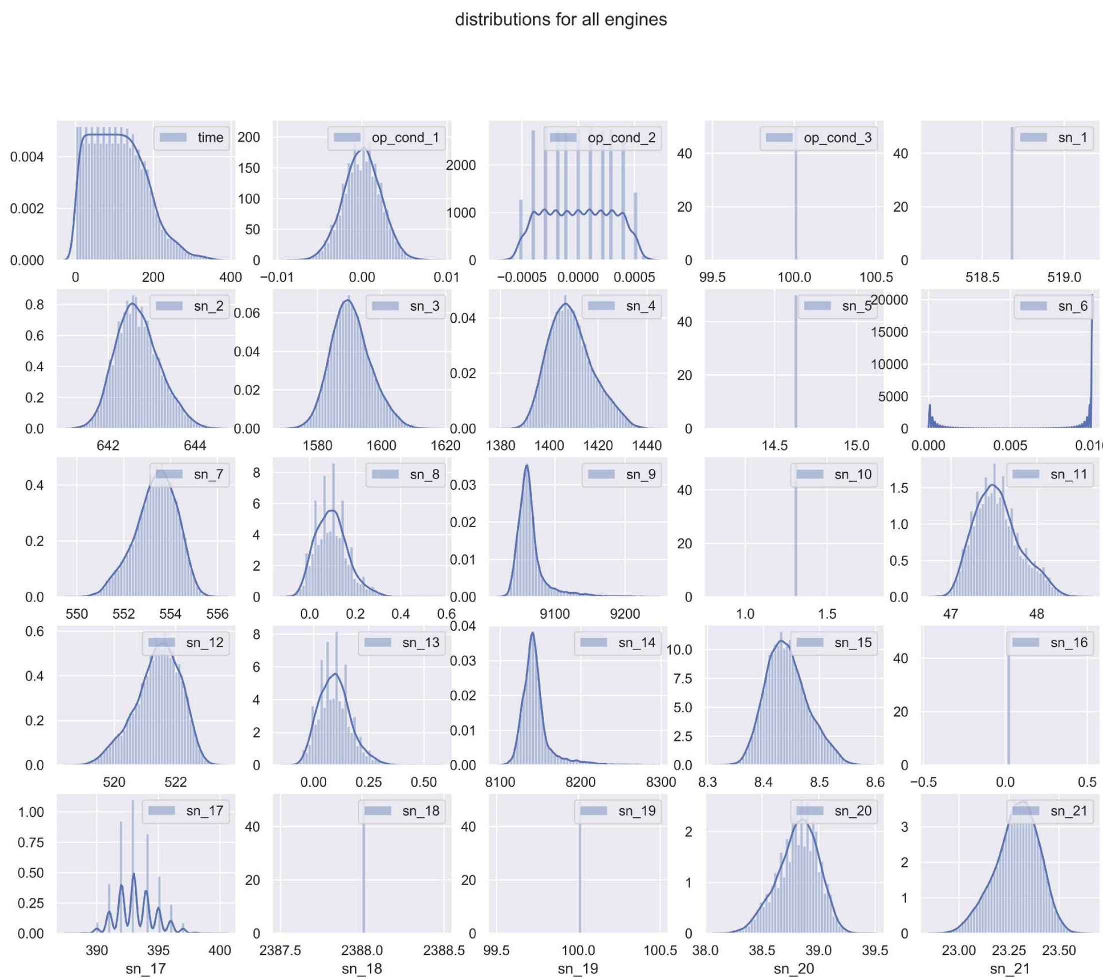

Figure 3.2: Distribution plot for all engines

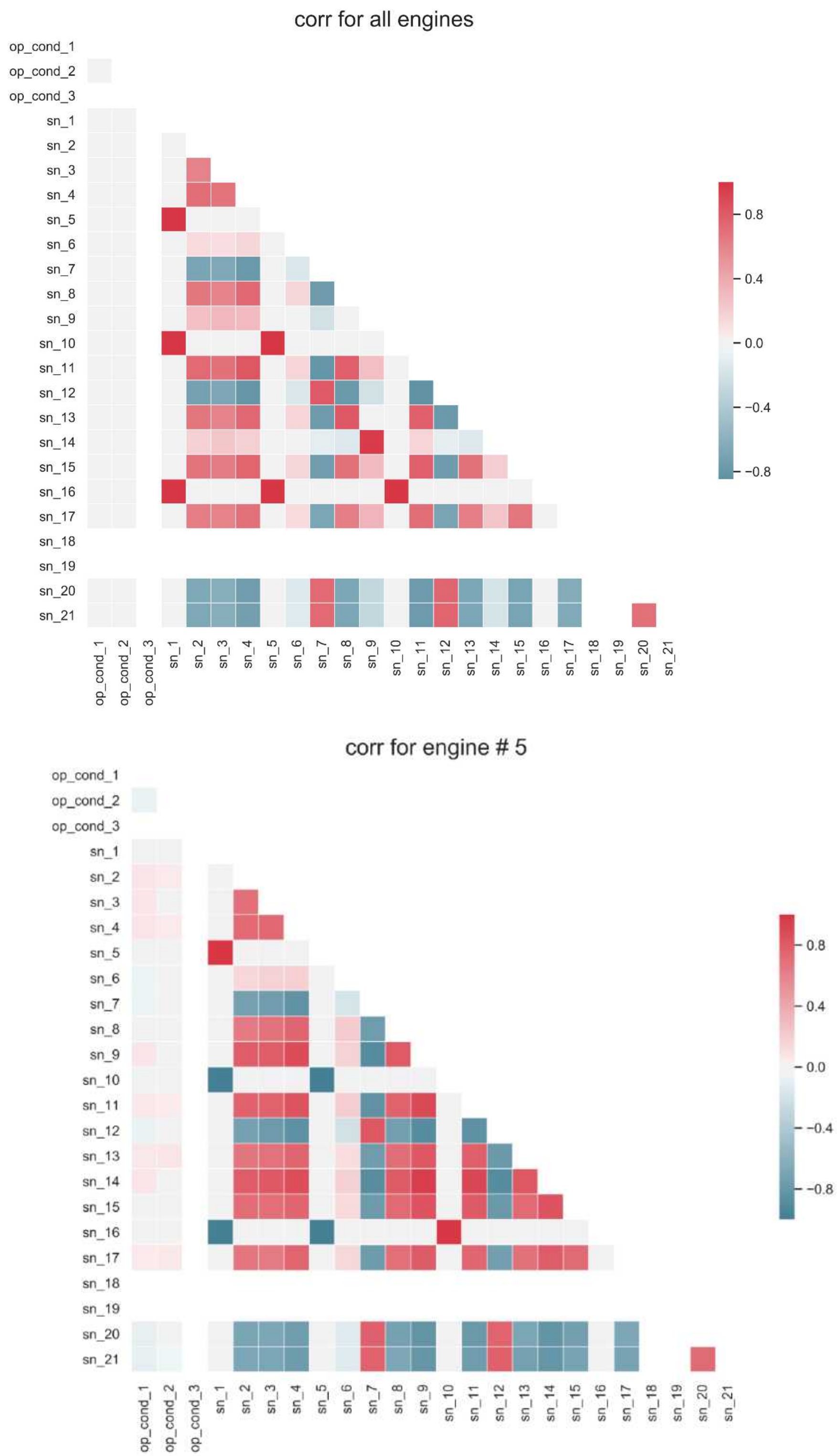

Figure 3.3: Correlation between covariates

This sets the premise for creating different experiments since it would make sense to take some columns off to alleviate redundancy in the dataset for better performance. Figure 3.4 shows the raw sensor data with the operational cycle for all engines. This gives an idea about the time variation of the sensors and variability compared to other sensors. The time-series plot shows that the first two columns, which corresponds to operating conditions 1 and 2, do not have particular trend toward the end life of the engine, whereas sensor measurements 9 and 14 are spread out hinting towards an engine specific trend. These measurements tend to decrease for some engines at the end of life for while others they tend to decrease.

3.1.2 Experimental Setup

Three experiments are conducted for each of the two models in section 3.2 and Section 3.3. In first experiment, all covariates are considered in the model. In the second experiment, some sensor mesurments are only considered in the model such that the sensor measurements that do not change over time, i.e., they have zero variance are taken off for experiment 2 towards the reduction of data dimensions. It can be observed that the distribution of almost all variables is single skewed Gaussian, operating condition 2 and sensor measurement 17 appear to be discrete variables. It was also observed that the observations holds when plotting all engines and when the plot is made for a specific engine, meaning all engine are very similar in their output responses. Besides the distribution plot, the analysis summary of FD001 suggests that operational setting 1 and 3, sensor measurements 1, 5, 10, 16, 18 and 19 have standard deviations less than 0.001. Therefore, these columns have been taken off from the dataset for experiment 2 since they do not seem to contribute significantly to the variations in the dataset. For experiment 3, only the significant covariates are considered. In particular, an ANOVA test is performed to find significant covariates such that the cut-off point of p-value = 0.05 is used to determine the significant factors . The only difference between the three experiments are the covarites as an input data considered in each model, as described below:

- Covariates for Experiment 1: All covariates with non zero standard deviations
- Covariates for Experiment 2: OS1, OS2, SM2, SM3, SM4, SM6, SM7, SM8, SM9, SM11, SM12, SM13, SM14, SM15, SM17, SM20 and SM21

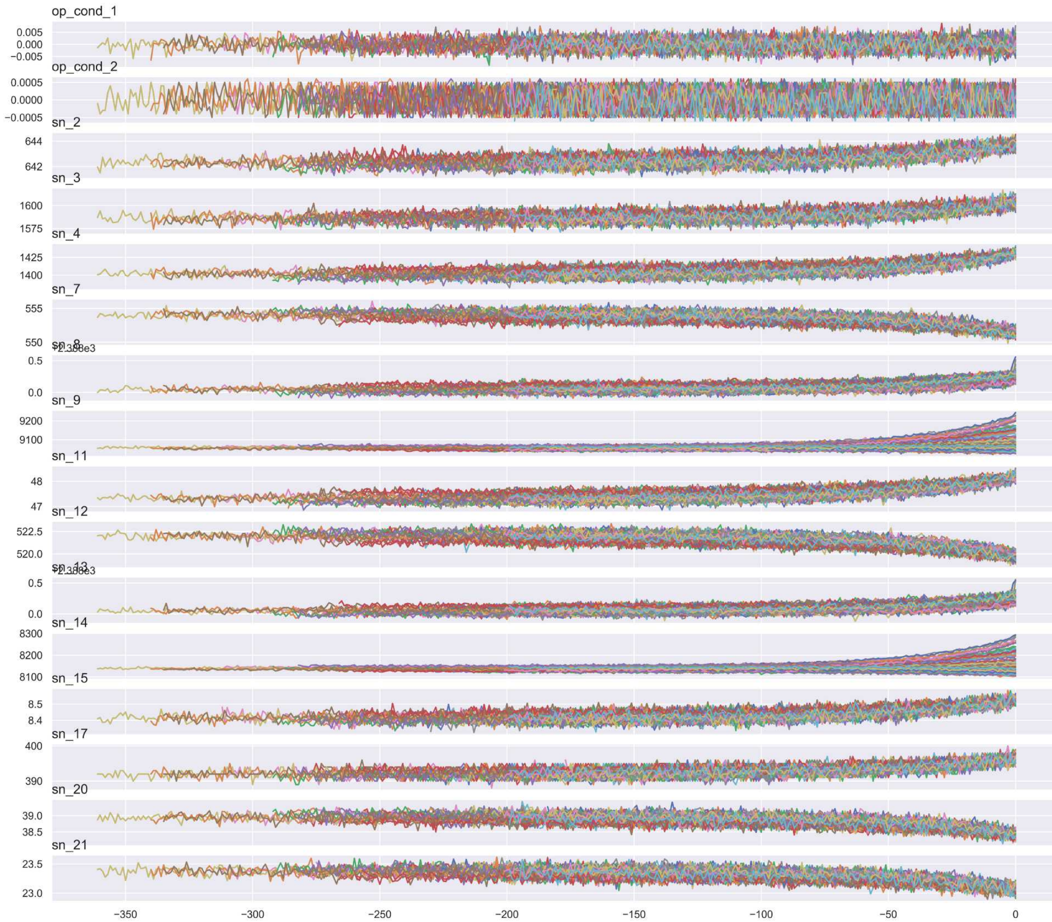

Figure 3.4: Time series plot for all engines for FD001. Each line corresponds to a different engine. Points on the right most are the last cycle for all engines

- Covariates for Experiment 3: OS1, OS2,SM2, SM4, SM7, SM11, SM12 and SM20

Time-varying cox's PHM is applied in the next section as mentioned in the previous chapter. Following that, a feed forward neural network non proportional cox time model proposed by [94] is applied to the same dataset. For the proposed models, we add an event column to the dataset which is necessary for the survival models. It indicates the time at which the event, in this case, the engine failure, occurs. Naturally, the last row in the sequence for every engine has a value of 1 and the rest of the rows has zero. It is also important to note that since the dataset is not missing any information, there is no censoring and we have all the measurements for all the engines. However, the event columns have been considered as they are and the application of the models to a dataset with censoring involved will only affect the results, not the framework and experimental structure.

3.2 Time-varying Cox's PHM model

Often an individual will have a covariate change over time. An example of this is hospital patients who enter the study and, at some future time, may receive a heart transplant. We would like to know the effect of the transplant, but we must be careful if we condition on whether they received the transplant. Consider that if patients needed to wait at least 1 year before getting a transplant, then everyone who dies before that year is considered as a non-transplant patient, and hence this would overestimate the hazard of not receiving a transplant. We can incorporate changes over time into our survival analysis by using a modification of the Cox model. The general mathematical description is as follows:

$$\underbrace{h(t|x)}_{\text{hazard}} = \overbrace{b_0(t)}^{\text{baseline}} \underbrace{\exp\left(Z(t,x)\right)}_{\text{partial hazard}}. \tag{29}$$

It has been established that the Non-Proportional Cox-Time model for time-to-event prediction is an extension of the cox's PHM with neural networks. After the pre-processing which included analysis of the covariate distributions, correlations between them and the comparative observation of covariates over time for the engines, the input dataset used for the experiment is a training set containing 20631 rows corresponding to 100 engines and 20 columns corresponding to a set of 1 operating condition, 15 sensor measurements, a column each corresponding to event and unit numbers and two columns for the start and end times for each data point (row). Lifelines is a complete survival analysis library, written in Python [114]. It has became popular in the recent years because of its easy installation, simple and intuitive API, its capability to handle right, left and interval censored data and the fact that it contains the most popular parametric, semi-parametric and non-parametric models. The model is implemented in lifelines as CoxTimeVaryingFitter. Lifelines requires that the dataset be in what is called the long format, that is, each row is one time point per subject. So each subject will have data in multiple rows. Any variables that don't change across time will have the same value in all the rows. This can also be stated as one row per state change, including an ID, the left and right time points. The start and stop times denote the boundaries, ID is the unique identifier per subject, and event denotes if the subject died (engine failure in our case) at the end of that period.

Once the dataset is in the correct orientation, the CoxTimeVaryingFitter is used to fit the model to the processed data. We fit the model to the dataset using fit(). While creating the class, the fitter takes default values of the level in the confidence intervals, alpha as 0.5, l1ratio of 0. We set the value of penalizer to 0.0001. Penalizer Attaches a penalty to the size of the coefficients during regression. This improves stability of the estimates and controls for high correlation between covariates. Fitting the Cox model to the data involves an iterative gradient descent. The model took 9, 5, 8 and 5 iterations iterations to converge for the FD001, FD002, FD003 and FD004, respectively. The results are discussed in Section 3.4

3.2.1 Prediction Limitations

It is important to note that in order to predict, we would need to know the covariates values beyond the observed times, but if we knew that, we would also know if the machine was still functional or not. However, it is still possible to compute the hazard values of subjects at known observations, the baseline cumulative hazard rate, and baseline survival function. So while CoxTimeVaryingFitter exposes prediction methods, there are logical limitations to what these predictions mean.

3.3 FFNN Non-Proportional Cox Time model

As discussed in the chapter 2, Cox model's proportionality is restrictive and using neural network to parameterize the relative risk function does not help. A way to circumvent this could be grouping the data based on a categorical covariate and then applying a layered version of Cox's model [116]. The model proposed by reference [94] is applied in this section. The primary reason is that this model considers both non proportionality and time dependency.

3.3.1 Loss

The model lets $Z(t, x)$ handle the time as a regular covariate and models interactions between time and other covariates like the conventional approaches. So the model is not a PHM anymore but still a relative risk model with an additional covariate. Similar to the Cox model in Eq. 20, the loss function is now described as:

$$loss = \frac{1}{n} \sum_{i:Di=1} \log \left\{ \sum\nolimits_{l \in R_{\mathrm{i}}} [\exp(\beta\,' Z(T_{\mathrm{i}}, x_{\mathrm{j}})) - \exp(\beta\,' Z(T_{\mathrm{i}}, x_{\mathrm{i}}))] \right\}. \tag{30}$$

Here i individuals have covariates x_{i} and durations T_i, $Z(T_{\mathrm{i}}, x_{\mathrm{i}})$ and β are covariate and parameter vector and R_{i} is the risk set. Note that R_{i} includes individuals with event times at T_{i}, so i is part of R_{i}. n denote the number of events in the dataset. Also the loss has the same T_i for both x_{i} and the $x_{\mathrm{j}}'s$. Also, the model uses a subset instead of the full risk set to calculate the loss to reduce the computational expenses for large datasets. It is important to note that his model would need a non-parametric baseline because $Z(t, x)$ would only model interactions between the covariates and time.

Predictions from relative risk can be obtained by estimating the survival function as in Eq. 16, with the addition that the exponential part is dependent on both x and t. The predictions in this and other models with non-proportional hazards are way more inefficient computationally. Cox-Time is therefore trained on continuous-time data though the predictions are discrete whose benefit is seen after fitting the model. In practice, the baseline on a random subset of the training data is estimated and the time grid resolution is controlled through the sample size to achieve discretization.

3.3.2 Dataset and Model Limitations

The dataset schema required for this model is to have the co-variates, duration and event columns. Though this model's implementation does not explicitly support time-dependent covariates, it can be achieved by partly conditional modeling [117]. The benefit of this approach is that it only requires pre-processing and no changes to the Cox-Time code. The idea of partly conditional modeling is that every time the covariates of an engine changes, it is treated as new individual considering the residual time. So, for an engine with event time t and a new set of covariates $x(s)$ at time s, we would consider this a new engine with event time $t - s$. We therefore use event as uncensored in each row and duration to failure instead of the cycle time. The new dataset will then contain many ”copies” of the individuals and the Cox-Time model can be fitted to this larger dataset. Survival predictions should works as before. The other suggestion by the authors is to include time as covariate. However, one of the issues with the dataset at hand is that there are more than 20,000 total rows and to include every change of every covariate in a new row will explode the data into something that will create computational and dimensional issues rather than resolving the time-dependent covariate problem. That is why, the approach taken was to consider each row as a new individual (engine). Meaning, instead of the timestamp column, a new column is created for event time for each row and since event times of the engines are known, the event time of each row for each engine is subtracted with the time stamp to get the event time of that individual row for the engine. So for engine E, if the event time is T_{e}, then there would be T_{e} rows for that engine, therefore, time stamps for that engine will be of the form $1, 2, 3, 4, 5, \ldots.., T_{\text{e-1}}, T_{\mathrm{e}}$. The event times for those very rows will be $T_{\mathrm{e}} - 1, T_{\mathrm{e}} - 2, T_{\mathrm{e}} - 3, T_{\mathrm{e}} - 4, T_{\mathrm{e}} - 5, \ldots..T_{\mathrm{e}} - T_{\text{e-1}}, 0$.

3.3.3 Training

We have the same training set as before and now we have a test set since we can use that to make predictions. The test set again has 13096 rows corresponding to 100 engines. Our training set is divided into 80% for training and 20% for validation. The model is a simple Multi Layer Perceptron with two hidden layers, ReLU activations, batch norm and dropout. The model contains a simple Multi-Layer Perceptron with two hidden layers, ReLU activations, batch norm and dropout. To train the model we use the Adam optimizer, but instead of choosing a learning rate, we will use the scheme proposed by Smith 2017 to find a suitable learning rate. However, this learning rate is often a little high, so we instead set it manually to 0.01

$labtrans$ was also set which connects the output nodes of the network the the label transform of the durations. This is only useful for prediction and does not affect the training procedure. In order to improve performance, We include the Early stopping callback to stop training when the validation loss stops improving. After training, this callback will also load the best performing model in terms

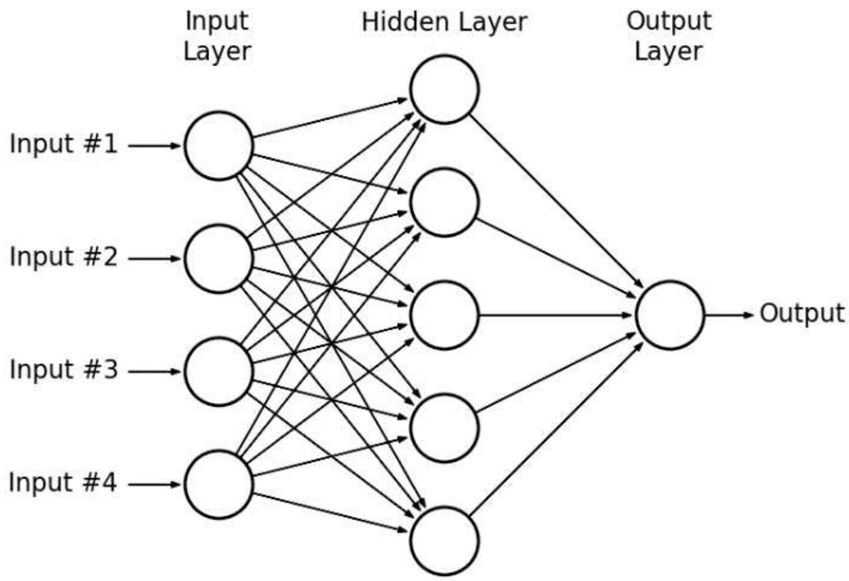

Figure 3.5: A hypothetical example of Multilayer Perceptron Network [119]

of validation loss. Since the model is based on Pytorch, the variables and the targets (durations and events) needs to be arrays of type 'float32'. We use labeltransform function within the model and we standardize the duration. With another function called TupleTree, we can easily repeat the validation dataset multiple times. This will be useful for reduce the variance of the validation loss.

3.3.4 Prediction

We run the model and in 512 epochs, we can get the partial log-likelihood and the value of -6.527. For evaluation, we first need to obtain survival estimates for the test set. This can be done with model.predictsurv function which returns an array of survival estimates, or with model.predictsurvdf which returns the survival estimates as a dataframe. However, as Cox-Time is semi-parametric, we first need to get the non-parametric baseline hazard estimates with compute_baseline_hazards. Note that for large datasets the sample argument can be used to estimate the baseline hazard on a subset. Note that because we set labtrans in CoxTime we get the correct time scale for our predictions.

It can be concluded that though this model was able to give prediction values unlike the traditional time-varying cox model, it did not perform very well looking at the RUL and the error values, even though the training concordance index was good. The main reason for this performance is partial conditioning. Our approach was that since all time dependency is captured by the covariates up to a given time s, i.e, $x(s)$, the predictions are conditioned on this time and this time then treated as the starting point (0) of survival predictions. For evaluation of the predictions though, the problem is that the multiple survival predictions $S(t|x(s), t > s)$ for different s are highly correlated, so considering them independent might be problematic.

3.3.5 Concordance Index

Though the dataset was developed ideally for a remaining useful life prediction, it was modified to be used for a survival analysis experiment, and it is for that reason that a typical survival analysis model metric be used to evaluate the model. It has been established that cox's PHM provides a way to estimate survival and cumulative hazard function in the presence of additional covariates. This is possible, because it assumes that a baseline hazard function exists and that covariates change the "risk" (hazard) only proportionally. In other words, it assumes that the ratio of the "risk" of experiencing an event of two patients remains constant over time. That is, after fitting Cox's PHM, $S(t)$ and $H(t)$ can be estimated easily. However, for other survival models that do not rely on the proportional hazards assumption, it is often impossible to estimate survival or cumulative hazard function. Their predictions are risk scores of arbitrary scale. If samples are ordered according to their predicted risk score (in ascending order), one obtains the sequence of events, as predicted by the model. Consequently, predictions are often evaluated by a measure of rank correlation between predicted risk scores and observed time points in the test data and that is one of the most widely used approaches, for both conventional and modern survival analysis models. In particular, Harrell's concordance index, computes the ratio of correctly ordered (concordant) pairs to comparable pairs and is the default performance metric when calling a survival model's score function.

Concordance Index, or C-index is a generalization of the area under the ROC curve (AUC) that can take into account censored data. It represents the global assessment of the model discrimination power, that is, the model's ability to correctly provide a reliable ranking of the survival times based on the individual risk scores. It can be computed with the following formula:

$$\text{C-index} = \frac{\sum_{i,j} 1_{T_j<T_i} \cdot 1_{\eta_j>\eta_i} \cdot \delta_j}{\sum_{i,j} 1_{T_j<T_i} \cdot \delta_j}, \tag{31}$$

where

- η_i is the risk score of a unit i
- $1_{T_j<T_i} = 1$ if $T_j < T_i$ else 0.
- $1_{\eta_j>\eta_i} = 1$ if $\eta_j > \eta_i$ else 0.

Similarly to the AUC, C-index $= 1$ corresponds to the best model prediction, and C-index $= 0.5$ represents a random prediction.

3.4 Results and Conclusion

For the cox time-varying model, by fitting the Weibull distribution on the dataset, the shape (β) and scale parameters (γ) are estimated which provide the baseline hazard function. The results are presented in Table 3.2. Also, in order to run experiment 3, we perform ANOVA test to find significant covariates for all four datasets which the summery of results are presented in Figs. 3.6-3.7. In particular, the first column "coef" represents the estimated coefficients, "exp(coef)" is just the exponential of column 1. Then "se(coef)" represents the standard errors of the estimates. The next columns have the lower and upper confidence intervals for the hazard coefficients. Then we have the z and p values, where the z value is simply a numerical measurement that describes a value's relationship to the mean of a group of values and the p-value is the probability of obtaining results at least as extreme as the observed results of a statistical hypothesis test, assuming that the null hypothesis is correct. A smaller p-value means that there is stronger evidence in favor of the alternative hypothesis. An initial experiment on both cases led to considering only the significant covariates in experiment 2. Whether or not the choice turned out to improvise the results and why is discussed in the next section.

The Table 3.3 describes the concordance indices of all eight sets of experiments performed, 2 on each of the 4 datasets. The figures 3.8 through 3.12 describe the training and validation losses for

model	lifelines.CoxTimeVaryingFitter
event col	'event'
penalizer	0.0001
number of subjects	100
number of periods	20631
number of events	100
partial log-likelihood	-121.28

	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	z	p	-log2(p)
OS1	0.06	1.07	57.85	-113.33	113.46	0.00	1.88e+49	0.00	1.00	0.00
SM2	1.26	3.51	0.44	0.39	2.13	1.47	8.39	2.83	<0.005	7.75
SM3	0.02	1.02	0.03	-0.04	0.08	0.96	1.08	0.74	0.46	1.12
SM4	0.13	1.14	0.03	0.07	0.19	1.07	1.21	4.27	<0.005	15.64
SM6	0.00	1.00	501.24	-982.40	982.40	0.00	inf	0.00	1.00	0.00
SM7	-0.79	0.46	0.31	-1.39	-0.18	0.25	0.83	-2.55	0.01	6.55
SM8	-0.76	0.47	3.19	-7.02	5.49	0.00	241.80	-0.24	0.81	0.30
SM9	-0.00	1.00	0.02	-0.03	0.03	0.97	1.03	-0.01	0.99	0.01
SM11	3.85	46.81	1.04	1.80	5.89	6.08	360.69	3.69	<0.005	12.13
SM12	-1.40	0.25	0.36	-2.11	-0.69	0.12	0.50	-3.87	<0.005	13.18
SM13	5.15	172.53	3.27	-1.25	11.56	0.29	1.04e+05	1.58	0.12	3.12
SM14	0.01	1.01	0.02	-0.02	0.05	0.98	1.05	0.73	0.47	1.10
SM15	4.94	139.68	6.33	-7.46	17.34	0.00	3.39e+07	0.78	0.43	1.20
SM17	0.24	1.28	0.14	-0.02	0.51	0.98	1.66	1.81	0.07	3.82
SM20	-4.40	0.01	1.30	-6.94	-1.85	0.00	0.16	-3.39	<0.005	10.49
SM21	-3.58	0.03	1.98	-7.47	0.31	0.00	1.37	-1.80	0.07	3.81

Figure 3.6: FD001 time-varying Cox's PHM results summary

model	lifelines.CoxTimeVaryingFitter
event col	'event'
penalizer	0.0001
number of subjects	249
number of periods	61249
number of events	248
partial log-likelihood	-905.05
time fit was run	2020-08-27 21:38:07 UTC

	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	z	p	-log2(p)
OS1	0.05	1.05	0.02	0.01	0.09	1.01	1.09	2.47	0.01	6.19
SM2	-0.02	0.98	0.01	-0.04	-0.00	0.96	1.00	-2.35	0.02	5.72
SM3	0.01	1.01	0.00	0.00	0.02	1.00	1.02	2.74	0.01	7.34
SM4	0.02	1.02	0.00	0.01	0.02	1.01	1.02	6.25	<0.005	31.15
SM6	-0.09	0.92	0.07	-0.22	0.05	0.80	1.05	-1.27	0.20	2.30
SM7	-0.00	1.00	0.00	-0.01	0.00	0.99	1.00	-1.08	0.28	1.83
SM8	-0.01	0.99	0.00	-0.02	-0.01	0.98	0.99	-4.54	<0.005	17.47
SM9	-0.00	1.00	0.00	-0.00	0.00	1.00	1.00	-1.00	0.32	1.66
SM11	0.78	2.18	0.11	0.57	0.99	1.76	2.69	7.21	<0.005	40.75
SM12	-0.00	1.00	0.00	-0.01	0.00	0.99	1.00	-1.05	0.29	1.77
SM13	-0.01	0.99	0.00	-0.02	-0.01	0.98	0.99	-4.43	<0.005	16.69
SM14	0.02	1.02	0.00	0.01	0.02	1.01	1.02	7.48	<0.005	43.61
SM15	1.82	6.15	0.36	1.12	2.52	3.06	12.37	5.09	<0.005	21.44
SM17	0.04	1.04	0.01	0.01	0.06	1.01	1.07	2.77	0.01	7.50
SM20	-0.06	0.94	0.04	-0.14	0.01	0.87	1.01	-1.69	0.09	3.47
SM21	-0.11	0.89	0.06	-0.24	0.01	0.79	1.01	-1.81	0.07	3.84

Figure 3.7: FD004 time-varying Cox's PHM results summary

	β	γ	Partial Log Likelihood		
			Experiment 1	Experiment 2	Experiment 3
FD001	4.409	225.026	-226.11	-120.56	-142.31
FD002	4.388	225.664	-1011.67	-1024.34	-1101.31
FD003	2.929	276.818	-83.52	-91.44	-113.31
FD004	3.471	272.870	-889.40	-905.05	-987.62

Table 3.2: Time-varying Cox model experiments Summary

	Experiment 1		Experiment 2		Experiment 3	
	PLL	CI	PLL	CI	PLL	CI
FD001	-6.5313	0.7159	-6.5275	0.7071	-6.7342	0.6669
FD002	-7.7000	0.7037	-7.6793	0.6880	-7.7869	0.6666
FD003	-6.6518	0.7388	-6.6483	0.7335	-6.7924	0.7097
FD004	-7.8401	0.7078	-7.8436	0.6352	-7.9512	0.6787

Table 3.3: Partial Log Likelihood and Concordance Indices for FFNN Cox Time model

each of the eight experiment, the combined survival curve for engines 1 through 5 in the test set and the separate survival curves for different engines. As mentioned above, it should be noted that each

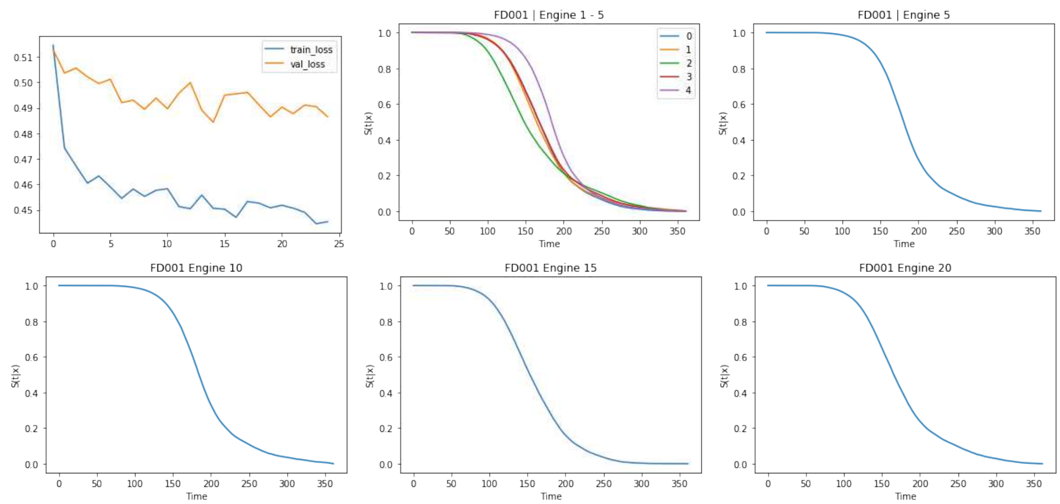

Figure 3.8: FD001 Experiment 1

row in this model is considered an engine as opposed to the actual engines. The graphs however, are good indication of how and at what time point does the probability drops reaches zero.

To restate, the idea behind the experiments were to see if the models used in healthcare can be applied to survival analyses on maintenance perspective. Before discussion on the results, it should be noted that the result metrics are different for both the models, owing to the limitations with the dataset and models and the interpretation of predictions in case of time-varying cox model. The model gives the final equation, using which, the survival probability can be estimated at any given point in future ideally. However, as mentioned, the prediction capability of the time-varying model depends on the information of the covariates at those times and if the covariates are known based on sensor measurements then it will be known whether or not the engine failure occurred. Similarly, with the FFNN non proportional cox time model, the model does not explicitly model the time-dependent covariates and they have to be forced. Also, the performance metric, the concordance index does not explicitly provide us with the survival probability of the subjects. There are no benchmarks to compare our results with given this dataset was originally created for RUL predictions and all the available experiments are done with the objective of RUL predictions. However, the concordance indices for the non-proportional neural network model range between 0.63 to 0.73 for the datasets. It can be noted that the CIs for experiment 3 are generally less than those of experiment one. One of the reasons for this could be over-fitting caused due to reduction of data to

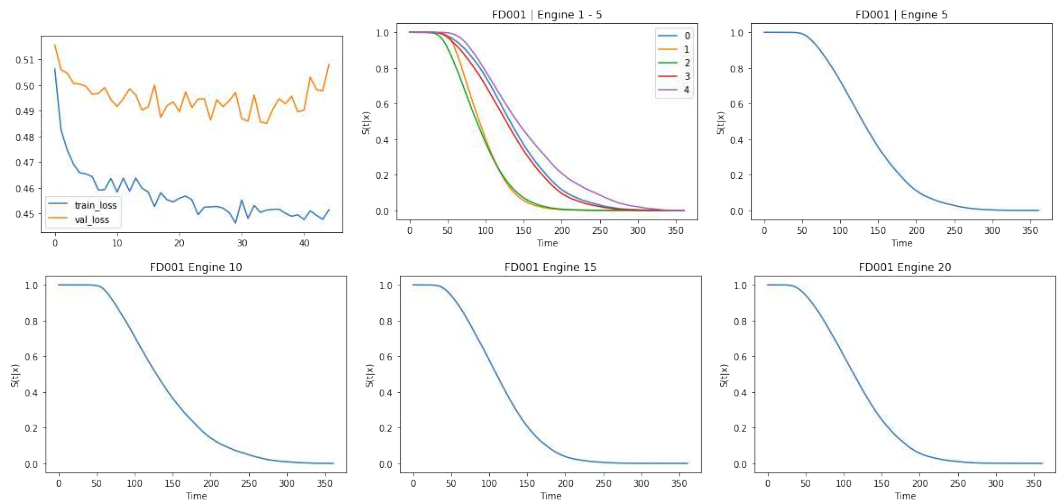

Figure 3.9: FD001 Experiment 2

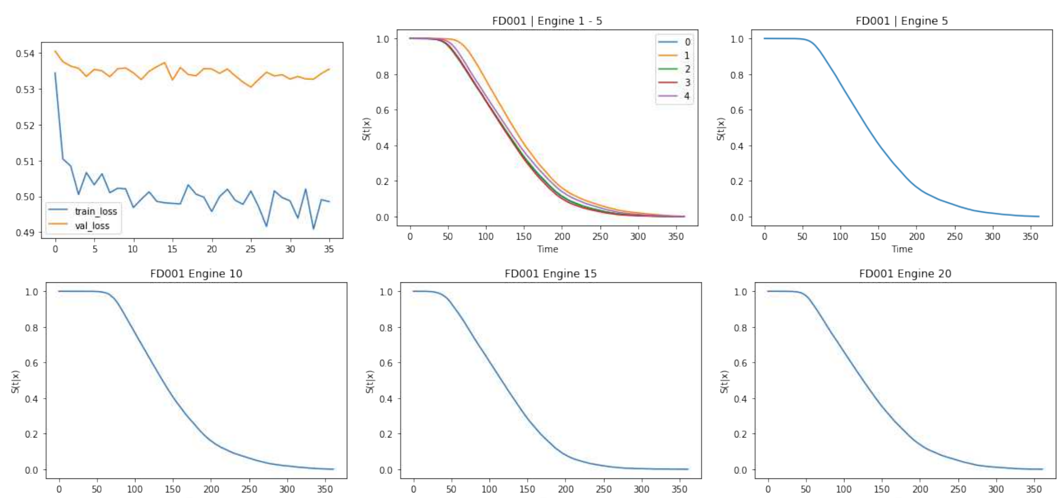

Figure 3.10: FD001 Experiment 3

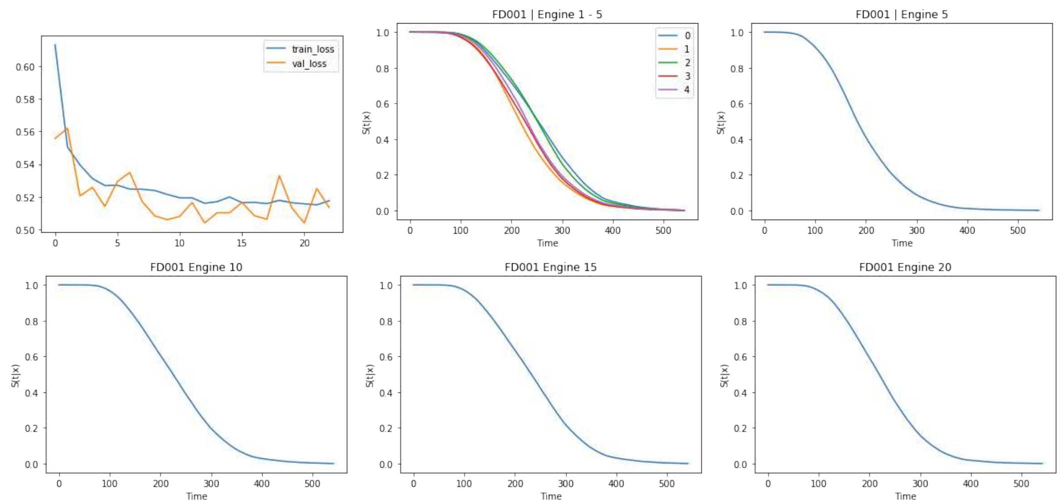

Figure 3.11: FD004 Experiment 1

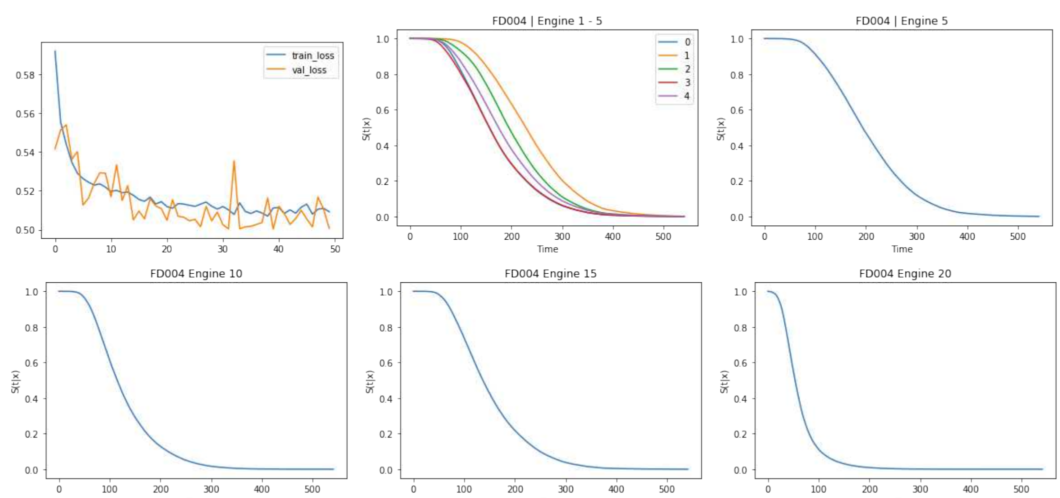

Figure 3.12: FD004 Experiment 2

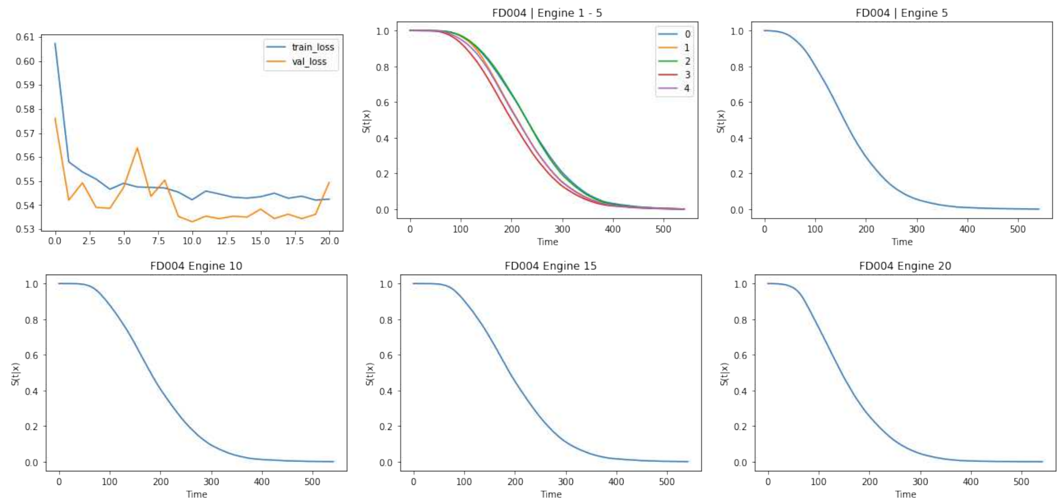

Figure 3.13: FD004 Experiment 3

only 7 covariates. This was also found to be the case with the Cox time-varying model. The partial log likelihood were reduced in each of the four cases in the second experiment compared to one. This follows from the knowledge that generally, more data means better results given there are no dimensionality issues. Whereas it was important to take out the columns with insignificant variance, it was more suited to keep all the other covariates.

Both the models eventually provide survival probability of the engines with time with some variations, the time-varying cox model with the equation and the neural network model with the $S(t)$ curve over time. The survival probabilities for some engines considering different experiments and two of the four data sets are presented in Figs. 3.8-3.13. As the graphs show the survival probability is equal to one when the engines starts operation at time 0, which indicates they are operating in healthy condition, however, due to degradation effect, the survival probability is decreasing over time as expected. The great advantageous of the proposed model is that there is no need to extract features from data, and raw signals are fed into model.

It is worth mentioning that though the CI is typically used in the medical literature to quantify the capacity of the estimated risk score in discriminating among subjects with different event times, it is a good measure to our experiment and can be further used to for a maintenance decision support system that would typically suggest optimal policy approaches towards maintenance of the engines.

The curve gives an insight about the engine health over time and provides with the necessary information needed towards formulating a maintenance policy combined with the CI in the way that a generic policy can be determined using a Markov decision process where one of the inputs for feedback is the CI. To summarize, this chapter included a case study of attempts at using existing survival analysis models for the maintenance field. Though used extensively in the healthcare sector, there are several challenges in using the existing survival models in maintenance perspective, as evident and discussed though out the chapter. Though it is possible to overcome most of the challenges, it is important to understand the objectives and then select a model based on that along with the dataset and associated information at hand.

Chapter 4

Process Monitoring Via DL-based Techniques

Advancements in online sensing technologies and wireless networking reshaped the competitive landscape of manufacturing systems lead to exponential growth of data. Among various type of data, in particular, high-dimensional data sources such as images and videos play an important role in process monitoring to providing thousands of rich data points. Efficient utilization of such high-dimensional data sources leads to highly accurate results in fault diagnostics. While the researches on SPC tools is tremendous, the application of SPC tools considering high-dimensional data sets has received less attention due to complex characteristics of such high-dimensional data. In this regard, this thesis tries to address this gap by designing and developing a hybrid model based on DL algorithms and SPC models to monitor the manufacturing process in present of high-dimensional data. In particular, we first apply a Fast R-CNN model in order to monitor the image sequences over time. Then, some statistical features are derived and plotted on the multivariate EWMA control chart. The effectiveness of proposed hybrid model is illustrated through a numerical example.

The reminder of this chapter is organized as follows: Section 4.1 deals with the details on problem description. Solution methodology is presented in Section 4.2. The effectiveness of the proposed hybrid model is illustrated through the numerical example presented in Section 4.3. Finally, section 4.4 concludes the chapter.

4.1 Problem Description

The primary objective of this chapter is to develop an effective and dynamic processing methodology to monitor the stochastic processes in manufacturing and industries based on the measurements collected in the form of image/video sequences. In this regard, we investigate potential approaches to utilize and incorporate the high-dimensional image/video sequences into SPC framework.

Generally speaking a manufacturing process can be either in the following two states, namely, (i) in-control and (ii) out-of-control state. Process can make a transition from in-control state to out-of-control state when an assignable cause is occurred at random time. The main focus of the control charts is to detect the shift in the process as soon as possible in order to minimize the number of defective items. Therefore, fast detection plays a significant role in quality control context. Developing an efficient control chart for fast detection in presence of image observation is considered a challenging task.

An image is usually represented as a matrix of picture elements, i.e., pixels, where row and column of this matrix represent spatial coordinates of a given pixel. The value of pixel at any location is known as intensity and its value depends on whether the image is black and white (binary), grayscale, or colored format. In a binary image, each pixel can only have an intensity value of either 0 (black) or 1 (white). In case of a grayscale image, a pixel can take any integer value between 0 (black) and 255 (white) for an 8-bit image. Finally when the image is in colored format, it is represented by a tensor of three individual components corresponding to red, green, and blue (RGB) channels with values between 0 and 255 for each component. Fig. 4.1 illustrates a color image representation in three wavelength channels (RGB). In all the aforementioned three cases, an image can be viewed as a collection of multivariate, spatially distributed observations with high-dimensionality that depends on the number of pixels and the type of image.

Therefore, we need to address the following key research challenges: (i) How to analyze massive amount of non-linear, non-Gaussian, high-dimensional sensory data? (ii) How to incorporate the huge amount of monitoring data into process model for control and diagnosis purposes? In this regard, we propose a hybrid estimation algorithm based on DL and SPC tool which will be discussed

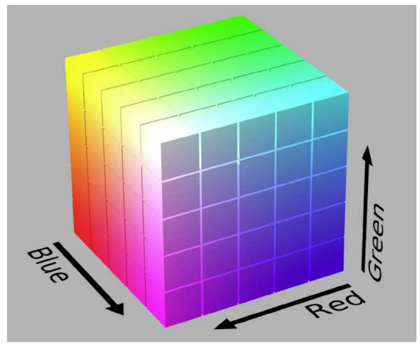

Figure 4.1: RGB Image Matrix

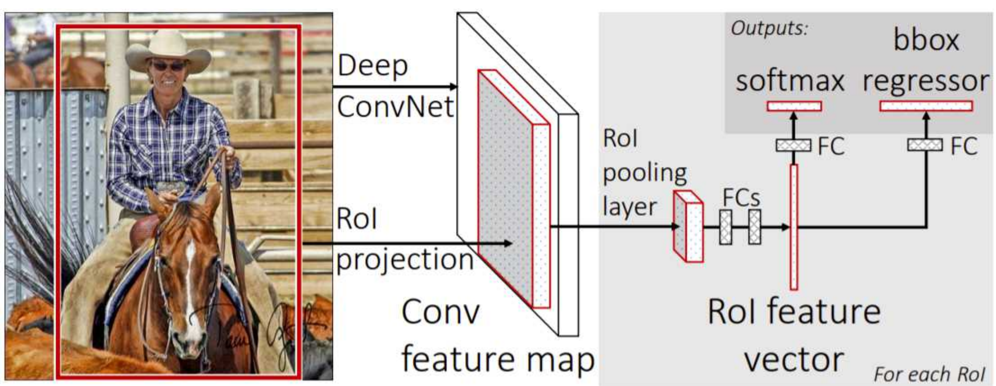

Figure 4.2: Fast R-CNN architecture [135]

in details in the next section.

4.2 Solution Methodology

As the first step, the camera is placed alongside the process to capture image sequences from the process in real-time. Then, DL-based solution is applied in order to track the region of interest over time. Then, some statistical features are derived and plotted on the EWMA control chart. The proposed methodology is based on the following two general steps: In the first step, we are dealing with non-linear observations and trying to track the observations over time using DL-based object detection technique, and; in the second step, we try to monitor the process and incorporate the observations into the process state using EWMA control chart. The detailed analysis of each step will be discussed in the following sub-sections.

4.2.1 Object Tracking and Detection

As a first step, we should utilize the information from the color image data to be able to monitor and control the process over time. At each time epoch, a sample of N images are collected. We need to monitor these non-linear observations in order to detect quickly any change in the process. For this purpose, we formulate and view the aforementioned problem in the context of tracking a specified object or region of interest in the image or video sequence.

Compared to image classification, object detection is a more challenging task that requires more complex methods to solve. Fast R-CNN is a Convolutional Neural Network model that proposes a single-stage training algorithm that jointly learns to classify object proposals and refine their spatial locations. This method can train a deep detection network with less time and great accuracy [135].

Figure 4.2 shows an input image and multiple ROIs which are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers. The network has two output vectors per RoI, namely; (i) Softmax probabilities and (ii) Per-class bounding-box regression offsets. The architecture is trained end-to-end with a multi-task loss.

The sample video generated using MATLAB is fed through Fast R-CNN ([135]) using Tensorflow object detection API ([136]) which is an open source framework built on top of TensorFlow ([137]). The generated video was split into images divided into 180 images such that 160 images are used to train the network and remaining is used for testing purpose. The images are fed through Fast R-CNN and the resulting accuracy of the model is achieved by 0.994.

4.2.2 Monitoring using EWMA Control Chart

In order to control the process, the results obtained from previous step are used to develop a statistical method for process monitoring and control. For this purpose, the EWMA control chart is designed to monitor the process which is first introduced by Roberts (1959) and the author showed that the EWMA is useful for detecting small shifts in the mean of a process. An EWMA control scheme is based on the following statistic:

$$Z_i = \lambda Y_i + (1 - Z_{i-1}), \quad 0 < \lambda \leq 1, \tag{32}$$

where

- Y_i is the individually observed values from the process.
- λ is a constant that determines the depth of memory of the EWMA chart.

The starting value, i.e., Z_0 is often considered as the target value. The process is considered to be out of control whenever the value of Z_i falls outside the range of the upper and lower control limits denoted by UCL and LCL which are calculated as follows:

$$\begin{aligned} UCL &= Z_0 + L\sigma_Z \\ LCL &= Z_0 - L\sigma_Z, \end{aligned} \tag{33}$$

such that

$$\sigma_Z = \frac{\lambda}{2 - \lambda}\sigma_Y^2, \tag{34}$$

where σ_Y^2 is the variance of Y_i observations. The center line for the control chart is the target value or Z_0. This completes our discussion on the solution methodology. Next, we further elaborate on details through different simulation test cases.

4.3 Experimental Results

In this section, we evaluate the performance of the proposed hybrid algorithm based on a set of simulated data as proof-of-concept and through several different experimental scenarios. As a first step, we define the process model and then, the observation model is developed. Finally, the proposed solution methodology is applied.

Modeling Process: Similar to [138], we generate the video which is consist of image sequences over time. The time index is set to 350. The in-control images which are $300 \times 300 \times 3$ RGB-color images consist of a red circle on a black background. Three sets of parameters are used to randomly generate the in-control circles as follows:

- Two-dimensional center location, i.e.,
  $(X_t, Y_t) \sim \mathcal{N}(\mu_{0x}, \mu_{0y} = 150, \sigma_{0x}^2 = \sigma_{0y}^2 = 1)$.
- RGB numbers follow Normal distribution as follows:
  $R_0 \sim \mathcal{N}(100, 0.01); G_0 \sim \mathcal{N}(20, 0.01); B_0 \sim \mathcal{N}(15, 0.01)$.
- In-control circle radius set to $r_0 = 50$.

We consider different out-of-control scenarios as follows:

- Scenario I: In this scenario we assume the mean of the circle is shifted to a new value, i.e., $(\mu_{1x}, \mu_{1y}) = (\mu_{0x}, \mu_{0y}) + \delta_1\sigma_{0x}^2 + \delta_2\sigma_{0y}^2$ where δ_1 and δ_2 represent the magnitude of the quality shift in both horizontal and vertical directions, respectively.
- Scenario II: In this scenario, we assume the shape of the circle is changed, which is created by increasing the radius of the circle, i.e., $r_1 = r_0 + \delta_r$.

All in-control and out-of-control scenarios are illustrated in Fig. 3.

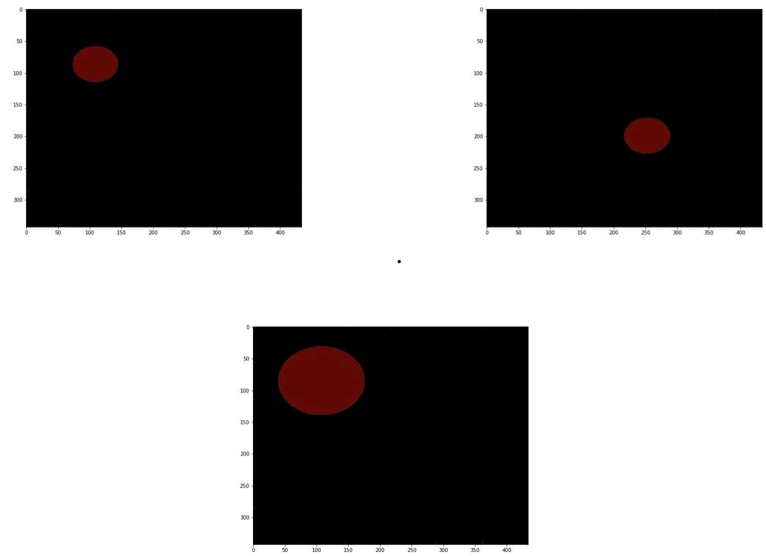

Figure 4.3: A set of 3 different experimental scenarios: (a) In-control image; (b) Out-of-control image: Location change. (c) Out-of-control image: Shape change.

Proposed Methodology: Based on the DL-based solution explained in section 4.2.1, the bounding box is created in each frame which can be seen in Fig 4.4. Then, the coordinates of the resulting bounding boxes are derived in each time frame which is the center of the bounding box denoted by $\bar{X}_k^b$ and $\bar{Y}_k^b$. Then, the output error vector is calculated which represent the distance value between the reference (in-control) and the mean of bounding box center in each time frame. The error vector is calculated as follows:

$$\boldsymbol{e}_{d_k} = \sqrt{(\mu_{0x} - \bar{X}_k^b)^2 + (\mu_{0y} - \bar{Y}_k^b)^2}. \tag{35}$$

The error is used as the input statistics (observations) to the EWMA control chart. The EWMA statistic is plotted on the control chart. As soon as any point exceeds the control limits, the process should be stopped and investigation should be triggered. Similar to the location change, we need to find the error for shape change scenario. The error is defined as the difference between the radius of the in-control image, i.e., reference radius r_0 and the radius of bounding box denoted by r_d which

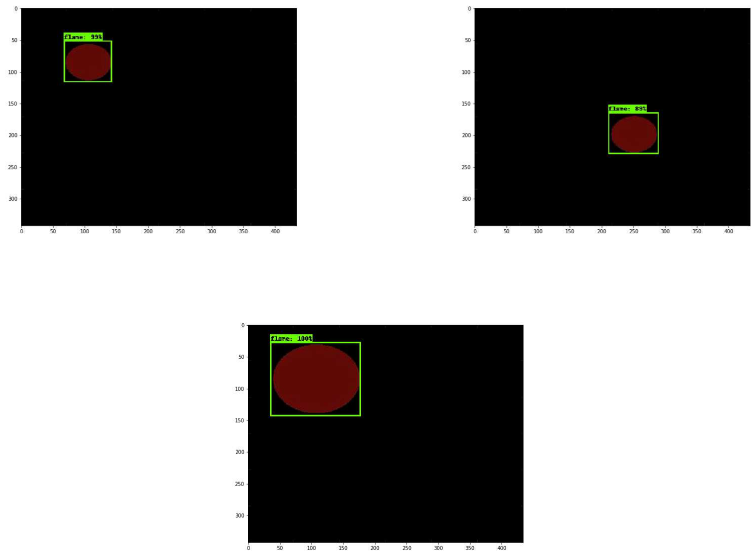

Figure 4.4: Resulting Bounding Boxes and accuracy for the four scenarios: (a) In-control image; (b) Out-of-control image: Location change. (c) Out-of-control image: Shape change.

is calculated as follow:

$$e_{r_k} = \sqrt{\left(r_0 - r_k^d\right)^2}. \tag{36}$$

For constructing EWMA chart, the training data includes 180 in-control images. In total, four control charts are developed for monitoring the location and shape, respectively. In constructing the EWMA charts, the value of λ is set to 0.4. In order to test the effectiveness of the control chart for detecting different shift sizes, we consider the following shift sizes in scenario 1:

$$\begin{aligned} \text{Small shift size}: &\ \delta_1 = 3, \delta_2 = 3, \\ \text{Moderate shift size}: &\ \delta_1 = 5, \delta_2 = 5, \\ \text{Big shift size}: &\ \delta_1 = 7, \delta_2 = 7. \end{aligned} \tag{37}$$

In particular, in image sequence 1 to 47, we consider the smallest shift size. The results are shown in Figs 4.5 and 4.6

As the results indicate, the EWMA control chart can easily detect the out-of-control condition

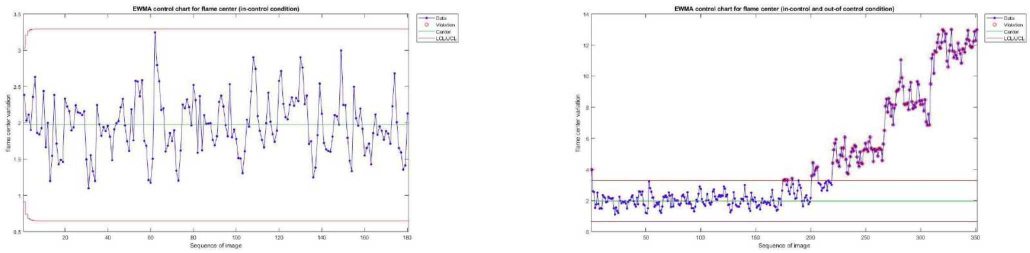

Figure 4.5: EWMA Control Chart for Location Monitoring: Train and Test

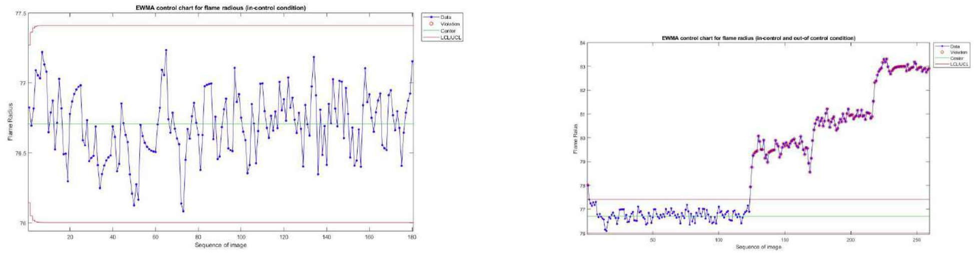

Figure 4.6: EWMA Control Chart for Redius Monitoring: Train and Test

as soon as it happens. In particular, for location change at image sequence 180, the process goes to out-of-control state and EWAM chart can detect this shift in the process as soon as it happens. Also, for the shape change, in the simulation, we set the out-of-control condition to image sequence of 124 and EWMA control chart can detect this shift as soon as it happens.

4.4 Conclusion

In this chapter, a novel hybrid approach based on DL and SPC is presented for on-line monitoring of the manufacturing and industrial processes in order to detect a shift in the process quickly which is a step forward contribution in quality control context. The raw image sequences are directly used as the input of the proposed frameworks. The proposed algorithm has several main advantageous in comparison to conventional DL and SPC techniques for process monitoring. First of all, it doesn't require feature extraction when high-dimensional data is available. The networks can be fed from raw images. Also, the combination of SPC tools and Fast R-CNN will significantly decrease the computational complexity. In addition, it doesn't require huge number of images in order to train the network. For future research, it is interesting to see the effectiveness of the proposed method on real data set from the manufacturing processes. Also, applying more advanced control charts such as Bayesian control chart will be considered.

Chapter 5

Summary and Future Research Directions

The chapter concludes the thesis with a list of important contributions made in the dissertation and some proposed directions for future work.

5.1 Summary of Thesis Contributions

In this thesis, we tried to develop a framework towards CBM in prognostic and diagnostic context through two case studies. The overall objective was to utilize specific DL based survival analysis and object tracking models on high dimensional data. In the first case, one conventional and one DL based survival analysis model was applied on the NASA Turbofan engine failure dataset. The results from the conventional time varying model typically provides us with the equation that can be used to predict $S(t)$. However, the interpretation of what prediction means have certain logical limitations when it comes to time varying models. For the Non-Proportional Cox-Time model, the results are survival curves, which are informative. However, the existing Cox-Time implementation does not explicitly support time-dependent covariates. Achieving this by partly conditional modeling has an advantage in the sense that the dataset only requires some pre-processing and no changes to the actual model code. However, the challenge on the other hand with this approach is each row is then considered a new subject. Though this does not affect the prediction capability of the model, the issue is with the scale of a new dataset created post partial conditional modelling. The combined process of applying both models does provide an insight into usability of survival analysis models towards condition monitoring data with the purpose of formulating a CBM policy.

In the second case, we try to alleviate the gap in works on application of SPC tools that consider high dimensional datasets in diagnostic context. by designing and developing a hybrid model based on DL algorithms and SPC models to monitor the manufacturing process in present of high-dimensional data. The application of Fast R-CNN and plotting EWMA control charts to monitor the image sequences over time provides a novel framework in the context of quality control. The challenges facing application of this approach for a real dataset would often not be related to the framework itself, rather it would be approaching the object detection algorithm that is effectively able to detect, track and measure the changes through the visual dataset with high sensitivity.

5.2 Future Research

Three potential future research directions are discussed below to further improve the proposed contributions made throughout this thesis. The first two, focuses on each case study separately the third discusses the elements that would apply to both and the possibility of a holistic or balanced CBM policy:

- Prognostic:

  An important direction to go in case of maintenance management via DL based survival analysis will be to modify the existing non proportional cox time model code to incorporate the time dependent covariates instead of partial conditioning. This would not only reduce the fulmination of the existing dataset by accounting for every change in the covariates but also save a significant amount of redundancy and computation time while doing so. Using other DL based survival analysis models, especially the ones that incorporate time dependent to the same dataset is another important direction that would contribute parallel to this work as it will provide an objective comparison of different DL models for the same dataset.

- Diagnostic:

The work done on integrated DL and SPC model for real time monitoring of manufacturing process is composed of three main components: The dataset, the DL model for tracking and the SPC chart. Changing any or a combination of the three components could provide an interesting amount of insights towards developing hybrid models for process monitoring. It should be noted that the later two components would depend on the dataset used. In fact, a set of combination of models for different datasets will provide insights into effective usages of models for a specific dataset type, thus saving time in real time application of such frameworks.

- Holistic:

  It should be noted that the case studies presented in chapters 3 and 4 present processes of applying some advanced DL based algorithms towards high dimensional data in maintenance context. The two case studies are mutually exclusive and conducted on separate datasets and in specific contexts, the first in the prognostic and the second in diagnostic. For future research, the scope of each of these studies can be widened by using similar approaches to other datasets. Using different datasets will not only affect the preprocessing steps but also the selection of models from the ones mentioned in the literature review, the modifications required to successfully and efficiently apply the models to the datasets and perhaps present new set of challenges.

The needs and benefits of intelligent machine and system fault diagnosis and prognosis are apparent. The positive impacts of these diagnostic and prognostic capabilities on reducing operation and support costs and total ownership costs are significant. Such capabilities are thus key enablers of the new and revolutionary maintenance concepts being implemented today. All the technology elements and tool-kit capabilities are available and coming together to enable this vision of accurate and real intelligent machine and complex system diagnosis and prognosis and health management. And finally, all these diagnostic and prognostic technology elements, techniques, and capabilities must be applied and implemented wisely to obtain the maximum benefit impacts. In the timeline of failure progression starting from left to right, the ideal goal is to detect ”state change” as far to the left as possible. Diagnostic capabilities traditionally have been applied at or between the initial detection of a system failure and complete system catastrophic failure. More recent diagnostic technologies such as the one presented in our work in chapter 4 enable detections to be made at much earlier in an initial fault stages. In order to maximize the benefits of continued operational life of a system, maintenance often will be delayed until the early incipient fault progresses to a more severe state but before an actual failure event. This area between very early detection of incipient faults and progression to actual system or component failure states is the realm of prognostic technologies. And the specific domain of real predictive prognosis is being able to accurately predict useful life remaining along a specific failure progression timeline for a particular system or component. It is also understandable that in sometimes, the available sensors currently used for diagnosis provide adequate prognostic state awareness inputs, and sometimes advanced sensors or additional incipient fault detection techniques are required. This brings up the questions: How early or small of an incipient fault detection, and how small of a material state change is the best? Exploring the answer to these questions could be a way towards a holistic maintenance policy for a particular system that has an ideal balance between prognostic and diagnostic approaches.

Bibliography

[1] Aalen OO. (1975) ”Statistical inference for a family of counting processes.” *PhD thesis, Univ. of California, Berkeley.*

[2] Andersen PK, Borgan O, Gill RD, Keiding N. (1993) “Statistical Models Based on Counting Processes” *Springer, New York*

[3] Fleming TR and Harrington DP. (1991) “Counting Processes and Survival Analysis” *Wiley, New York*

[4] Kalbfleisch JD and Prentice RL. (2002) “The Statistical Analysis of Failure Time Data” *Wiley, New York*

[5] Lawless JF. (2003) “Statistical Models and Methods for Lifetime Data” *John Wiley & Sons*

[6] Klein JP and Moeschberger ML. (1997) “Survival Analysis: Techniques for Censored and Truncated Data” *Springer-Verlag Inc*

[7] Therneau T and Grambsch P. (2000) “Modeling Survival Data: Extending the Cox Model” *Springer-Verlag New York*

[8] Tableman M and Kim JS. (2003) “Surival Analysis Using S” *Chapman and Hall/CRC*

[9] Martinussen T and Scheike TH. (2006) “Dynamic Regression Models for Survival Data” *Springer, New York*

[10] Kaplan, E. L.; Meier, P. (1958) ”Nonparametric estimation from incomplete observations” *J. Amer. Statist. Assoc.* 53 (282): 457–481. doi:10.2307/2281868. JSTOR 2281868.

[11] Klein, J. P. and Moeschberger, M. L. (2005) “Survival Analysis: Techniques for Censored and Truncated Data” *Springer Science & Business Media*

[12] COX, J.C., SMITH, V.L. and WALKER, J.M. (1984) “Theory and Behavior of Multiple Unit Discriminative Auctions” *The Journal of Finance* 39: 983-1010. doi:10.1111/j.1540-6261.1984.tb03888.x

[13] Ping Wang, Yan Li, and Chandan k. Reddy. 2019 “Machine Learning for Survival Analysis: A Survey” *ACM Computing Surveys* 51, 6, Article 110 (February 2019), 36 pages. https://doi.org/10.1145/3214306

[14] Ching-Fan Chung, Peter Schmidt, and Ana D Witte. 1991 “Survival analysis: A survey” *Journal of Quantitative Criminology* 7, 1 (1991), 59–98.

[15] David G Kleinbaum and Mitchel Klein. 2006 “Survival analysis: a self-learning text” *Springer Science Business Media*

[16] Elisa T Lee and John Wang. 2003 “Statistical methods for survival data analysis” *John Wiley & Sons* Vol. 476

[17] Paul D Allison. 2010 “Survival analysis using SAS: a practical guide” *Sas Institute*

[18] Joseph A Cruz and David S Wishart. 2006 “Applications of machine learning in cancer prediction and prognosis” *Cancer informatics 2 (2006)*

[19] Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios IFotiadis. 2015 “Machine learning applications in cancer prognosis and prediction” *Computational and structural biotechnology journal 13 (2015)* 8–17

[20] Hans C. van Houwelingen and Hein Putter. 2011 “Dynamic prediction in clinical survival analysis” *CRC Press*

[21] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001 ”The elements of statistical learning” *Springer* Vol. 1. Springer series in statistics

[22] Robert Tibshirani. 1997 ”The lasso method for variable selection in the Cox model” *Statistics in medicine* 16, 4 (1997), 385–395.

[23] Arthur E Hoerl and Robert W Kennard. 1970 ”Ridge regression: Biased estimation for nonorthogonal problems” *Technometrics* 12, 1 (1970), 55–67.

[24] Pierre JM Verweij and Hans C Van Houwelingen. 1994 ”Penalized likelihood in Cox regression” *Statistics in medicine* 13, 23-24 (1994), 2427– 436.

[25] Pierre JM Verweij and Hans C Van Houwelingen. 1994 ”Penalized likelihood in Cox regression” *Statistics in medicine* 13, 23-24 (1994), 2427– 436.

[26] Hui Zou and Trevor Hastie. 2005 ”Regularization and variable selection via the elastic net” *Journal of the Royal Statistical Society: Series B Statistical Methodology* 67, 2 (2005), 301–320.

[27] Noah Simon, Jerome Friedman, Trevor Hastie, Rob Tibshirani, and others. 2011 ”Regularization paths for Cox's proportional hazards model via coordinate descent” *Journal of statistical software* 39, 5 (2011), 1–13.

[28] Sen Yang, Lei Yuan, Ying-Cheng Lai, Xiaotong Shen, Peter Wonka, and Jieping Ye. 2012 ”Feature grouping and selection over an undirected graph” *In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* ACM, 922–930.

[29] Jieping Ye and Jun Liu. 2012 ”Sparse methods for biomedical data” *ACM SIGKDD Explorations Newsletter 14*, 1 (2012), 4–15.

[30] Harald Binder and Martin Schumacher. 2008 ”Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models” *BMC bioinformatics*, 9, 1 (2008), 1–10.

[31] Blaz Zupan, Janez Dem˘sAr, Michael W Kattan, Robert J Beck, and Ivan Bratko. 2000 ”Machine learning for survival analysis: a case study on recurrence of prostate cancer” *Artificial intelligence in medicine 20*, 1 (2000), 59–75.

[32] Korvesis, Panagiotis ”Machine Learning for Predictive Maintenance in Aviation” *Theses, https://pastel.archives-ouvertes.fr/tel-02003508*, Nov, 2017..

[33] Ciampi, R.S. Bush, M. Gospodarowicz, and J.E. Till. 1981 ”An approach to classifying prognostic factors related to survival experience for non-Hodgkin's lymphoma patients: Based on a series of 982 patients” : 1967–1975. Cancer 47, 3 (1981), 621–627.

[34] Louis Gordon and Richard A Olshen. 1985 ”Tree-structured survival analysis” *Cancer treatment reports* 69, 10 (1985), 1065–1069.

[35] Louis Gordon and Richard A Olshen. 1985 ”A survey of decision tree classifier methodology” *IEEE transactions on systems, man, and cybernetics* 21, 3 (1991), 660–674.

[36] Roger B Davis and James R Anderson. 1989 ”Exponential survival trees” *Statistics in Medicine* 8, 8 (1989), 947–961.

[37] Gensheimer MF, Narasimhan B ”A scalable discrete-time survival model for neural networks” *PeerJ. 2019*;7:e6257. Published 2019 Jan 25. doi:10.7717/peerj.6257

[38] Michael LeBlanc and John Crowley. 1992 ”Relative risk trees for censored survival data” *Biometrics* 48, 2 (1992), 411–425.

[39] Ciampi, C-H Chang, S. Hogg, and S. McKinney. 1987 ”Recursive partition: a versatile method for exploratory-data analysis in biostatistics” *In Biostatistics. Springer*, 23–50.

[40] Antonio Ciampi, Johanne Thiffault, Jean-Pierre Nakache, and Bernard Asselain, 1986. ”Stratification by stepwise regression, correspondence analysis and recursive partition: a comparison of three methods of analysis for survival data with covariates” *Computational statistics data analysis* 4, 3 (1986), 185–204

[41] Mark R Segal. 1988 ”Regression trees for censored data” *Biometrics 44*, 1 (1988), 35–47.

[42] Imad Bou-Hamad, Denis Larocque, Hatem Ben-Ameur, and others. 2011 ”A review of survival trees” *Statistics Surveys* 5 (2011), 44–71.

[43] Nir Friedman, Dan Geiger, and Moises Goldszmidt. 1997 ”Bayesian network classifiers” *Machine learning 2*, 2 (1997), 131–163.

[44] Igor Kononenko, 1993 ”Inductive and Bayesian learning in medical diagnosis” *Artificial Intelligence an International Journal 7*, 4 (1993), 317–337.

[45] Margaret S Pepe. 2003 ”The statistical evaluation of medical tests for classification and prediction” *Oxford University Press, USA*

[46] Blaz Zupan, Janez Dem˘sAr, Michael W Kattan, Robert J Beck, and Ivan Bratko. 2000. ”Machine learning for survival analysis: a case study on recurrence of prostate cancer” *Artificial intelligence in medicine 20*, 1 (2000), 59–75.

[47] Adrian Raftery, David Madigan, and Chris T. Volinsky. 1995 ”Accounting for model uncertainty in survival analysis improves predictive performance” *Bayesian Statistics 5 (1995)*, 323–349.

[48] Riccardo Bellazzi and Blaz Zupan. 2008 ”Predictive data mining in clinical medicine: current issues and guidelines” *International journal of medical informatics 77*, 2 (2008), 81–97.

[49] Mahtab J Fard, Ping Wang, Sanjay Chawla, and Chandan K Reddy. 2016 ”A bayesian perspective on early stage event prediction in longitudinal data” *IEEE Transactions on Knowledge and Data Engineering* 28, 12 (2016), 3126–3139.

[50] Paulo JG Lisboa, H Wong, P Harris, and Ric Swindell. 2003 ”A Bayesian neural network approach for modelling censored data with an application to prognosis after surgery for breast cancer” *Artificial intelligence in medicine* 28, 1 (2003), 1–25.

[51] David JC MacKay. 1995 ”Probable networks and plausible predictions-a review of practical bayesian methods for supervised neural network” *Network: Computation in Neural Systems* 6, 3 (1995), 469–505.

[52] Adrian E Raftery. 1995 ”Bayesian model selection in social research” *Sociological methodology* 25 (1995), 111–163.

[53] Sattar Ameri, Mahtab J Fard, Ratna B Chinnam, and Chandan K Reddy. 2016 ”Survival analysis-based framework for early prediction of student dropouts” *In Proceedings of ACM International Conference on Conference on Information and Knowledge Management* ACM, 903–912.

[54] Frank Rosenblatt. 1958 ”The perceptron: a probabilistic model for information storage and organization in the brain” *Psychological review* 65, 6 (1958), 386–408.

[55] David Faraggi and Richard Simon. 1995 ”A neural network model for survival data.” *Statistics in medicine* 14, 1 (1995), 73–82.

[56] L Mariani, D Coradini, E Biganzoli, P Boracchi, E Marubini, S Pilotti, B Salvadori, R Silvestrini, U Veronesi, R Zucali, and others. 1997 ”Prognostic factors for metachronous contralateral breast cancer: a comparison of the linear Cox regression model and its artificial neural network extension” *Breast cancer research and treatment* 44, 2 (1997), 167–178.

[57] Bart Baesens, Tony Van Gestel, Maria Stepanova, Dirk Van den Poel, and Jan Vanthienen. 2005 ”Neural network survival analysis for personal loan data” *Journal of the Operational Research Society* 56, 9 (2005), 1089–1098.

[58] Knut Liestbl, Per Kragh Andersen, and Ulrich Andersen. 1994 ”Survival analysis and neural nets” *Statistics in medicine* 13, 12 (1994), 1189–1200.

[59] Elia Biganzoli, Patrizia Boracchi, Luigi Mariani, and Ettore Marubini. 1998 ”Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach” *Statistics in medicine* 17, 10 (1998), 1169–1186.

[60] Stephen F Brown, Alan J Branford, and William Moran. 1997 ”On the use of artificial neural networks for the analysis of survival data” *Neural Networks, IEEE Transactions on 8*, 5 (1997), 1071–1077.

[61] Peter M Ravdin and Gary M Clark. 1992 ”A practical application of neural network analysis for predicting outcome of individual breast cancer patients” *Breast Cancer Research and Treatment* 22, 3 (1992), 285–293.

[62] L Mariani, D Coradini, E Biganzoli, P Boracchi, E Marubini, S Pilotti, B Salvadori, R Silvestrini, U Veronesi, R Zucali, and others. 1997 ”Prognostic factors for metachronous contralateral breast cancer: a comparison of the linear Cox regression model and its artificial neural network extension” *Breast cancer research and treatment* 44, 2 (1997), 167–178.

[63] Jiawen Yao, Xinliang Zhu, Feiyun Zhu, and Junzhou Huang. 2017 ”Deep Correlational Learning for Survival Prediction from Multi-modality Data” *In Medical Image Computing and Computer-Assisted Intervention MICCAI 2017, Maxime Descoteaux, Lena Maier-Hein, Alfred Franz, Pierre Jannin, D. Louis Collins, and Simon Duchesne (Eds.). Springer International Publishing* 406–414.

[64] Jared Katzman, Uri Shaham, Jonathan Bates, Alexander Cloninger, Tingting Jiang, and Yuval Kluger. 2016 ”Deep survival: A deep cox proportional hazards network” *arXiv preprint* arXiv:1606.00931 (2016).

[65] Katzman, J.L., Shaham, U., Cloninger, A. et al ”DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network” *BMC Med Res Methodol* 18, 24 (2018). https://doi.org/10.1186/s12874-018-0482-1

.

[66] Rajesh Ranganath, Adler Perotte, No´emie Elhadad, and David Blei. 2016 ”Deep survival analysis” *arXiv preprint* arXiv:1608.02158 (2016).

[67] X. Zhu, J. Yao and J. Huang, 2016 ”Deep convolutional neural network for survival analysis with pathological images” *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Shenzhen, 2016*, pp. 544-547, doi: 10.1109/BIBM.2016.7822579.

[68] How Jing and Alexander J Smola. 2017 "Neural survival recommender" *In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining ACM*, 515–524.

[69] Guolei Yang, Ying Cai, and Chandan K Reddy. 2018 "Spatio-Temporal Check-in Time Prediction with Recurrent Neural Network based Survival Analysis" *In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*

[70] Alex J Smola and Bernhard Sch¨olkopf. 2004 "A tutorial on support vector regression" *IStatistics and computing* 14, 3 (2004), 199–222.

[71] lex J Smola and Bernhard Sch¨olkopf. 1998 "Learning with kernels" *https://mitpress.mit.edu/books/learning-kernels*

[72] Pannagadatta K Shivaswamy, Wei Chu, and Martin Jansche. 2007 "A support vector approach to censored targets" *In Proceedings of the IEEE International Conference on Data Mining (ICDM). IEEE*, 655–660.

[73] Sariel Har-Peled, Dan Roth, and Dav Zimak. 2002 "Constraint classification: A new approach to multiclass classification" *Algorithmic Learning Theory. Springer*, 365–379.

[74] Faisal M Khan and Valentina B Zubek. 2008 "Support vector regression for censored data (SVRc): a novel tool for survival analysis" *In Proceedings of the IEEE International Conference on Data Mining (ICDM). IEEE*, 863–868.

[75] Belle V Van, Kristiaan Pelckmans, Johan AK Suykens, and S Van Huffel. 2007 "Support vector machines for survival analysis" *In Proceedings of the Third International Conference on Computational Intelligence in Medicine and Healthcare (CIMED2007)* 1–8.

[76] Belle Van, Kristiaan Pelckmans, Huffel S Van, and Johan AK Suykens. 2011 "Support vector methods for survival analysis: a comparison between ranking and regression approaches" *Artificial intelligence in medicine* 53, 2 (2011), 107–118.

[77] Achmad Widodo and Bo-Suk Yang. 2011 "Application of relevance vector machine and survival probability to machine degradation assessment" *Expert Systems with Applications* 38, 3 (2011), 2592–2599.

[78] Achmad Widodo and Bo-Suk Yang. 2011 "Relevance vector machine for survival analysis" *IEEE transactions on neural networks and learning systems* 27, 3 (2016), 648–660.

[79] Thomas G Dietterich. 2000 "Ensemble methods in machine learning. In International workshop on multiple classifier systems" *Springer*, 1–15.

[80] Leo Breiman. 1996 "Bagging predictors" *Machine learning* 24, 2 (1996), 123–140.

[81] Leo Breiman. 2001 "Random forests" *Machine learning 45*, 1 (2001), 5–32.

[82] Torsten Hothorn, Berthold Lausen, Axel Benner, and Martin Radespiel-Tr¨oger. 2004. "Bagging survival trees" *Statistics in medicine* 23, 1 (2004), 77–91.

[83] Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. 2008 "Random survival forests" *The annals of applied statistics* 2, 3 (2008), 841–860.

[84] Hemant Ishwaran, Udaya B Kogalur, Xi Chen, and Andy J Minn. 2011 "Random survival forests for high dimensional data" *Statistical analysis and data mining* 4, 1 (2011), 115–132.

[85] Torsten Hothorn, Peter Buhlmann, Sandrine Dudoit, Annette Molinaro, and Mark J Van Der Laan. 2006 "Survival ensembles" *Biostatistics* 7, 3 (2006), 355–373.

[86] Peter Buhlmann and Torsten Hothorn. 2007 "Boosting algorithms: Regularization, prediction and model fitting" *Statist. Sci.* 22, 4 (2007), 477–505.

[87] Bhanukiran Vinzamuri, Yan Li, and Chandan K Reddy. 2014 "Active learning-based survival regression for censored data" *In Proceedings of the 23rd ACM international conference on conference on information and knowledge management.* ACM, 241–250.

[88] Sinno J Pan and Qiang Yang. 2010 "A survey on transfer learning" *IEEE Transactions on knowledge and data engineering* 22, 10 (2010), 1345–1359.

[89] Yan Li, Lu Wang, Jie Wang, Jieping Ye, and Chandan K Reddy. 2016c "Transfer learning for survival analysis via efficient L2,1-norm regularized Cox regression" *In Proceedings of the IEEE International Conference on Data Mining (ICDM)*. IEEE, 231–240.

[90] Yan Li, Jie Wang, Jieping Ye, and Chandan K Reddy. 2016d "A multi-task learning formulation for survival analysis" *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1715–1724.

[91] Alan D. Saul 2016 "Gaussian process-based approaches for survival analysis" *University of Sheffield, UK*, 2016

[92] Alan D. Saul 2016 ”Wsisa: Making survival prediction from whole slide histopathological images” *Proceedings of the IEEE Conference on Computer, 2017*

[93] C. Lee, W. R. Zame, J. Yoon, M. van der Schaa ”DeepHit: A Deep Learning Approach to Survival Analysis with Competing Risks” *AAAI Conference on Artificial Intelligence (AAAI)*, 2018

[94] Håvard Kvamme, Ørnulf Borgan, and Ida Scheel ”Time-to-event prediction with neural networks and Cox regression” *Journal of Machine Learning Research*, 20(129):1–30, 2019. PYCOX

[95] Ching T, Zhu X, Garmire LX (2018) ”Cox-nnet: An artificial neural network method for prognosis prediction of high-throughput omics data” *PLoS Comput Biol* 14(4): e1006076. https://doi.org/10.1371/journal.pcbi.1006076

[96] Ren, Kan et al 2019 ”Deep Recurrent Survival Analysis” *AAAI* (2019).

[97] Bottou, Léon ”Stochastic gradient learning in neural networks” *Proceedings of Neuro-Nımes* 91.8 (1991): 12.

[98] Rodríguez, G. (2007) ”Lecture Notes on Generalized Linear Models” *URL: http://data.princeton.edu/wws509/notes/*

[99] S. F. Brown, A. J. Branford and W. Moran 1997 ”On the use of artificial neural networks for the analysis of survival data” *IEEE Transactions on Neural Networks*, vol. 8, no. 5, pp. 1071-1077, Sept. 1997, doi: 10.1109/72.623209

[100] P Nolan, D. (2011) ”Handbook of Fire and Explosion Protection Engineering Principles” doi: https://doi.org/10.1016/C2009-0-64221-5

[101] Berk, Z. (2018) ”Food process engineering and technology” doi: https://doi.org/10.1016/B978-0-12-373660-4.X0001-4

[102] Ali, S., Pievatolo, A., Göb, R. (2016) ”An overview of control charts for high-quality processes” *Quality and reliability engineering international*, 32(7), 2171-2189.

[103] Reynolds Jr, M. R. (2013) ”The Bernoulli CUSUM chart for detecting decreases in a proportion” *Quality and Reliability Engineering International*, 29(4), 529-534.

[104] Woodall, W. H. (1997) ”Control charts based on attribute data: bibliography and review” *Journal of quality technology*, 29(2), 172-183.

[105] Woodall, W. H. (1997) ”A review and perspective on control charting with image data” *Journal of Quality Technology*, 43(2), 83-98.

[106] Prabhu, S. S., Runger, G. C. (1997) ”Designing a multivariate EWMA control chart” *Journal of Quality Technology*, 29(1), 8-15.

[107] A. S. Neubauer (1997) ”The EWMA control chart: properties and comparison with other quality-control procedures by computer simulation” *Clinical Chemistry*, vol. 43, no. 4, pp. 594–601, 1997

[108] Young, William Y; Svrcek, Donald P; Mahoney, Brent R (2014) ”A Brief History of Control and Simulation” *A Real Time Approach to Process Control (3 ed.). Chichester, West Sussex, United Kingdom: John Wiley Sons Inc.* pp. 1–2. ISBN 978-1119993872.

[109] ”NIST/SEMATECH e-Handbook of Statistical Methods” *http://www.itl.nist.gov/div898/handbook* Last updated: 10/30/2013

[110] ”Control Chart” *https://asq.org/quality-resources/control-chart* © 2020 American Society for Quality

[111] Nembhard, H. B.; Ferrier, N. J.; Osswald, T. A.; and Sanz-Uribe, J. R. (2003) ”An Integrated Model for Statistical and Vision Monitoring in Manufacturing Transitions” *Quality and Reliability Engineering International* 19, pp. 461–476.

[112] Lyu, J. and Chen, M. (2009) ”Automated Visual Inspection Expert System for Multivariate Statistical Process Control Chart” *Expert Systems with Applications* 36, pp. 5113–5118.

[113] A. Saxena, K. Goebel, D. Simon and N. Eklund (2008) ”Damage propagation modeling for aircraft engine run-to-failure simulation” *International Conference on Prognostics and Health Management*, Denver, CO, 2008, pp. 1-9, doi: 10.1109/PHM.2008.4711414.

[114] Cameron Davidson-Pilon, Jonas Kalderstam, Noah Jacobson, sean-reed, Ben Kuhn, Paul Zivich, ... Abraham Flaxman. (2020, July 9) *CamDavidsonPilon/lifelines*: v0.24.16 (Version v0.24.16). Zenodo. http://doi.org/10.5281/zenodo.3937749

[115] A. Saxena and K. Goebel (2008). ”Turbofan Engine Degradation Simulation Data Set” *NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository)* NASA Ames Research Center, Moffett Field, CA

[116] John P. Klein and Melvin L. Moeschberger (2003) ”Survival Analysis: Techniques for Censored and Truncated Data” *Springer, New York*, 2. edition, 2003.

[117] Zheng, Y. and Heagerty, P.J. (2005) " Partly Conditional Survival Models for Longitudinal Data" *Biometrics*, 61: 379-391. doi:10.1111/j.1541-0420.2005.00323.x

[118] L. N. Smith (2017) "Cyclical learning rates for training neural networks" *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472, 2017.

[119] Hassan, Hassan Negm, Abdelazim Zahran, Mohamed Saavedra, Oliver. (2015) "Assessment of Artificial Neural Network for bathymetry estimation using High Resolution Satellite imagery in Shallow Lakes: Case Study El Burullus Lake" *International Water Technology Journal. 5*

[120] Hastie, Trevor. Tibshirani, Robert. Friedman, Jerome (2009) "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" *Springer, New York, NY*, 2009.

[121] Uno H, Cai T, Pencina MJ, D'Agostino RB, Wei LJ "n the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data" *Stat Med. 2011 May* 10;30(10):1105-17. doi: 10.1002/sim.4154. Epub 2011 Jan 13. PMID: 21484848; PMCID: PMC3079915.

[122] F.M.cMegahed, W. H. Woodall, and J.A. Camelio, "A review and perspective on control charting with image data," *Journal of Quality Technology*, vol. 43, no. 2, pp. 83–98, 2011.

[123] H. Hotelling, "The generalization of student's ratio," *Ann. Math. Statist.*, vol. 2, no. 3, pp. 360–378, 08 1931.

[124] Donald Knuth, "Designing a Multivariate EWMA Control Chart," *Journal of Quality Technology*, vol. 29, no. 1, pp. 8–15, 1997.

[125] A. S. Neubauer, "The EWMA control chart: properties and comparison with other quality-control procedures by computer simulation," *Clinical Chemistry*, vol. 43, no. 4, pp. 594–601, 1997.

[126] A. McCallum and K. Nigam, "A comparison of event models for naive Bayes text classification," *Proceedings in Workshop on Learning for Text Categorization*, pp. 41–48, 1998.

[127] A.J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, pp. 199–222, 2004.

[128] Y. Liu and J.Xu, "Periodic Surface Defect Detection in Steel Plates Based on Deep Learning", *Applied Science*, vol. 9, pp-1-14, 2019.

[129] M. Haselmann. D.P. Gruber, "Pixel-Wise Defect Detection by CNNs without Manually Labeled Training Data," *Applied Artificial Intelligence*, vol. 33, 6, pp. 548-566, 2019.

[130] L. Scime, J. Beuth, “A Multi-scale Convolutional Neural Network for Autonomous Anomaly Detection and Classification in a Laser Powder Bed Fusion Additive Manufacturing Process,” *Additive Manufacturing*, vol. 24, pp. 273-286, 2018.

[131] T. Nakazawa and D.V. Kulkarni, “Anomaly Detection and Segmentation for Wafer Defect Patterns Using Deep Convolutional Encoder–Decoder Neural Network Architectures in Semiconductor Manufacturing,” *IEEE Trans. Sem. Manufacturing*, vol. 32, 250-256, 2019.

[132] D. Tabernik, S. Sela, J. Skvarce and D. Skocaj, “Segmentation-based Deep-learning Approach for Surface-Defect Detection,” *Journal of Intelligence Manufacturing*, pp. 1-18, 2019.

[133] F. Imani, R. Chen, E. Diewald and E. Reutzel, “Deep Learning of Variant Geometry in Layerwise Imaging Profiles for Additive Manufacturing Quality Control,” *Journal of Manufacturing Science and Engineering*, pp. 1-16, 2019.

[134] N. Enshaei and F. Naderkhani, “Application of Deep Learning for Fault Diagnostic in Induction Machine’s Bearings,” *IEEE International Conference on Prognostic and Health Management*, pp. 1-7, 2019.

[135] R. Girshick, “Fast r-cnn,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015.

[136] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, K. Murphy, “Speed/accuracy trade-offs for modern convolutional object detectors,” *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7310–7311, 2017.

[137] M. Abadi *et al*, “TensorFlow: Large-scale machine learning on heterogeneous Systems,” *Software available from tensorflow.org*, 2015.

[138] Hao Yan ; Kamran Paynabar ; Jianjun Shi “Image-Based Process Monitoring Using Low-Rank Tensor Decomposition,” *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 216-227, 2015.