DEVELOPMENT OF EQUIPMENT FAILURE PROGNOSTIC MODEL BASED ON LOGICAL ANALYSIS OF DATA (LAD)

by

Sasan Esmaeili

Submitted in partial fulfilment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
July 2012

DALHOUSIE UNIVERSITY

DEPARTMENT OF INDUSTRIAL ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "DEVELOPMENT OF EQUIPMENT FAILURE PROGNOSTIC MODEL BASED ON LOGICAL ANALYSIS OF DATA (LAD)" by Sasan Esmaeili in partial fulfilment of the requirements for the degree of Master of Applied Science.

Dated: July 27, 2012

Supervisor: _____

Readers: _____

_____

_____

DALHOUSIE UNIVERSITY

DATE: July 27, 2012

AUTHOR: Sasan Esmaeili

TITLE: DEVELOPMENT OF EQUIPMENT FAILURE PROGNOSTIC MODEL BASED ON LOGICAL ANALYSIS OF DATA (LAD)

DEPARTMENT OR SCHOOL: Department of Industrial Engineering

DEGREE: M.A.Sc. CONVOCATION: October YEAR: 2012

_____

Signature of Author

پیشکش به شیوا و خانواده ی دلسوزم که در این راه مرا صمیمانه تشویق و پشتیبانی نمودند.

To my loving Shiva and family, who have wholeheartedly inspired and supported me along the way.

TABLE OF CONTENTS

LIST OF TABLES vii
LIST OF FIGURES viii
ABSTRACT ix
LIST OF ABBREVIATIONS AND SYMBOLS USED x
ACKNOWLEDGEMENTS xii
CHAPTER 1 : INTRODUCTION 1
1. Diagnostics and Prognostics 1
1.1. Physical Model-Based Approaches 1
1.2. Knowledge-Based Approaches 2
1.3. Data-Driven Approaches 4
1.3.1. Artificial Intelligent (AI) Approaches 4
1.3.2. Statistical Approaches 7
2. Logical Analysis of Data (LAD) 10
2.1. Data Binarization and Support Set Selection 11
2.2. Pattern Generation and Pattern Selection 16
2.3. Theory Formation 21
2.4. Developments 22
2.5. Performance Comparisons 25
3. Objective of Research 26
CHAPTER 2 : METHODOLOGY 27
1. Data Binarization 28
1.1. Sensitive Discriminating Method 28
1.2. Equipartitioning Method 29
2. Pattern Generation 31
2.1. Mixed Integer Linear Programming (MILP) Method 31
2.2. Hybrid Greedy Method 36
3. Pattern's Quality Evaluation 40
4. Model Formation 42
4.1. Failure Diagnostic 43
4.2. Failure Prognostic 45

CHAPTER 3 : EXPERIMENTS 50
1. Data Preparation 50
2. Sample Prognostic Results 52
3. Design Of Experiment (DOE) 55
4. Comparisons 56
4.1. Method # 1 vs. Method # 2 56
4.2. Hybrid Greedy # 1 vs. … vs. Hybrid Greedy # 12 57
4.3. Hybrid Greedy vs. MILP vs. PHM 57
CHAPTER 4 : CONCLUSION 64
Reference List 66

LIST OF TABLES

Table 1. Sample Set of Monitored Data 27
Table 2. Sorted Age Attribute and Its Corresponding Classes 28
Table 3. Sorted Condition Attribute and Its Corresponding Classes 29
Table 4. Binary Transformation of Sample Set of Monitored Data 29
Table 5. Sorted Age Attribute 30
Table 6. Sorted Condition Attribute 30
Table 7. Binary Transformation of Sample Set of Monitored Data 30
Table 8. Positive and Negative Patterns - MILP Method 36
Table 9. Positive and Negative Patterns - Hybrid Greedy Method 40
Table 10. Quality Measures of Positive Patterns - MILP Method 41
Table 11. Quality Measures of Negative Patterns - MILP Method 41
Table 12. Quality Measures of Positive Patterns - Hybrid Greedy Method 42
Table 13. Quality Measures of Negative Patterns - Hybrid Greedy Method 42
Table 14. Sample Set of Monitored Data for Diagnostic or Prognostic 44
Table 15. Binary Transformation of Sample Set for Diagnostic or Prognostic 44
Table 16. Classification of Sample Set based on MILP Patterns 44
Table 17. Classification of Sample Set based on Hybrid Greedy Patterns 45
Table 18. List of Observations from Train Set, Covered by Hybrid Greedy Patterns 46
Table 19. KM Estimation of Conditional Survival Probability-Hybrid Greedy Patterns 47
Table 20. KM Estimation of Baseline Conditional Survival Probability 47
Table 21. Conditional Survival Probabilities of Equipment–1st Calculation Method 48
Table 22. Conditional Survival Probabilities of Equipment–2nd Calculation Method 49
Table 23. Correlation Matrix 50
Table 24. Eigenvalue, Variability, and Cumulative Variability 51
Table 25. Set of Monitored Data for a Test Equipment 52
Table 26. Prognostic Results for the Test Equipment 53
Table 27. Design Of Experiments (DOE) 56
Table 28. Method #1 vs. Method #2 (sample experiment) 57
Table 29. Hybrid Greedy #1 vs. … vs. Hybrid Greedy #12 57
Table 30. Hybrid Greedy vs. MILP vs. PHM 58
Table 31. Run-Time: Hybrid Greedy vs. MILP vs. PHM 63

LIST OF FIGURES

Figure 1. Bottom-Up Phase……... 37
Figure 2. Top-Down Phase…… 39
Figure 3. Conditional Survival Probability of the Last Observation……. 49
Figure 4. A Comparison Between MRL and Actual RL……... 54
Figure 5. Survival Function for the Next 5 Periods at Each Observation Moment……...55
Figure 6. Survival Function using Hybrid Greedy Method……... 58
Figure 7. Survival Function using MILP Method……. 59
Figure 8. Survival Function using PHM Method…….. 59
Figure 9. Difference Between MRL and Actual RL using Different Methods……. 60
Figure 10. Survival Function for the Next 5 Periods at Each Observation Moment…….61

ABSTRACT

This research develops an equipment failure prognostics model to predict the equipment's chance of survival, using LAD. LAD benefits from not relying on any statistical theory, which enables it to overcome the problems concerning the statistical properties of the datasets. Its main advantage is its straightforward process and self-explanatory results.

Herein, our main objective is to develop models to calculate equipment's survival probability at a certain future moment, using LAD. We employ the LAD's pattern generation procedure. Then, we introduce a guideline to employ generated patterns to estimate the equipment's survival probability.

The models are applied on a condition monitoring dataset. Performance analysis reveals that they provide comprehensible results that are greatly beneficial to maintenance practitioners. Results are compared with PHM's results. The comparison reveals that the LAD models compare favorably to the PHM. Since they are at their beginning phase, some future directions are presented to improve their performances.

LIST OF ABBREVIATIONS AND SYMBOLS USED

AI	Artificial Intelligent
ANN	Artificial Neural Network
CBM	Condition Based Maintenance
DOE	Design Of Experiment
ES	Expert System
FL	Fuzzy Logic
HMM	Hidden Markov Model
HR	Hazard Rate
HSMM	Hidden Semi-Markov Model
ILP	Integer Linear Programming
KM	Kaplan-Meier
LAD	Logical Analysis of Data
LAND	Logical Analysis of Numerical Data
LASD	Logical Analysis of Survival Data
MILP	Mixed Integer Linear Programming
MLE	Maximum Likelihood Estimation
MRL	Mean Residual Life
PCA	Principal Component Analysis
PHM	Proportional Hazards Model
PHR	Proportional Hazard Rate
RL	Residual Life
RUL	Remaining Useful Life
SPC	Statistical Process Control

a	numerical value of attribute
b	binary value of attribute
$b_{i,j}$	binary value showing attribute j exists in observation i
c	cut-point value
d	degree of pattern
E	set of pieces of equipment
H_p	weight of pattern p in discriminant function
N^*	number of observations in pattern's class
NP	negative pattern
PP	positive pattern
P	pattern
q	number of binary attributes
$r_{i,j}$	binary value showing attribute j exists in pattern i
S	set of observations
S^*	set of observations in pattern's class
S^{*-}	set of observations in opposite class
SP_b	baseline survival probability
SP_p	pattern survival probability
SP_{former}	survival probability of equipment at former observation moment
SP_{obs}	survival probability of equipment
T	maximum available survival period
w_j	binary value showing attribute j exists in pattern
y_i	binary value showing observation i is covered by pattern
z_i^e	performance measurement of experiment e for observation i
τ	failure time of equipment
τ_0	current age of equipment
Δ	observation period length

ACKNOWLEDGEMENTS

I would like to thank my supervisor Dr. Alireza Ghasemi for his guidance, assistance and advice for preparation of this research. Without his follow-ups none of this achievement would have been possible. I would also like to thank Dr. Claver Diallo, Dr. Pemberton Cyrus, and Dr. Ahsan Habib, who gave me useful advice as members of the supervisory committee.

I would also like to take this opportunity to thank my friend Shiva Naseri and my family for all the inspiration and support they provided so that I can successfully accomplish my goals.

Halifax, July 2012

Sasan Esmaeili

CHAPTER 1 : INTRODUCTION

Widely applied in maintenance, *Condition Based Maintenance* (*CBM*) [Jardine et al. (2006)] is a maintenance program that engages the equipment's health condition in optimizing or improving the maintenance activities. The equipment's age and health condition indicators are the factors based on which CBM diagnoses a fault in equipment or predicts an imminent failure. CBM constructs a model, which represents the relation between the equipment's age and health condition indicators with its failure, based on a given historical dataset, called the *Train Set*. Then, it examines the quality of the model by applying it on another part of the historical dataset, called the *Test Set*. The former process is *Train Phase*, while the latter one is *Test Phase*.

1. DIAGNOSTICS AND PROGNOSTICS

Applications of CBM can be divided into two categories: *Diagnostics*, which focus on the detection of a fault in equipment at the current moment, and *Prognostics*, which focus on the prediction of a fault in equipment before it happens. Diagnostics aim to detect age and health condition indicators in the measurement space representing a fault in the physical space (equipment). Prognostics aim to predict the probability of or the time left before a fault in the physical space (equipment) based on age and health condition indicators in the measurement space. The following sections describe different approaches in both diagnostics and prognostics, divided into three categories: *Physical Model-Based* approaches, *Knowledge-Based* approaches, and *Data-Driven* approaches.

1.1. Physical Model-Based Approaches

Physical model-based approaches construct mathematical models that directly illustrate the process of physical deterioration in the equipment health condition. These approaches usually employ expert knowledge to construct the model, and then, validate the model by applying it on some test data. They detect equipment failure indicators, called *Residuals*, by applying methods such as *Kalman Filter* [Kalman (1960)] on the train set. They decide whether a failure has occurred or not by comparing the equipment's health condition indicators with their corresponding thresholds, detected as the equipment's residuals. Several physical model-based approaches have been proposed in both

diagnostics and prognostics. [Y. Li et al. (1999)] proposed a *Defect Propagation Model* to estimate the *Remaining Useful Life* (*RUL*) of equipment. It employs a time-based defect growth propagation model. In order to estimate the parameters of the model, it employs a *Recursive Least Square* algorithm. [Luo et al. (2003)] introduced an integrated prognostic model that is constructed based on the data provided using the model-based simulations. However, the accuracy of model-based approaches greatly depends on whether an accurate mathematical model is achievable or not. In cases where an accurate mathematical model is achievable, model-based approaches outperform other approaches. Model-based approaches usually are not applicable for complex systems of equipment.

1.2. Knowledge-Based Approaches

As opposed to physical model-based approaches, knowledge-based approaches are not dependent on any mathematical interpretation of the physical process of deterioration in equipment health condition. *Expert Systems* (*ESs*) and *Fuzzy Logic* (*FL*) are two most widely used knowledge-based approaches [Peng et al. (2010)].

Expert Systems (ESs) are computer systems that interface an expert with a program in order to store the expert knowledge. The stored knowledge is used to train an ES, mostly in the form of *Rules*, based on which the program automatically simulates the expert inference procedure to solve the problem. Rules are generally demonstrated in the form of '*IF fact*, *THEN result*'. The result can also be employed as a fact to build a new rule, or be tied up with other rules to build a new rule. Many ES approaches have been proposed in both diagnostics and prognostics. [Wen et al. (2003)] employed an ES, based on *Case-Based Reasoning*, for diagnostics. It simulates the process of encoding and training the past observations. The ES is trained based on the experts' judgments about the equipment health condition at different cases (different sets of observed health condition indicators) for the past observations. Then, for each newly observed equipment, a distance measure is calculated which represents how close the new case is to the previously judged cases. Eventually, the diagnosis is performed based on the distance measure. [Araiza et al. (2002)] presented the diagnostic and prognostic software, based on employing *Model-Based Reasoning* in an ES. It simulates the equipment health condition based on the current health condition indicators. It is based on constructing a

Fault/Symptom Matrix, which represents the connection between faults and observed health condition indicators. A diagnostic framework is constructed based solving a *Set-Covering Problem* based on the data provided in the Fault/Symptom Matrix. [Stanek et al. (2001)] proposed an ES, based on the combination of Case-Based and Model-Based Reasoning, for diagnostics. Aided by a computer model of the equipment, it simulates all the possible failure cases based on different sets of health condition indicators (Model-Based Reasoning phase). The simulation inputs (health condition indicators) along with the simulation output (equipment's state) are collected as a new possible case in a database. The database is eventually used as a reference for diagnostic. For each newly observed equipment, the closest case is found by looking through the database (Case-Based Reasoning phase). For this purpose, a *Single Nearest Neighbor* classifier is employed, which chooses the case with the minimum *Euclidean* distance from the current case. However, the ESs have the disadvantage of *Combinatorial Explosion* at the rule generation phase. Combinatorial explosion is referred to as a computational problem that is caused due to a severe increase in the number of variables. Another disadvantage of ESs is their limited perception of new situations. In other words, they can only solve the situations for which they are trained. A main limitation of ESs is that both the expert knowledge and the reasoning method are usually inaccurate and uncertain. As a result, uncertainty measures such as those in fuzzy logic are combined with ESs.

Fuzzy Logic (FL) is a knowledge discovery method that detects the relation between a certain output and a set of inaccurate inputs. Since it utilizes linguistic variables, it benefits from resembling the human reasoning procedure, which enables the FL to handle the inaccurate inputs. Using the phrases such as 'high' and 'low' provides the ability to describe overlapping situations, and is superior to numerical description. FL is generally employed in other diagnostics and prognostics methodologies such as ESs and *Artificial Neural Networks* (*ANNs*). [Choi et al. (1995)] introduced an on-line fuzzy ES in order to diagnose the equipment's health condition. In order to extract the equipment's prognostic information from its diagnostic information at past observation moments, the *Levinson* algorithm is employed. The Levinson algorithm is a prediction tool that works based on the simple moving average of the historical data. This enables the model to predict the equipment's future health condition based on its current health condition.

1.3. Data-Driven Approaches

Data-driven approaches are mostly originated from the *Pattern Recognition* theory. They perceive diagnostics and prognostics knowledge by investigating the relation between the inputs (age or/and health condition indicators) and the outputs (equipment's state) of monitored data. Data-driven approaches are categorized into *Artificial Intelligent* (*AI*) approaches, which employ the training methods, and *Statistical* approaches, which employ the statistical methods.

1.3.1. Artificial Intelligent (AI) Approaches

Among the AI approaches, *Artificial Neural Networks* (*ANNs*) is the most widely used. It imitates the human brain structure to construct a network structure. The network structure consists of three layers: input layer, hidden layer, and output layer. Each input is indirectly connected with an output. The network relates the inputs to the outputs by assigning adjustable weight to the hidden layer that connects the inputs and outputs. This structure helps perceive a complicated function of multi-input and multi-output. So, this enables ANN to solve the problems for which analytical and traditional approaches are difficult to apply, or do not exist. Due to the reduction in run-time and problem difficulty, ANN possesses many attractions for diagnostic and prognostic purposes.

Many ANN approaches have been proposed in diagnostics. [Fan et al. (2002)] proposed a *Feed-Forward NN* for diagnostics. It divides an n-dimensional space of n measured health condition indicators into its sub-spaces. Each sub-space is called a *Rule*, for which the amount of its *Support* is calculated. The rule's support shows the number of data with the failure state that agree with the rule. Then, constructed rules are reduced and simplified by removing the rules whose support is lower than a pre-defined value. Extracted rules are eventually used to diagnose the equipment, from which new observations are collected, based on the value of its health condition indicators. [Spoerre (1997)] proposed a *Cascade Correlation NN* for diagnostics. It starts with a network that is only composed of the input and output layers, and tries to relate these layers directly. If the *Sum-Squared Error* measure of the function that relates these layers exceeds a pre-defined value, a hidden layer is added to the network. At each step, the candidate hidden layer with the maximum correlation with the output layer is added to the network. This

procedure is performed up to a point where the sum-squared error measure of the obtained network meets the user-defined acceptable level. The obtained network has the advantage of being the minimum-size network required for the diagnostics. [Baillie et al. (1996)] proposed a *Radial Basis Function NN* for diagnostics, and compared the performance of this approach with that of other autoregressive time series approach including *Back Propagation NN* and *Linear Regressive*. This approach is a Feed-Forward NN, in which the hidden layers are in the form of the radial basis functions such as *Gaussian Kernel* function. [C. J. Li et al. (1999)] proposed a *Recurrent NN* for diagnostics. The most important feature of this approach is that it starts with an existing network, trains a separate network, and then combines these two networks to form a new network. At each step, the new network's parameters are adjusted using the *quasi-Newton* method. This procedure is performed up to a point where the sum-squared error measure of the obtained network meets the user-defined acceptable level, same as [Spoerre (1997)], or the network size exceeds a pre-defined value. This approach has the advantage of not requiring the initialization of number of hidden layers, and their corresponding weights. However, the network train procedure is more difficult in comparison with Feed-Forward NN and Back Propagation NN.

Applications of ANN in equipment's failure prognostics consider the prognosis as a time series prediction problem where the future state is predicted based on the sequence of the past states. Applying a Recurrent NN, [Yam (2001)] proposed a method to predict the equipment's health condition indicators at the next observation moment. In this method, current and previous health conditions of the equipment represent the input nodes of the Recurrent NN, while the next health condition of the equipment represents the output node of the Recurrent NN. [Sheppard et al. (2005)] employed a *Dynamic Bayesian Network* in order to perform the one-step health condition prediction. Since the equipment's state is unobservable, it employed a *Hidden Markov Model* (*HMM*) to estimate the state. [Przytula et al. (2007)] also employed a Bayesian network. An interesting contribution of its approach is the proposal of a method to convert the graphical probabilistic models into the conditional failure probabilities. [Byington et al. (2004)] employed a Feed-Forward NN to perform the one-step health condition indicator prediction. Then, the predicted health condition indicators are supplied to a FL system in

order to predict the equipment's health condition. Finally, a Kalman filter is employed in order to minimize the prediction errors. [Tian (2012)] employed an ANN to estimate the equipment's RUL. First, using the *Levenberg-Marquardt* algorithm, the ANN is trained by setting both the age and health condition indicators as the model inputs, and the survival probability as the model output. Next, each series of measured health condition indicators is fitted to a Weibull distribution failure rate function. Then, for each newly observed equipment, the failure rate function is updated based on the updated series of measured health condition indicators. Finally, the equipment's survival probability is calculated using the trained ANN, and the current age and health condition indicators of the equipment. Equipment's RUL is calculated based on its survival probability. However, the ANNs have several limitations: First, lack of simple procedure to take the expert knowledge into consideration for practical problems. Second, lack of simple procedure to train ANN for big-size problems. Third, lack of explanatory power of the trained ANN.

Apart from ANNs, some other AI approaches have been proposed in both diagnostics and prognostics. *Genetic Algorithm*, as the most widely used approach in *Evolutionary Algorithm*, is employed for diagnostics in [Sampath et al. (2002)]. Evolutionary Algorithm imitates the mechanisms of biological evolution. The authors simulated all the possible failure cases based on different sets of health condition indicators. Then, an objective function is defined, which represents the difference between the simulated scenarios and the actual ones in an appropriate manner. Eventually, using the Genetic Algorithm technique, the objective function is minimized. This approach has the advantage of possessing highly accurate diagnostics, while the disadvantage of requiring a long run-time. *Logical Analysis of Data* (*LAD*) is another AI approach that has been recently applied in diagnostics. It extracts knowledge hidden in observations of the train set in order to detect the sets of health condition indicators that would lead to either failure or survival of the equipment. [Yacout (2010)] proposed application of LAD in equipment's diagnostics. First, it transforms the health condition indicators from the numerical format into the binary format. Next, it looks through all the failure observations available in the historical data, and tries to find the sets of health condition indicators, called *Pattern*, that are appeared in some of the failure (survival) observations

but not in any of the survival (failure) observations. Then, it constructs a discriminant function by assigning positive weights to the patterns that represent the failure observations, and negative weights to the patterns that represent the survival observations. Eventually, for each newly observed equipment, its corresponding discriminant function value is used as a measure for diagnosis. As a function of failure probability, [Yan et al. (2004)] defined a *Logistic Regression* function. Then, it applied the logistic regression in order to estimate the parameters of the function. Estimation is performed using the *Maximum Likelihood Estimation* (*MLE*). Eventually, using the estimated parameters, it proposed a method to calculate the equipment's RUL based on its current health condition indicators.

1.3.2. Statistical Approaches

Among the statistical approaches, *Hidden Markov Model* (*HMM*) and *Hazard Rate* (*HR*) are two most widely used methods.

Equipment's failure development procedure takes a series of deteriorative steps. This procedure can be formulated in the form of a HMM; a stochastic process model with a normal state, several deteriorative states, and a failure state. HMM is a parametric model, for which many statistical parameter estimation techniques can be employed. [Bunks et al. (2000)] first showed that HMM is a potent tool to be applied in equipment's diagnostics and prognostics. After estimating the parameters of HMM, the authors proposed an approach to model the probability of staying at the normal state (state probability), along with the probability of transition from normal to failure state (state transition probability). [Ying et al. (1999)] proposed an on-line failure diagnostic approach in HMM framework, which is able to deal with the states that are not directly observable, but are imperfectly investigated based on the observed outcomes. Using the MLE method, it proposed a procedure to estimate the parameters of observed outcomes probabilities, along with that of transition probabilities. [Baruah et al. (2005)] introduced a methodology to employ HMM in equipment's failure prognostics. Parameters of HMM are estimated based on the measured historical data. MLE is applied to carry out the parameter estimation. Based on HMM's parameters, the matrix of state transition probability is constructed. Next, the conditional probabilities of transition from any given

state to all the states are estimated. Then, for each health condition indicator associated with the currently observed equipment, HMM estimates the conditional probability of reaching the failure state knowing the current state. Finally, HMM calculates the equipment's RUL. [Zhang et al. (2005)] employed HMM coupled with an adaptive stochastic failure prognostic model. HMM supplies the estimated conditional probabilities to the prognostic model. An adaptive algorithm continuously updates the parameters of the prognostic model with respect to the latest changes in the equipment's conditions. The updated model is employed to calculate the equipment's RUL knowing its current health condition indicators. [Dong et al. (2007)] proposed application of *Hidden Semi-Markov Model* (*HSMM*) for equipment's failure prognostics. HMM and HSMM differ over definition of the state. A state in HMM is defined as a single observation while in HSMM is defined as a sequence of observations. The HSMM's state transition probability matrix is estimated via a forward-backward parameter estimation algorithm. In order to simplify the parameter estimation procedure, the state duration density is assumed to distribute in *Gaussian* form. The equipment's current state is determined using a diagnostic framework. Knowing the equipment's current state, its RUL is calculated using a backward recursive algorithm. [Lin et al. (2004)] investigated the partially observable Markov process, a process in which the states, except for the failure state, are not visible. A recursive formula is proposed in order to estimate the state. Using the *Expectation-Maximization* algorithm, another formula is presented in order to estimate the model parameters. A major limitation of HMM is that it assumes the state probability at each observation moment only depends on the state probability at the prior observation moment. A major limitation of HSMM is that it is a complex model although it modifies the HMM's weakness in definition of state probability.

Hazard Rate (*HR*) represents the ratio of failure probability in a period to the period length, where failure probability represents the proportion of number of failed components during the period to the number of components at the beginning of period [Banjevic et al. (2006)]. HR is a helpful tool in calculation of RUL, which has been employed in many applications in prognostics. [Banjevic et al. (2006)] investigated a Markov process coupled with a HR function. It proposed theoretical and numerical methods in order to calculate the conditional *Reliability Function*. Using the conditional

Reliability Function, it presented a method to estimate the RUL and its expected value, *Mean Residual Life* (*MRL*), based on the current health condition indicators of the equipment. [Cox (1972)] introduced a novel definition for the HR, called *Proportional Hazard Rate* (*PHR*). As opposed to the conventional definition, PHR depends not only on the equipment's age, but also on its health condition. The proposed approach, called *Proportional Hazards Model* (*PHM*), is able to model the relation between the equipment's age and health condition with its failure probability. [Wang (2002)] introduced the concept of *Conditional Residual Delay Time*, which diverges from the traditional concept of *Conditional Residual Time*. The proposed model predicts the equipment's residual life not only based on the equipment's age, but also based on its health condition indicators. The model is fitted to a Weibull distribution, for which a MLE approach is presented for parameter estimation.

Apart from HMM and HR, some other statistical approaches have been proposed in both diagnostics and prognostics. *Statistical Process Control* (*SPC*) is a traditional statistical approach in diagnostics. It was first introduced in the field of quality control, but has been widely applied in equipment's failure diagnostics as well. It decides whether the equipment's health condition indicator is within the control limit or not by comparing the health condition indicators of the equipment with that of the reference normal equipment. [Fugate et al. (2001)] built the control charts based on the health condition indicators of the normal equipment. This approach has the advantage of being executable in an unsupervised train process. Then, it used the control charts to track the changes in the health condition indicators of any equipment from which new observations are collected. [Goode et al. (2000)] applied SPC in equipment's failure prognostics. It divides the equipment's lifetime into two intervals: the *Installation-Potential failure* (*I-P*), called stable zone, and the *Potential failure-Functional failure* (*P-F*), called failure zone. While the equipment is in its stable (failure) zone, the probability of arriving at its potential (functional) failure moment is calculated based on the HR of the Weibull distribution associated to the I-P (P-F) interval. *Cluster Analysis* is another conventional statistical approach in diagnostics. It groups the health condition indicators based on their similarities, and tries to minimize the difference in a group, meanwhile, maximize the difference between the groups. [Skormin et al. (1999)] divided a *n*-dimensional space of

n measured health condition indicators into all of its possible sub-spaces, and for each sub-space, calculated the *Informativity* criterion, which is a function of number of discrimination within and between the normal and failure states. This approach has the disadvantage of performing highly computational efforts. Then, it proposed an approach to select the most informative sub-spaces based on which *Separating Rules* were defined. Separating rules were used to diagnose any equipment, from which new observations were collected, based on the value of its health condition indicators.

2. LOGICAL ANALYSIS OF DATA (LAD)

Logical Analysis of Data (LAD), first introduced in [Crama et al. (1988)], is a combinatorics, optimization and Boolean logic based methodology for the analysis of datasets. The typical aim of LAD is to extract knowledge hidden in observations of a dataset in order to detect the sets of causes that would lead to certain effects. In maintenance, a cause can be the monitored equipment's age or any health condition indicator value, while an effect can be the equipment's survival or failure. Each cause is called an *Attribute*. A *literal* is either an attribute or its *Negation*. Negation of an attribute contradicts the attribute. Based on certain effects, observations are classified into two classes: observations of failure during the coming period, referred to as the *Positive Class*, and observations of survival at least until the end of the coming period, referred to as the *Negative Class*. A *Positive* (*Negative*) *Pattern* is a set of literals that is reflected in one or more of the observations of the positive (negative) class while not reflected in any (many) of the observations of the negative (positive) class. The number of literals forming the pattern is called the *degree* of pattern. A pattern cannot be formed of an attribute and its negation.

The main application of LAD is the pattern-based classification of new observations, which are not classified in the dataset, into either the positive or negative class. Like other recently developed knowledge discovery methodologies, such as AI, *Machine Learning* and *Data Mining*, LAD constructs a classification model based on a given historical dataset, called *Train Set*. Then, by using the classification model, it tests the quality of this model by classifying another part of the historical dataset, called *Test Set*.

Since its introduction, LAD has been widely applied for the analysis of datasets from different fields such as medicine, biotechnology, economics, finance, politics, properties, oil exploration, manufacturing and maintenance.

[Abramson et al. (2005), G. Alexe et al. (2006-1), S. Alexe et al. (2003), and Lauer et al. (2002)] applied LAD in medical fields such as cell growth, breast cancer, coronary risk, and electrocardiography in order to predict behavior of medical models. [G. Alexe et al. (2005), and G. Alexe et al. (2004)] used LAD in medical fields such as B-cell lymphoma, and ovarian cancer in order to diagnose medical diseases. [Yacout (2010), Bennane et al. (2012), Mortada et al. (2012), and Mortada et al. (2011)] applied LAD on industrial equipment such as power transformer, oil transformer, aircraft, and rotor bearing in order to diagnose equipment failure. [G. Alexe et al. (2008), Boros et al. (2000), A. B. Hammer et al. (1999), P. L. Hammer et al. (2006-2), P. L. Hammer et al. (2004-2), and Kim et al. (2008)] applied LAD in various fields such as voting, credit card scoring, housing, labor productivity, country risk, composition of soil in the oil, genotyping, and psychometric in order to discover knowledge from the data and estimate the behavior of the models.

The LAD is composed of five fundamental stages: *Data Binarization*, *Support Set Selection*, *Pattern Generation*, *Pattern Selection*, and *Theory Formation* [Boros et al. (2000)]. The following sections describe basic concepts and different theoretical developments in each stage.

2.1. Data Binarization and Support Set Selection

The original LAD approach was proposed for *Boolean* attribute values. A Boolean represents an expression that takes only the values TRUE and FALSE. However, in many real life problems, the attribute values may appear in numerical form (e.g. temperature), nominal form (e.g. color), or ordered form (e.g. color describing a traffic light). The need of being applicable to any form of attribute values resulted in proposal of many developments for adapting LAD to non-Boolean values or for binarizing data with non-Boolean values to be used with the original LAD. For example, [P. L. Hammer et al. (2004-2)] introduced *Logical Analysis of Numerical Data* (*LAND*) with respect to LAD general concepts, which can deal with numerical values. Aside from the latter example, all the other developments are proposed in the field of data binarization. The *binarization*

procedure transforms each non-binary attribute into several binary ones, by comparing attribute values to certain thresholds called *Cut-Points*. [Boros et al. (1997)] for the first time developed a method for binarizing numerical data in LAD. According to [Boros et al. (2000)], with each numerical attribute, two types of binary attributes can be associated. The first type associates to every cut-point a binary attribute, called *Level Attribute*, and defines it as following:

$$b_{a,c} = \begin{cases} 1 & ; if \quad a \geq c \\ 0 & ; if \quad a < c \end{cases} \tag{1}$$

Where a is the numerical value of attribute, c is the cut-point value, and $b_{a,c}$ is the binary value, associated with a and c. As a result, each numerical attribute is converted to n binary attributes, where n is equal to the number of cut-points.

The second type associates to every pair of cut-points a binary attribute, called *Interval Attribute*, and defines it as following:

$$b_{a,c',c''} = \begin{cases} 1 & ; if \quad c' \leq a < c'' \\ 0 & ; otherwise \end{cases} \tag{2}$$

Where c' and c'' are the cut-point values, and $b_{a,c',c''}$ is the binary value, associated with a and pair of cut-points c' and c''. As a result, each numerical attribute is converted to n binary attributes, where n is equal to the number of pairs of cut-points.

The most crucial task, in binarization process, is to find appropriate cut-points. In the literature, earlier discretization techniques applied simple ideas such as *Equal-width* and *Equal-frequency*, which might lead to information loss if the width or frequency is improperly defined. *Discretization* is the mathematical process of converting continuous values to their equivalent discrete values. Binarization, as a branch of discretization, is aimed at finding a set of cut-points, which divides the range into several intervals in a way that preserves most of the relevant information and keeps the attributes consistent with classes. According to [Kotsiantis et al. (2006)], the objective of any discretization method is to minimize the number of cut-points and the number of inconsistencies. The lower the number of cut-points, the lower the number of binary attributes into which the numerical attribute is transformed. This results in simpler binarized attribute, but might result in loss of the discriminating ability of the binarized attribute too, which is known as the loss of the *Interdependency* between attributes and classes. The optimal binarization method with respect to the number of cut-points and the number of inconsistencies are

called *Simplicity Preferred* and *Consistency Preferred*, respectively. Consistency measure is calculated according to different evaluation functions.

Both goals are achieved at the support set selection stage. *Support Set* is a concept, which was first introduced by [Crama et al. (1988)]. It is defined as a set of binary attributes, which preserves the classification consistency if all the other attributes are removed. The authors introduced a *Set-Covering Problem* to find the minimal support set. A set-covering problem is defined to minimize the number of attributes in the support set while the support set is subject to be composed of at least one of the attributes whose removal leads to identical observations in different classes. [Boros et al. (2000)] modified the set-covering problem in three directions: First, to guarantee that observations of different classes are distinguishable by more than one attribute. Second, to assign weights to objective function based on *Discriminating Power* of an individual binary attribute. Discriminating power shows how well a binary attribute distinguishes observations of different classes. The discriminating power can be defined based on different measures. Third, to assure that cut-points well separate the observations. [Boros et al. (1997)] showed that finding the minimal support set is a NP hard problem. [Chvatal (1979)] proposed a *Greedy Recursive* procedure to find a near optimal support set. The procedure starts with the set of all attributes and tries to remove from the set as many attributes as possible up to the point where removal of the remaining attributes results in identical observations in different classes. In each step, the attribute, whose removal leads to the maximum decrease in the objective function, is removed from the set. [Boros et al. (2000)] introduced a *Greedy Iterative* procedure to construct a near optimal support set. The procedure starts with an empty support set. At each iteration, it adds to the support set an attribute, which leads to the maximum reduction in the objective function. [Boros et al. (2003)] proposed similar greedy algorithms to find a near optimal support set where the optimality is defined with respect to the maximum discriminating power of the set rather than the minimum number of attributes in the set.

As previously mentioned, LAD is mainly applied to construct a classifier based on which new observations, that are not classified in the dataset, are classified into either the positive or negative class. Performance of a classifier is measured based on its *Accuracy* measure, which represents the proportion of number of correctly classified observations

to all classified observations. The better the discretization method, the higher discriminating power of the set of attributes, which is supplied to the classifier. This results in a classifier with the better ability to discriminate observations of different classes, and consequently, a classifier with the more accurate classification. According to [Liu et al. (2002)], in the case that discretization is used to construct a classifier, the accuracy measure, which represents how well the discretization method assists the classifier, is also required to be considered besides the simplicity and consistency measures. In general, some other factors such as discretization time and train time represent the quality of discretization method. Hence, another goal is to find a tradeoff between accuracy and speed. [Liu et al. (2002)] declared that the more discretization time results in the more accurate classifier. The authors also concluded that using discretized data, in comparison with continuous data, leads to 50% reductions in training.

Discretization methods are generally categorized based on different needs: *Supervised vs. Unsupervised* depending on whether discretization takes the class information into consideration or not, *Dynamic vs. Static* depending on whether discretization is done before or during the classification, *Global vs. Local* depending on whether discretization is applied on the entire range or on its sub-partitions, *Splitting vs. Merging* depending on whether discretization starts with an empty set of attributes or the set of all attributes, *Direct vs. Incremental* depending on whether user defines the number of cut-points or the terminating condition [Liu et al. (2002)]. Another categorization classifies the methods as *Chi-square based* methods, which perform a significance test on the relationship between an attribute and the class, *Entropy based* methods, which decide based on the amount of information that each candidate set of attributes provides about the class, *Wrapper based* methods, which decide based on the number of *False Positive* and *False Negative* errors that each candidate set of attributes causes, and *Evolutionary based* methods, which delegate to evolution the decision about the best set of attributes [Kotsiantis et al. (2006)]. As [Liu et al. (2002)] suggested, entropy based methods are the best choices if the goal is only to discretize the data, chi-square based methods are suggested if the goal is to clean off the data from irrelevant attributes, and dynamic methods are suggested for the cases that discretization is followed by a train phase.

An appropriate discretization method should consider the interdependency between the class and the set of attributes [Kotsiantis et al. (2006), and Liu et al. (2002)]. However, most of the mentioned methods consider the interdependency between the class and only an independent attribute in order to avoid the additional complexity, which results in an increase in discretization time [Liu et al. (2002)].

Recently, some methods focusing on data binarization as a preparing phase specifically for LAD classification approach have been proposed [G. Alexe et al. (2006-2), G. Alexe et al. (2006-3), Bruni (2007), and Mayoraz et al. (1999)]. [Almuallim et al. (1991)], and later followed by [Almuallim et al. (1994)], proposed some algorithms to find the minimal subset of attributes with respect to the consistency preference. The presented algorithms include two optimal algorithms, which result in minimal subset of attributes in quasi-polynomial time, and three greedy heuristic algorithms, which prefer less computational effort to optimality. The mentioned greedy heuristics are iterative algorithms. In each iteration, the cut-point, which has the maximum interdependency with the class, is added to the set. [Almuallim et al. (1994)] showed that the greedy heuristics provide an excellent approximation to the optimal algorithms. [Mayoraz et al. (1999)] proposed an approach similar to the simple-greedy heuristic, introduced by [Almuallim et al. (1994)]. The only difference is that the latter approach iteratively removes the cut-point, which has the minimum interdependency with the class, from the set.

All the global algorithms begin by sorting the attribute values in increasing order. Adequate number of cut-points is obtained by only considering those thresholds that are on the boundary of different classes [P. L. Hammer et al. (2006-1)]. Thus, a cut-point is defined as average of two consecutive attribute values, each belonging to different classes. This way, the outcome cut-point represents a boundary, which is able to differentiate between positive and negative classes. This satisfies the consistency goal. The next task is to remove all the redundant cut-points in order to address the simplicity goal. [Mayoraz et al. (1999)] proposed an *Iterative Discriminant Elimination* algorithm, which is based on checking the effect of removing different cut-points on the discriminating power of the cut-points set. [Bruni (2007)] based its decision criterion on the discriminating power of each individual cut-point. This criterion represents how well

an individual cut-point assists the discriminating power of the set of cut-points based on the accuracy of its outcome classifier. In order to measure the accuracy of a classifier, the actual class of each observation is required to be determined so that it can be compared with the classification provided by the classifier. So, accuracy of the classifier cannot be measured until the actual class is available.

[Bruni (2007)] proposed a weighted extension to the original set-covering problem, introduced by [Crama et al. (1988)]. It showed that the proposed extension results in a much shorter solving time in comparison with the original one. Besides, it applied a *Lagrangean Subgradient* heuristic to find a feasible sub-optimal solution for the proposed weighted model, and showed that the heuristic method provides as good classification accuracy as that achieved by the optimal solution. [G. Alexe et al. (2006-3)] based its attribute selection procedure on the generated patterns. It introduced a *Two-step Attribute Selection* procedure, which starts by filtering the attributes based on several criteria. In the second step, the number of times that each attribute is included in the patterns set is considered as a measure of its relevancy. The attributes are ranked based on their relevancy measures, and half of the top-ranked attributes are selected to construct a new model. This procedure is iteratively performed up to the point where the accuracy of the outcome classifier does not progress anymore. The most significant advantage of this procedure is that it considers the interdependency not only between the class and an individual attribute, but also between the class and a set of attributes. [G. Alexe et al. (2006-2)] applied a similar pattern-based attribute selection procedure to that proposed in [G. Alexe et al. (2006-3)]. It showed that the relevancy measure based on *Spanned* patterns is preferred to that based on *Prime* patterns. We shall describe spanned and prime patterns in detail in the following section.

2.2. Pattern Generation and Pattern Selection

A pattern is a set of literals that is reflected in one or more of the observations of its class while not reflected in any (many) of the observations of the opposite class. An observation will be considered *Covered* only if all the literals forming the pattern are reflected in the observation. Two most significant measures affecting performance of LAD are *Homogeneity* and *Prevalence* of the patterns [Boros et al. (2000)]. Associated

with each pattern, homogeneity is defined as the proportion of number of correctly covered observations to all covered observations by the pattern, and prevalence is defined as the proportion of number of covered observations by the pattern to all observations [Chvatal (1979)]. The former measure demonstrates how precisely a pattern can distinguish observations of its class from the opposite class, while the latter measure demonstrates the pattern's ability to detect observations of both classes.

Given the binarized attributes of observations, the most crucial phase of the LAD method is to develop patterns that reflect the characteristics of corresponding class well. Concurrently to LAD introduction, [Crama et al. (1988)] introduced two methods to generate prime patterns. A pattern is called prime if any shortened set of its literals, which is obtained by eliminating one of the literals, is not a pattern anymore. The first method is based on solving a set-covering problem whose minimal solutions are all the possible prime patterns. The second method is based on enumerating all combinations of literals of limited degree and examining whether each of the combinations can be considered as a pattern. The second method is performed in a polynomial time only if the number of literals forming a combination is given. However, it requires a huge computational effort.

[Boros et al. (2000)] introduced two techniques for the enumeration of all prime patterns, which are called *Top-Down* and *Bottom-Up*. The top-down approach associates the literals that form the observation to a pattern, and eliminates the *Redundant* literals. The redundant literal is an expression that is used for a literal, which if removed from the pattern the result is still a pattern. The bottom-up approach favors shorter patterns. It starts by finding patterns that are only composed of one literal. If such a literal exists in some of the positive (negative) observations while does not exist in any of the negative (positive) ones, it is considered as a prime pattern. Otherwise, it looks through all possible combinations of two literals, and attempts to find a combination that forms a pattern. It keeps adding literals one by one, up to the point that all observations are covered.

[Boros et al. (2000)] proposed a hybrid bottom-up top-down approach, which favors the bottom-up strategy up to a pre-defined maximum degree, and after that, applies the top-down strategy for all the observations that are not covered using the bottom-up strategy.

Finally, at the pattern selection stage, any pattern, whose set of covered observations is a subset of that of any other patterns, is considered as a redundant pattern and is removed from the patterns set. [S. Alexe et al. (2006)] also proposed a polynomial time algorithm for the enumeration of all prime patterns of limited degree. The algorithm eliminates the binarization stage, and deals with all possible intervals in the range of numerical attributes. Each interval is considered as a potential pattern, and its corresponding prevalence is calculated. The algorithm defines prevalence matrix, whose elements are the prevalence corresponding to all possible intervals. Then, prime patterns are detected based on iterative calculation of prevalence matrices. The authors showed that the proposed method results in the generation of the set of low degree patterns in a very short time.

Since LAD introduction, several studies have focused on generation of various types of patterns such as prime, spanned, *Strong*, and *Maximum*, all of which will be described in detail in the following. For this purpose, [G. Alexe et al. (2008), Boros et al. (2000), S. Alexe et al. (2006), G. Alexe et al. (2006-4), Bonates et al. (2008), P. L. Hammer et al. (2004-1), and Ryoo (2009)] developed different algorithms and examined the accuracy of generated patterns.

[P. L. Hammer et al. (2004-1)] defined three preferences of *Simplicity*, *Selectivity*, and *Evidence* in order to compare suitability of different types of patterns. A pattern P_1 is simplicity-wise preferred to a pattern P_2 if the set of literals in P_1 is a subset of that in P_2. Thus, the prime pattern, introduced in [Crama et al. (1988)], is the optimal pattern regarding the simplicity preference. The simplicity preference leads to reduction in number of false negative errors (incorrectly cover observations from the opposite class), but it does not guarantee the reduction in the number of false positive errors (fail to cover observations of its class). As opposed to a widespread belief in machine learning, [P. L. Hammer et al. (2004-1)] concluded that the simplicity preference would not result in a good performance. A pattern P_1 is selectivity-wise preferred to a pattern P_2 if the set of observations incorrectly covered by P_1 is a subset of that by P_2. The selectivity preference is exactly contrary to the simplicity preference. Hence, it guarantees the reduction in number of false positives errors (fail to cover observations of its class), while it may result in increase in number of false negative errors (incorrectly cover observations from

the opposite class) too. A pattern P_1 is evidentially preferred to a pattern P_2 if the set of observations covered by P_2 is a subset of that by P_1. The optimal pattern with respect to the evidential preference is called strong pattern. [P. L. Hammer et al. (2004-1)] concluded that the evidential preference would result in a good performance. Interestingly, if a pattern P_1 is simplicity-wise preferred to a pattern P_2, it would also be evidentially preferred to the pattern P_2. Aside from three mentioned preferences, the authors also defined a new type of preference based on the combination of selectivity and evidential preferences, and defined the optimal pattern regarding the combination of these two preferences as spanned pattern. Strong patterns that are modified with respect to simplicity preference are called strong prime patterns, while those that are modified with respect to selectivity preference are called strong spanned patterns. Moreover, [P. L. Hammer et al. (2004-1)] proposed three different polynomial time algorithms to transform any pattern to a prime, or strong, or spanned pattern. The strong spanned patterns benefit from the property that restricts their number from reaching that of other type of patterns, and this makes them the most attractive in comparison to the other type of patterns. The classifiers based on the strong spanned patterns possess a lower number of classification errors, while those based on the strong prime patterns possess a lower number of unclassified observations.

Patterns with lower degree have advantage of covering more observations of its class but have the disadvantage of covering more observations of the opposite class too. This type of patterns is called *Comprehensible* patterns. Patterns with higher degree have disadvantage of covering less observations of its class but have the advantage of covering less observations of the opposite class too. This type of patterns is called *Comprehensive* patterns. Prime patterns have lower degree, and are comprehensible patterns while spanned patterns have higher degree, and are comprehensive patterns [G. Alexe et al. (2008)]. Due to their inherent characteristics, [G. Alexe et al. (2006-2)] suggested prime patterns for the purpose of classification, while it suggested spanned patterns for the purpose of attribute ranking. The authors also proposed a polynomial time algorithm for the generation of all spanned patterns. According to [G. Alexe et al. (2006-4)], the significant advantage of the proposed algorithm is that an increase in the number of cut-points does not affect the algorithm complexity while, as reported by [Ryoo (2009)], it

affects the algorithm complexity in the case of generating prime patterns. [G. Alexe et al. (2006-4)] also concluded that spanned patterns are preferred to prime patterns in the case of classifying low-quality datasets.

[G. Alexe et al. (2008)] proposed two different polynomial time algorithms for the generation of all strong prime patterns and strong spanned patterns. It focused on comparing large and comprehensive sets of patterns with small and comprehensible sets of patterns provided by both strong prime and strong spanned strategy. Performance of classifiers with small and comprehensible sets of patterns is almost as good as that of those with larger and comprehensive sets of patterns. It also concluded that finding the best choice, between strong prime patterns and strong spanned patterns, greatly depends on the datasets. Choosing optimal parameter values with respect to homogeneity and prevalence is also dataset dependent.

A maximum pattern is a pattern whose coverage is the maximum [Bonates et al. (2008)]. Patterns that possess higher coverage can better cover new observations in comparison with those whose coverage is lower [Boros et al. (2000)]. [P. L. Hammer et al. (2006-1)] formulated the problem of generation of the maximum patterns as an *Integer Linear Programming* (*ILP*) set-covering problem. [Bonates et al. (2008)] proposed four heuristic algorithms for approximation of the ILP problem along with the exact solution of the polynomial set-covering problem to generate maximum patterns. It concluded that the classifiers, based on the heuristically generated patterns, provide as good performance as those based on the ILP generated patterns. [P. L. Hammer et al. (2004-2)] applied *Branch-and-Bound* algorithm in order to construct a maximum pattern. It reported an outstanding performance of the classifier with respect to both homogeneity and prevalence measures. It also showed that the number of observations in the test set, which are classified as both positive and negative class, is almost zero.

[Ryoo (2009)] introduced a *Mixed Integer Linear Programming* (*MILP*) approach to generate different types of optimal patterns including strong, strong prime, strong spanned, maximum, maximum prime, maximum spanned, and optimal patterns with specified degree. It showed that the MILP approach results in an improvement in the efficiency of LAD classifier. A major advantage of MILP approach is that it finds the minimum number of patterns required to construct a classifier [Ryoo (2009)]. [Mortada

(2010)] modified the strong pattern generation approach, introduced by [Ryoo (2009)], in order to increase the discriminating ability of the patterns set. To do so, the value of *Discriminating Factor* is pre-defined to show the minimum number of patterns required to cover each observation. Then, the iterative MILP approach terminates only if all the observations are covered by at least the pre-defined number of patterns.

2.3. Theory Formation

At the train phase, all patterns are generated based on the observations in the train set. At the test phase, each of the patterns individually indicates the characteristics of observations in the test set. Hence, a sufficient number of patterns collected in the patterns set can provide a good indicator of characteristics of new observations in a way consistent with the historical observations. The patterns set is consequently used to construct a classification rule which is called a *Theory*. As the original version of LAD defined [Crama et al. (1988)], a new observation is classified as positive (negative) only if it is covered by some of positive (negative) patterns and not covered by any of negative (positive) patterns. As a result of this definition, none of the new observations, which are covered by some positive (negative) patterns and some negative (positive) patterns, can be classified.

[Boros et al. (2000)] modified the original LAD definition to adapt LAD to deal with the observations with the mentioned condition. It defined the following *Discriminant Function*, based on the relative weight of generated patterns, as a tool to classify new observations.

$$Discriminant\ Function\ = \sum_{k=1}^{r} H_k^+ PP_k + \sum_{l=1}^{s} H_l^- NP_l \qquad (3)$$

Where $PP_1,\ldots,PP_r$ and $NP_1,\ldots,NP_s$ represent respectively the positive and the negative patterns. And H_k^+ and H_l^- represent respectively non-negative weights for the positive and non-positive weights for the negative patterns. For any new observation, a positive (negative) value of the discriminant function value indicates that it is classified as a positive (negative) observation. This way, new observations covered by both positive and negative patterns can also be classified unless the value of their discriminant function equals to zero. As [S. Alexe et al. (2007)] reported, 99% of the observations, which were not classified by the original LAD classifier, are classified by the modified LAD

classifier, among which, 80% are classified correctly. Hence, the modification resulted in an increase in both number of classified observations and classification accuracy.

The weights can be determined in several ways. [Boros et al. (2000)] suggested three approaches to determine the value of weights: First, consider the number of observations covered by each pattern to define the pattern's relative weight. Second, consider the normalized inverse of the degree of each pattern as the pattern's relative weight. Third, solve a linear program to decide on the value of weights regarding the maximization of the discriminating power. [P. L. Hammer et al. (2006-1)] proposed a quadratic program to decide on the value of weights minimizing the overlap between positive and negative patterns.

2.4. Developments

One of the aspects of LAD developments is the introduction of concept of *Boxes* [P. L. Hammer et al. (2004-2), and Eckstein et al. (2002)]. The box concept is used in LAND: a development in LAD that can directly deal with numerical values. The box concept in LAND is equivalent to the pattern concept in LAD. The box concept was first introduced by [P. L. Hammer et al. (2004-2)] in an approach, which was purposed on adapting LAD to the numerical attributes. According to their approach, an individual box is analyzed with respect to two performance measures: homogeneity, which is the same definition as it is in LAD, and *Domain*, which is the equivalent to prevalence definition in LAD. A family of boxes is analyzed with respect to another performance measure: accuracy of a family of boxes, which is the equivalent to homogeneity of an individual box. The homogeneity is the most significant performance measure of a box, while the accuracy is the most significant performance measure of a *Saturated* family of homogeneous boxes. A family of boxes is called saturated if the merger of any two of its members covers an observation from the opposite class. [P. L. Hammer et al. (2004-2)] applied their approach on two types of datasets: *Clean Datasets*, referred to datasets for which favorable results have been provided by other methods, and *Blurred Datasets*, referred to datasets for which unfavorable results have been provided by other methods. [Boros et al. (1997)] concluded that the accuracy of the boxes families is outstanding for clean datasets, while it is still promising for blurred datasets. It also concluded that the *Overlap*

of the boxes families, which represents the proportion of boxes families that appear in both positive and negative classes, is insignificant for clean datasets, while it is significant for blurred datasets. [Eckstein et al. (2002)] also applied the box concept in order to solve heuristically the *Maximum Box* (*pattern*) *Problem*. A maximum box problem is defined to maximize the number of observations in the pattern's class that are covered by the box while the box is subject to not cover observations from the opposite class.

One other aspect of LAD developments is the relaxation of the homogeneity or prevalence requirements [G. Alexe et al. (2008), Bonates et al. (2008), and Ryoo (2009)]. In the original LAD, the homogeneity of patterns over the train set is restricted to 1, which means that patterns are not allowed to cover observations from the opposite class. By its relaxation, observations from the opposite class are allowed to cover by patterns [G. Alexe et al. (2008)]. Associated with each pattern, *Fuzziness* is defined as the proportion of number of observations covered from the opposite class to all covered observations by the pattern [Bonates et al. (2008)]. [Ryoo (2009)] also proposed several MILP approaches with respect to the relaxation of the homogeneity or prevalence.

Another aspect of LAD developments is the consideration of imperfect input data [Bennane et al. (2012), Boros et al. (2000), Boros et al. (1999)]. [Boros et al. (2000)] categorized imperfections of datasets into three groups: *Classification Errors*, *Measurement Errors*, and *Missing Data*. It also introduced some solutions to address each of these imperfections. In the case of classification errors, it suggested that a pattern is considered as positive (negative) only if the proportion of negative (positive) observations to positive (negative) observations does not exceed a pre-defined threshold. In the case of measurement errors, it suggested that a non-binary value to be considered as missing value if it is in the neighborhood of a cut-point. To deal with missing data, the attributes, whose values are unknown, are suggested to not take part in the constraints. This corresponds to both positive and negative observations.

[Bennane et al. (2012)] also proposed several methods for estimating the missing data. It concluded that the selection of the best method for estimating the missing data is case-dependent. [Boros et al. (1999)] also proposed several polynomial-time algorithms to deal with the missing data in various classes of Boolean functions.

Another concept related to the missing data is the *Outliers* [Bennane et al. (2012), and Han et al. (2011-2)]. [Bennane et al. (2012)] defined outliers as observations which are in contrast to most of the other observations. The outliers should be removed from the dataset and be treated as missing data. [Boros et al. (1997)] proposed a pattern selection problem, which takes the concept of outliers into consideration. After solving the problem, all of the patterns, whose coverage is lower than a pre-defined threshold, are removed from the pattern set, and the corresponding observations are treated as outliers.

Some researchers [S. Alexe et al. (2003), Lauer et al. (2002), and Kronek et al. (2008)] used information hidden in patterns to estimate survival function. [Lauer et al. (2002)] first applied LAD to estimate the survival function based on *Kaplan-Meier* (*KM*) method [Kaplan et al. (1958)]. Using the survival function, the authors calculated the proportion of the actual death rate to the predicted one, and compared it with that based on *Cox Regression Model* [Cox (1972)]. It concluded that LAD moderately outperforms Cox regression model, while LAD also benefits from the advantage that it is not based on any assumption regarding the input data. [S. Alexe et al. (2003)] also applied the same procedure, and concluded that LAD prognostic score provides as accurate prediction as that provided by Cox score, while LAD also benefits from the advantage that its score is more informative. [Kronek et al. (2008)] called the mentioned approach *Logical Analysis of Survival Data* (*LASD*), and applied it for the case of right-censored data. For this purpose, the authors introduced a heuristic algorithm for the generation of maximum survival patterns with the relaxation of homogeneity. The authors also proposed two survival function estimators with respect to both time-based survival, and condition-based survival. By comparing to some other estimators, the authors concluded that LAD moderately outperforms *Survival Trees* and *KM Estimators*.

Some researchers have extended LAD with respect to the number of distinguishable classes [Yacout (2010), and Moreira (2000)]. The original LAD [Crama et al. (1988)] was designed for the purpose of classification of observations into either positive or negative class. However, [Moreira (2000)] introduced an approach to handle the discrimination of more than two classes based on the main concept of LAD. The approach generates a pattern based on a pair of classes, and evaluates the relation between the pattern and the remaining classes. Accordingly, it associates to the pattern

several values, each of them showing the relation between the pattern and one of the classes. Finally, an observation in the test set is classified based on the corresponding values of different classes associated to its covering patterns.

2.5. Performance Comparisons

LAD has been widely applied for the analysis of datasets from different fields such as medicine, biotechnology, economics, finance, politics, properties, oil exploration, manufacturing and maintenance. The accuracy of the results provided by LAD has been compared with that provided by other classification methods. According to [S. Alexe et al. (2007)], LAD classifier provides more accurate and less deviated classification in comparison with the *Fisher Discriminant Classifier*. The authors also compared accuracy of LAD with that of several data mining classifiers such as *Neural Networks* and *Classification Trees*, and showed that it outperforms the best of other methods in 75% cases. [Bonates et al. (2008)] compared the accuracy of LAD with that of five widely used machine learning algorithms including *Support Vector Machines*, *Decision Trees*, *Random Forests*, *Multilayer Perceptron*, and *Simple Logistic Regression*. It concluded that, with respect to accuracy, it is difficult to tell apart LAD from Support Vector Machines and Random Forests algorithms, but it outperforms the Decision Trees, Multilayer Perceptron and Simple Logistic Regression algorithms. Applying LAD on different datasets, [Boros et al. (2000)] compared the prediction rate of LAD in each dataset with that of the best found in the literature for each dataset. It is worth mentioning that each of the best methods in the literature is only successful in a specific dataset, which means that their success is case dependent. LAD's result compares approvingly with the best-reported results in the literature. [Han et al. (2011-1)] used LAD's generated patterns as the input to other classification methods such as Decision Trees, and *K Nearest Neighborhood*. It claimed that if LAD, with accuracy of more than 70%, is supplied to these classification methods, it leads to an increase in the accuracy of their outcomes.

3. OBJECTIVE OF RESEARCH

The main objective of this research is to develop an equipment failure prognostics model based on LAD. We will also analyze the performance of the proposed LAD prognostic model by applying it on a real dataset, and will investigate its advantages and disadvantages. We will analyze the effect of using different methods, in the context of LAD, and their corresponding parameters on the quality of prognostic results obtained by these methods. Performance of the prognostic model will also be compared with that of a widely used prognostic model, called Proportional Hazards Model (PH Model).

CHAPTER 2 : **METHODOLOGY**

In this chapter, we will illustrate LAD's basic steps in the context of CBM by applying different techniques on a sample condition monitoring dataset. Patterns, LAD's modeling outcomes that characterize the failure and survival characteristics of equipment in the dataset, will be generated. Then, we will present a guideline to use the generated patterns for equipment's failure prognostics.

Table 1 shows a sample set of monitored data. The set is composed of the monitored data, including different observation moments, associated with different pieces of equipment, and their corresponding attributes.

Table 1. Sample Set of Monitored Data

Observations			Attributes	
Equipment ID.	Observation Time	Class	Age	Condition Indicator
1	0	-	0	14
1	1	-	1	16
1	2	-	2	20
1	3	-	3	18
1	4	+	4	20
2	0	-	0	12
2	1	-	1	18
2	2	+	2	22
3	0	-	0	16
3	1	-	1	18
3	2	-	2	20
3	3	+	3	22

Each row corresponds to an observation moment, for which the equipment identification and the observation time are respectively shown in the first and the second columns. The third column shows the class of each observation. The last observation moment of each piece of equipment, referred to as the observation that will fail during the current period, is shown with the dark background. The forth and the fifth columns respectively show the measurements of age and condition of equipment. Unlike the earlier introduced LAD methodology [Yacout (2010), Mortada et al. (2012), and Mortada et al. (2011)], we consider both age and condition of equipment as the equipment's attributes, and use both of them as LAD attributes. This approach provides the ability to calculate the *Survival Function*. Herein, our focus is not on the detection of the failure, which is the diagnostics objective of CBM, but on calculation of the probability of failure at certain moment in future, which is the prognostics objective of CBM, and has been comparatively untested. In this chapter, we will improve the LAD methodology to predict equipment's chance of

survival at each observation moment when new data on attributes of the equipment is available.

This chapter is structured as follows: Section 1 will describe two data binarization methods. Section 2 will describe two pattern generation methods. A measure for evaluating the quality of generated patterns will be described in section 3. Section 4 will describe a method to diagnose the new observations, and introduce two methods to prognosticate the new observations.

1. DATA BINARIZATION

As mentioned in the previous chapter, there are several approaches to define cut-points in the literature. In the following sections, we will describe how *Sensitive Discriminating* method and *Equipartitioning* method define the cut-points.

1.1. Sensitive Discriminating Method

The sensitive discriminating method begins by sorting the attribute values in increasing order. This method favors reinforcement of discrimination ability of cut-points. Therefore, a cut-point is defined as average of two consecutive attribute values, each belonging to different classes. The outcome cut-point represents a threshold, which is able to differentiate between positive and negative classes.

For the sample set of monitored data shown in the Table 1, sorted age attribute and sorted condition attribute are respectively shown in Table 2 and 3, along with their corresponding classes. The cells with the dark background are attribute values that correspond to a change in the class of observations. As it can be inferred from the Table 2 cut-points with respect to the age attribute, are defined between 1 and 2, 2 and 3, and 3 and 4, which results in 1.5, 2.5, and 3.5, respectively.

Table 2. Sorted Age Attribute and Its Corresponding Classes

Sorted Attribute	
Age	Class
0	-
1	-
2	- , +
3	- , +
4	+

Similarly, the Table 3 shows that cut-points with respect to the condition attribute, should be defined between 18 and 20, and 20 and 22, which results in 19, and 21, respectively.

Table 3. Sorted Condition Attribute and Its Corresponding Classes

Sorted Attribute	
Condition Indicator	Class
12	-
14	-
16	-
18	-
20	- , +
22	+

Table 4 shows the sample set of monitored data in the Table 1 transformed into binary format, using *eq.* (1) and the cut-points defined by sensitive discriminating binarization method. As it is shown in the Table 4, the age attribute is transformed into three binary attributes with respect to its cut-points (i.e.: 1.5, 2.5, and 3.5). Similarly, the condition attribute is converted into two binary attributes with respect to its proper cut-points (i.e.: 19, and 21).

Table 4. Binary Transformation of Sample Set of Monitored Data

Observations		Attributes				
Equipment ID.	Observation Time	Age			Condition Indicator	
		1.5	2.5	3.5	19	21
1	0	0	0	0	0	0
1	1	0	0	0	0	0
1	2	1	0	0	1	0
1	3	1	1	0	0	0
1	4	1	1	1	1	0
2	0	0	0	0	0	0
2	1	0	0	0	0	0
2	2	1	0	0	1	1
3	0	0	0	0	0	0
3	1	0	0	0	0	0
3	2	1	0	0	1	0
3	3	1	1	0	1	1

1.2. Equipartitioning Method

Like the former method, equipartitioning method also begins by sorting the attribute values in increasing order. But, this method follows an equal-frequency strategy. The cut-points are defined in such a way that all the attribute values are approximately equally divided into a pre-defined number of intervals. Appropriate number of intervals is selected by comparing the quality of results associated with different values. In this work, we set the interval number close to that provided by the sensitive discriminating method so that the effect of different binarization methods on the quality of results is comparable.

In our experiments, which will be discussed later in chapter 3, the interval number provided by the sensitive discriminating method varies between 30 and 40, according to different train sets. Therefore, for the equipartitioning method, the interval numbers are set equal to 20, 30, 40, and 50, as will be shown later in chapter 6 section 3.

For the sample set of monitored data shown in the Table 1, sorted age attribute and sorted condition attribute are respectively shown in Table 5 and 6. In this example, the number of intervals is set to three. So, the original twelve observations have to be placed in three groups, each consisting of four observations. Different groups are indicated by light and dark backgrounds. Borders of the groups, which are shown by arrow signs, specify the proper place of cut-points. As it is shown in the Table 5, cut-points with respect to the age attribute, are defined between 1 and 1, and 2 and 2, which results in 1, and 2, respectively. Similarly, the Table 6 shows that cut-points with respect to the condition attribute should be defined between 16 and 18, and 20 and 20, which results in 17, and 20, respectively.

Table 5. Sorted Age Attribute

Sorted Attribute
Age
0
0
0
1
1
1
2
2
2
3
3
4

Table 6. Sorted Condition Attribute

Sorted Attribute
Condition Indicator
12
14
16
16
18
18
18
20
20
20
22
22

Table 7 shows the sample set of monitored data in the Table 1 transformed into the binary format, using *eq.* (1) and the cut-points defined by equipartitionting binarization method. The age attribute is transformed into two binary attributes with respect to cut-points 1, and 2. Similarly, the condition attribute is converted to two binary attributes with respect to cut-points 17, and 20.

Table 7. Binary Transformation of Sample Set of Monitored Data

Observations		Attributes			
Equipment ID.	Observation Time	Age		Condition Indicator	
		1	2	17	20
1	0	0	0	0	0

1	1	1	0	0	0
1	2	1	1	1	1
1	3	1	1	1	0
1	4	1	1	1	1
2	0	0	0	0	0
2	1	1	0	1	0
2	2	1	1	1	1
3	0	0	0	0	0
3	1	1	0	1	0
3	2	1	1	1	1
3	3	1	1	1	1

2. PATTERN GENERATION

A pattern discriminates one or more of the observations of its class from all or most of the observations of the opposite class. The basic pattern generation algorithms are mainly based on generating all combinations of literals, and examining whether each of the combinations can be considered as a pattern. This results in a huge computational effort.
Recently, some heuristic methods have been introduced that require less computational effort while providing equivalent performance. In these methods, instead of generating all combinations of literals, only the most preferred combinations are generated at the initial step. If any of the combinations is considered as a pattern, the method stops. Otherwise, the less preferred combinations are generated, and this series of steps is performed up to a point that a pattern is detected. As mentioned in the previous chapter, preference might be given to different types of combinations based on the usage of generated patterns. Among all the proposed pattern generation algorithms, we will use *Mixed Integer Linear Programming* (*MILP*) method and *Hybrid Greedy* method.

2.1. Mixed Integer Linear Programming (MILP) Method

MILP-based pattern generation method, first introduced by [Ryoo (2009)], develops a Mixed Integer Linear Programming and formulates a linear set-covering problem to generate different types of patterns. He describes an algorithm for obtain optimal *Strong Pure* patterns. A pattern is strong if the set of observations covered by the pattern is not a subset of that covered by other patterns. A pattern is pure if it does not cover any observation from the opposite class.
According to [Ryoo (2009)], the binary variable vector $w_{pattern}(w_1,w_2,\ldots,w_{2q})$, corresponding to each binarized pattern, is defined such that $w_j=1$ if attribute j is included

in the pattern and w_j=0, otherwise. Where q is the number of binarized attributes. The set of (w_{q+1},w_{q+2},…,w_{2q}) is defined such that w_{q+j}=1 if negation of attribute j is included in pattern, and w_{q+j}=0, otherwise. This way, the characteristics of a class can be represented not only by values that are greater than a certain cut-point, but also by values that are lower than that too.

The MILP pattern generation model is formulated as a set-covering problem. The objective of the model is to generate a pattern that leads to the minimum number of observations in certain class, which are not covered by the generated pattern. Then, by modifying the previous model, different patterns are generated one by one, up to a point that all the observations are covered by at least one pattern.

Variable y_i is defined such that y_i =1 if observation i is not covered by the pattern, and y_i =0, otherwise. Since the optimality of a strong pattern is measured with respect to its coverage, the objective is to generate a pattern with the maximum number of covered observations, or in other words, with the minimum number of uncovered observations. So, the objective function is defined to minimize $\sum_{i\in S^*} y_i$, while satisfying the following conditions:

- An observation will be considered *covered*, only if all the attributes forming the pattern are reflected in the observation. Thus, constraint $\sum_{j=1}^{2q} b_{ij}w_j + qy_i \geq d$ should hold for all observations in the pattern's class (S^*), where $d = \sum_{j=1}^{2q} w_j$ is known as the degree of a pattern. By defining b_{ij} =1 if attribute j exists in observation i, and b_{ij} =0 otherwise, $\sum_{j=1}^{2q} b_{ij}w_j$ counts the number of attributes common to both the observation and the pattern. If an observation is covered by the pattern, then $\sum_{j=1}^{2q} b_{ij}w_j = d$. Otherwise, the model will satisfy the constraint by setting y_i =1. This is due to the fact that $q \geq d$.
- An observation will be considered *not covered*, if there is at least an attribute forming the pattern that is not reflected in the observation. Consequently, the constraint $\sum_{j=1}^{2q} b_{ij}w_j \leq d - 1$ should hold for all observations in the opposite class (S^{*-}). This constraint guarantees the pattern is pure.
- A pattern should not be formed of an attribute and its negation, which means that $w_j + w_{q+j} \leq 1$ should hold for all attributes.

Considering the above-mentioned conditions, the following MILP model is introduced to find a strong pure pattern [Ryoo (2009)], where N^* is the number of observations in the pattern's class.

Min $\sum_{i \in S^*} y_i$

subject to.

$$\sum_{j=1}^{2q} b_{ij} w_j + q y_i \geq d \qquad \forall i \in S^* \qquad (4)$$

$$\sum_{j=1}^{2q} b_{ij} w_j \leq d - 1 \qquad \forall i \in S^{*-} \qquad (5)$$

$$w_j + w_{q+j} \leq 1 \qquad j = 1,2,\dots,q \qquad (6)$$

$$\sum_{j=1}^{2q} w_j = d \qquad (7)$$

$$1 \leq d \leq q \qquad (8)$$

$w \in \{0,1\}^{2q}$

$y \in \{0,1\}^{N^*}$

For instance, the binarized dataset provided in the Table 4 is formulated as the following MILP model in order to generate a positive strong pure pattern.

Min $y_1 + y_2 + y_3$

subject to.

$$w_1 + w_2 + w_3 + w_4 + w_{10} + 5y_1 - d \geq 0 \qquad 1$$

$$w_1 + w_4 + w_5 + w_7 + w_8 + 5y_2 - d \geq 0 \qquad 2$$

$$w_1 + w_2 + w_4 + w_5 + w_8 + 5y_3 - d \geq 0 \qquad 3$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 4$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 5$$

$$w_1 + w_4 + w_7 + w_8 + w_{10} - d \leq -1 \qquad 6$$

$$w_1 + w_2 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 7$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 8$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 9$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 10$$

$$w_6 + w_7 + w_8 + w_9 + w_{10} - d \leq -1 \qquad 11$$

$$w_1 + w_4 + w_7 + w_8 + w_{10} - d \leq -1 \qquad 12$$

$$w_1 + w_6 \leq 1 \qquad 13$$

$$w_2 + w_7 \leq 1 \quad (14)$$

$$w_3 + w_8 \leq 1 \quad (15)$$

$$w_4 + w_9 \leq 1 \quad (16)$$

$$w_5 + w_{10} \leq 1 \quad (17)$$

$$w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 + w_8 + w_9 + w_{10} - d = 0 \quad (18)$$

$1 \leq d \leq 5$

$w_1, \ldots, w_{10} \in \{0,1\}$

$y_1, \ldots, y_3 \in \{0,1\}$

The constraints (1-3) and (4-12) respectively correspond to positive and negative observations. The constraints (13-17) display relation between attributes and their negations in the pattern formation. The constraint 18 shows the degree of the pattern.

For instance, as it is shown in Table 4, the positive observation 2 is formed of the literals (w_1, w_4, w_5) and negations of the literals (w_2, w_3). So this observation is shown in the form (w_1,w_4,w_5,w_7,w_8). The following inequality shows the constraint 2 corresponding to the positive observation 2.

$w_1 + w_4 + w_5 + w_7 + w_8 + 5y_2 - d \geq 0$

If the set of literals forming the pattern is a subset of this observation, the equality $w_1 + w_4 + w_5 + w_7 + w_8 = d$ is held. Therefore, the constraint is simplified to $5y_2 \geq 0$, which means that y_2 can take both the values 0 and 1. Since the objective is to minimize $\sum_{i \in S^*} y_i$, the model sets y_2 equal to 0, and consequently, the observation 2 is considered as *covered*. Otherwise, the inequality $w_1 + w_4 + w_5 + w_7 + w_8 - d = x < 0$ is held. Therefore, the constraint is simplified to $5y_2 + x \geq 0$, where $x < 0$. This means that y_2 can take only the value 1, and consequently, the observation 2 is considered as *uncovered*.

By solving the above-mentioned model, the following results are obtained.

Objective Function =1

w_4 =1 w_5 =1 y_1 =1 d =2 *all others* =0

The results show that a pattern of degree two has been generated which is composed of literals w_4 and w_5. This means that satisfaction of both *condition indicator value* ≥ 19 and *condition indicator value* ≥ 21 represents the characteristic of the positive observations in the example. The results also show that the positive observation 1 has not been covered

by the generated pattern (y_1 =1). Thus, the model should be reconstructed so that positive observation 1 gets covered by at least one pattern.

Reconstructing the Linear Programing is performed to generate more patterns, up to a point that all the observations are covered by at least one of the patterns. It is worth mentioning that, although each generated pattern differs from previously generated ones, yet it might cover the some or all of the previously covered observations while some remaining observations are still uncovered. This will result in generating redundant pattern while no more uncovered observation gets covered. In order to avoid generating redundant patterns, all the observations that were previously covered by generated patterns will be removed before reconstructing the model.

In order to prevent the model from generating the same pattern twice, a new constraint is required to be added to the model, after a pattern is generated. To do so, [Mortada (2010)] suggested to save the solution vector w_i ($w_{i,1}$, $w_{i,2}$,… $w_{i,2q}$), associated with generated pattern i, in such a way that:

$$r_{i,j} = \begin{cases} 1 & ;if \quad w_{i,j} = 1 \\ -1 & ;if \quad w_{i,j} = 0 \end{cases} \tag{9}$$

Assume that attribute $j \in \{1,2,\ldots,2q\}$ was already included in previously generated pattern i, ($r_{i,j}$ =1). If it is included in currently being generated pattern c too, ($w_{c,j}$ =1), then $r_{i,j} \times w_{c,j}$ =1 represents that attribute j is common to both pattern i and c. Otherwise if $w_{c,j}$ =0, then $r_{i,j} \times w_{c,j}$ =0, which represents that attribute j is not common between the patterns i and c. Then, $\sum_{j=1}^{2q} r_{i,j} w_{c,j}$ counts the number of attributes common to both pattern i and c. For each generated pattern $i \in \{1,2,\ldots,N\}$, where N is the number of previously generated patterns, the constraint $\sum_{j=1}^{2q} r_{i,j} w_{c,j} \leq d_i - 1$ [Mortada (2010)] prohibits the number of common attributes from being equal to the number of attributes included in pattern i, which means pattern i and c will not be identical.

The following constraint will be added to the model to constrain the model from generating the same pattern.

$- w_1 - w_2 - w_3 + w_4 + w_5 - w_6 - w_7 - w_8 - w_9 - w_{10} - d \leq -1$

As mentioned previously, constraints 2 and 3, correspond to the positive observations 2 and 3. Because these two observations were covered by the generated pattern during first run of the model, they will be removed from the next model.

By solving the reconstructed model, the following results are obtained.

Objective Function =0

w_3 =1 d =1 *all others* =0

The results show that a pattern of degree one has been generated which is composed of the literal w_3 only. This means *age* ≥ 3.5 represents positive observations uniquely. The results also show that there is no positive observation left uncovered.

Table 8 shows all the positive and negative patterns generated for the binarized dataset provided in Table 4. The cells with the dark background represent the required literals in an observation in order to be considered a covered observation.

Table 8. Positive and Negative Patterns - MILP Method

	Attributes									
	Age			Condition Indicator		Age*			Condition Indicator*	
	>= 1.5	>= 2.5	>= 3.5	>= 19	>= 21	< 1.5	< 2.5	< 3.5	< 19	< 21
+P1	0	0	0	1	1	0	0	0	0	0
+P2	0	0	1	0	0	0	0	0	0	0
-N1	0	0	0	0	0	0	0	1	0	1

After this phase of the algorithm, all the observations are covered by at least one pattern. The algorithm has to be redone as many times as needed, so that all the observations get covered by a pre-defined number of patterns, known as *Discriminating Power* [Mortada (2010)].

2.2. Hybrid Greedy Method

[Boros et al. (2000)] introduced two heuristic algorithms, called *Bottom-Up* and *Top-Down*, to obtain optimal *Prime* pure patterns. A pattern is prime if removal of any of its literals results in coverage of observations from the opposite class. The restriction on the generation of pure patterns can be relaxed by allowing the algorithm to cover observations from the opposite class. In this case, a pattern will be defined as a combination of literals covering at least a minimum number of observations of the pattern's class, and at most a maximum number of observations of the opposite class. The numbers are called *Coverage* and *Fuzziness* parameters, respectively.

Both algorithms aim to generate the shortest pattern while differing in the path that leads to this goal. The bottom-up algorithm starts with only one literal. Then it tries to add as many literals as required up to a point that the combination of literals forms a pattern.

The top-down algorithm starts with a combination of literals that certainly is a pattern. Then it tries to remove as many literals as possible from the pattern up to a point where the removal of the remaining literals will result in coverage of observations from the opposite class more than specified fuzziness parameter.

The hybrid greedy method is composed of two phases: the first and also the favored phase is the bottom-up phase. If any observation is left uncovered by the end of the first phase, the second phase, which is the top-down phase, is performed.

The bottom-up phase is a d step iterative procedure, where d is a pre-defined value showing the maximum degree of the patterns. At the n^{th} step, where $n \in \{1,2,\ldots, d\}$, any combination of n literals is examined to find out whether they form a pattern. When a pattern of a combination of n literals is generated, it is removed from the list that is used to generate the patterns with n+1 literals. This prevents the method from generating a pattern whose subset has already been detected as a pattern. Therefore, it guarantees the fewest possible numbers of literals in generated patterns.

Each step will be terminated if all the combinations are examined or all the observations are covered. The bottom-up phase will be terminated if it exceeds the d^{th} step or all the observations get covered. Figure 1 illustrates how the bottom-up phase proceeds.

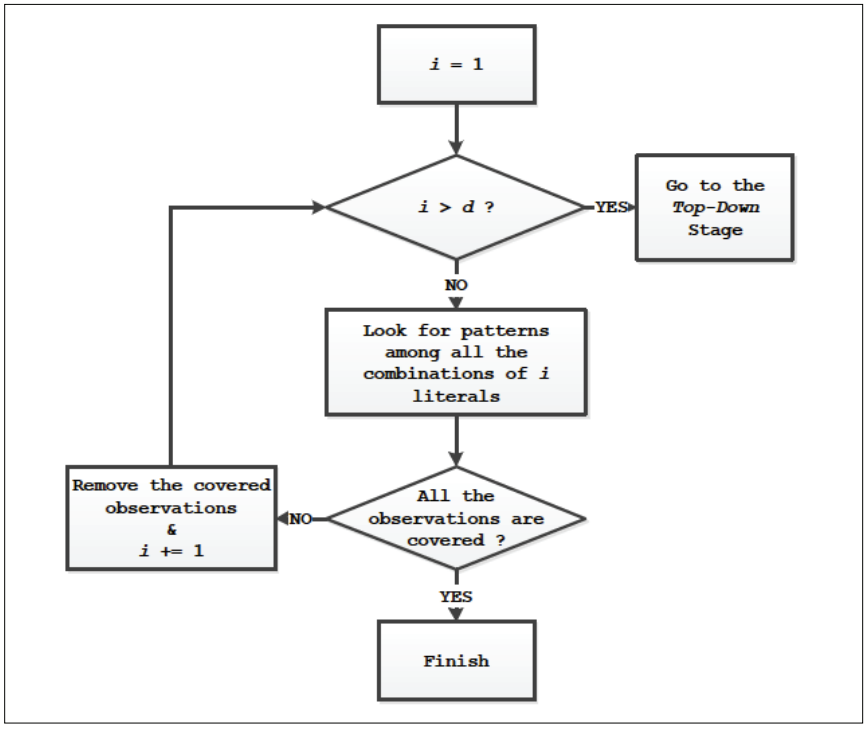


Figure 1. Bottom-Up Phase

If the bottom-up phase ends with any observation left uncovered, the method moves to the top-down phase. The top-down phase is a u step iterative procedure, where u is the number of uncovered observations at the previous phase. At the n^{th} step, where $n \in \{1,2,\ldots, d\}$, a pattern is formed from the literals that appear in the n^{th} uncovered observation. Then it tries to remove as many literals as possible from the pattern. Among all the candidate literals, the one whose removal leads to the maximum *Separability Power* of the pattern will be removed [Kronek et al. (2008)]. The authors defined separability power as the proportion of *Disagreement* between the pattern and observations of the opposite class to disagreement between the pattern and observations of the class. Disagreement between a pattern and an observation is defined as the number of literals in the pattern that do not appear in the observation. Associated with each pattern, the separability power is defined as following [Kronek et al. (2008)].

$$Separability\ Power = \frac{\sum_{i=1}^{m}\sum_{l=1}^{d} b_{i,l}}{\sum_{j=1}^{n}\sum_{l=1}^{d} b_{j,l}} \quad (10)$$

Where n and m respectively correspond to the number of observations of the pattern's class and that of the opposite class and d corresponds to the degree of the pattern. $b_{x,l}$ is defined such that $b_{x,l}=1$ if literal l appears in observation x, and $b_{x,l}=0$ otherwise.

Each step will be terminated if removal of the remaining literals results in destruction of the pattern, which means the number of covered observations from the opposite class exceeds the fuzziness parameter. The top-down phase will be terminated if it exceeds the u^{th} step or all the observations get covered. Figure 2 illustrates the top-down phase.

For instance, by assuming *degree* ≤ 3, *coverage* $\geq 50\%$ and *fuzziness* ≤ 1, we generate optimal positive prime patterns for the dataset in the Table 4. Starting with the bottom-up phase, at the first step, all the literals are examined to find out whether they can solely be considered as a pattern with degree one. The first literal to examine is the literal w_1, which appears in all the positive observations but in 3 out of 9 negative observations too. Since the fuzziness parameter is set to 1, this literal is rejected. The second literal to examine is the literal w_2, which appears in 2 out of 3 positive observations while 1 out of 9 negative observations. Since this literal passes both coverage and fuzziness conditions, it is accepted as a pattern. So, a pattern of degree one is generated which is composed of the literal w_2. Then, positive observations 1 and 3, which are covered by the pattern, are

removed from the set of positive observations. Similarly, all the other literals, $w_3,\ldots,w_{10}$, are examined. Another pattern of degree one is detected when the literal w_5 is examined. This literal appears in the remaining positive observation while in none of the negative observations. Similarly, the positive observation 2, which is covered by the pattern, is removed from the set of positive observations. Since all the positive observations are covered by these two patterns, the pattern generation method terminates successfully.

As another instance, by assuming *degree* ≤3, *coverage* ≥70% and *fuzziness* ≤1, the method is unable to find any pattern in either of the three steps of the bottom-up phase. Therefore, it moves to the top-down phase. In this phase, the positive observation 1, which is formed of the literals (w_1,w_2,w_3,w_4,w_{10}), is considered as a pattern. Then, the method tries to shorten the pattern by removing as many literals as possible from the pattern. By examining all the literals, the best candidate literal w_{10}, whose removal leads to the maximum separability power of 10, is detected and removed from the pattern. So, the pattern is shortened to the form of (w_1,w_2,w_3,w_4). Similarly, by examining all the remaining literals, the best candidate literal w_3, whose removal leads to the maximum separability power of 21, is detected and removed from the pattern, and the pattern is shortened to the form of (w_1,w_2,w_4). Similarly, all the remaining literals are examined, and the best candidate literals w_1 and w_4, whose removals lead to the maximum separability power of 15 and 8,

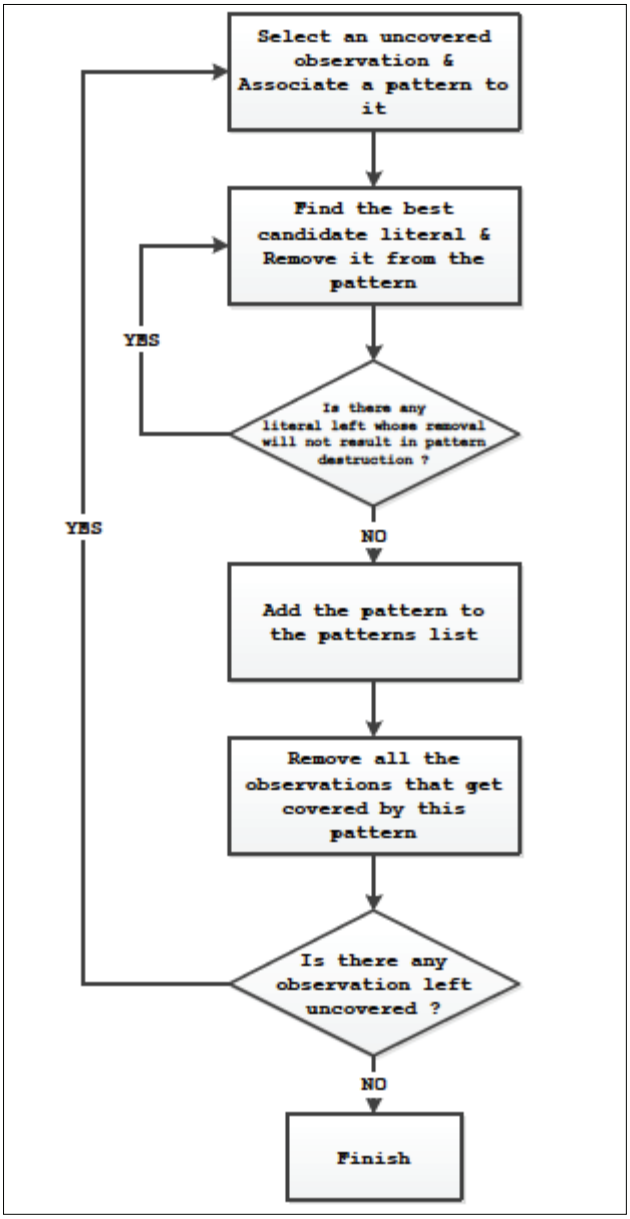


Figure 2 .Top-Down Phase

are removed one by one from the pattern. The final generated pattern is only composed of the literal w_2. It is worth mentioning that the separability power of each literal varies over different steps, based on the other literals that remain to be examined at each step. One might ask why this pattern was not generated while the method tried to find patterns of degree one at the bottom-up phase. The answer is that the method, at the bottom-up phase, takes coverage parameter into consideration, but not at the top-down phase. In the same way, in the previous example, positive observations 1 and 3, which are covered by the pattern, are removed from the set of positive observations. Since the only positive observation that is left uncovered is the positive observation 2, the same procedure is performed to shorten its corresponding pattern.

Table 9 shows all the positive and negative patterns generated with parameters *degree* ≤3, *coverage* ≥70% and *fuzziness* ≤1, for the binarized dataset provided in the Table 4. The cells with the dark background represent the literals that are required to be in a covered observation.

Table 9. Positive and Negative Patterns - Hybrid Greedy Method

	Attributes									
	Age			Condition Indicator		Age*			Condition Indicator*	
	>= 1.5	>= 2.5	>= 3.5	>= 19	>= 21	< 1.5	< 2.5	< 3.5	< 19	< 21
+P1	0	1	0	0	0	0	0	0	0	0
+P2	0	0	0	0	1	0	0	0	0	0
-N1	0	0	0	0	0	0	1	0	0	0
-N2	0	0	0	0	0	0	0	0	1	0

3. PATTERN'S QUALITY EVALUATION

Two most significant measures affecting performance of LAD are *Homogeneity* and *Prevalence* of the patterns [Boros et al. (2000)]. The homogeneity measure demonstrates how precisely a pattern can distinguish observations of the pattern's class from that of the opposite class. Associated with pattern p, homogeneity is defined as the proportion of number of correctly covered observations to all covered observations by the pattern [Boros et al. (2000)].

$$Homogeneity\ Ratio_P = \frac{\#(P \cap S^*)}{\#(P \cap S)} \tag{11}$$

Let X be defined as a set of observations. The expression $P \cap X$ symbolizes the set of observations in the set X that are covered by pattern P. Function #(O) counts the number

of observations in the set O. And S, S^*, and S^{*-} represent the set of all observations, observations of the pattern's class, and observations of the opposite class, respectively.

The prevalence measure demonstrates the pattern's ability to detect observations. Associated with pattern P, prevalence is defined as the proportion of number of covered observations by pattern P to all observations [Boros et al. (2000)].

$$Prevalence\ Ratio_P = \frac{\#(P \cap S)}{\#(S)} \tag{12}$$

The prevalence measure can be considered from two different viewpoints. The first point of view demonstrates the pattern's ability to detect observations of the pattern's class. *Proper Prevalence* is defined as the proportion of number of correctly covered observations by pattern P in a class to all observations of the class [Boros et al. (2000)].

$$Proper\ Prevalence\ Ratio_P = \frac{\#(P \cap S^*)}{\#(S^*)} \tag{13}$$

The second point of view demonstrates the pattern's weakness resulting in the wrong detection of observations of the opposite class. *Improper Prevalence* is defined as the proportion of number of incorrectly covered observations by pattern P from opposite class to all observations of the opposite class [Boros et al. (2000)].

$$Improper\ Prevalence\ Ratio_P = \frac{\#(P \cap S^{*-})}{\#(S^{*-})} \tag{14}$$

Table 10 and 11 show the quality measures of the patterns that were previously generated applying the MILP method. As previously mentioned, the MILP method guarantees the pattern is pure. So none of the generated patterns cover any observation from the opposite class. As a result, the homogeneity corresponding to all the patterns is equal to 1, which means that all the covered observations are from the pattern's class Similarly, the improper prevalence corresponding to all the patterns is equal to 0, which means that no observation from the opposite class is covered in the patterns.

Table 10. Quality Measures of Positive Patterns - MILP Method

Positive Patterns				
Patterns	Homogeneity	Prevalence	Proper Prevalence	Improper Prevalence
+P1	1	0.1666667	0.6666667	0
+P2	1	0.0833333	0.3333333	0

Table 11. Quality Measures of Negative Patterns - MILP Method

Negative Patterns				
Patterns	Homogeneity	Prevalence	Proper Prevalence	Improper Prevalence
-N1	1	0.75	1	0

Table 12 and 13 shows the quality measures of the patterns that were previously generated applying the hybrid greedy method with *degree* ≤3, *coverage* ≥70% and *fuzziness* ≤1. Since the hybrid greedy method allows the coverage of observations from the opposite class, the homogeneity of the patterns may take values lower than 1. Similarly, the improper prevalence of the patterns may take values greater than 0.

Table 12. Quality Measures of Positive Patterns - Hybrid Greedy Method

Positive Patterns				
Patterns	Homogeneity	Prevalence	Proper Prevalence	Improper Prevalence
+P1	0.6666667	0.25	0.6666667	0.1111111
+P2	1	0.1666667	0.6666667	0

Table 13. Quality Measures of Negative Patterns - Hybrid Greedy Method

Negative Patterns				
Patterns	Homogeneity	Prevalence	Proper Prevalence	Improper Prevalence
-N1	0.8888889	0.75	0.8888889	0.3333333
-N2	1	0.5833333	0.7777778	0

It is worth mentioning that the quality of negative patterns is generally better than that of positive patterns. The main reason is that the number of negative observations in the dataset is relatively higher, and as a result, discovering their characteristics is relatively more precise.

Interestingly, by comparing proper prevalence of negative patterns generated by different methods, it can be inferred that the MILP method generates patterns with relatively higher level of the correct coverage. Obviously, this is due to the fact that the MILP method is aimed at generation of strong patterns, which inherently possess the maximum coverage level.

4. MODEL FORMATION

As previously mentioned, LAD always takes two phases: at the *Train Phase*, it constructs a model based on a given historical dataset. Then, at the *Test Phase*, the model is used to diagnose or prognosticate a part of the historical dataset. So based on what is demanded at the test phase, the model may be constructed either from a *Diagnostic* or from a *Prognostic* point of view. In the following sections, we will discuss some methods using LAD to construct diagnostic models in the literature, and then will introduce methods to prognosticate failure in equipment.

4.1. Failure Diagnostic

A diagnostic model functions as a classifier to detect failures. A LAD pattern-based classification tool is built to categorize any new observation either as failed or survived observation. A sufficient number of patterns can provide a good indicator of characteristics of new observations in a way that is consistent with the historical observations. In the original definition of LAD [Crama et al. (1988)], a new observation is classified as positive (negative) only if it is covered by some of the positive (negative) patterns while not covered by any of the negative (positive) patterns. This way, none of the new observations, which are covered by some of the positive (negative) patterns as well as some of the negative (positive) patterns, can be classified. As a modification to the original definition, a *Discriminant Function* is built based on the relative significance of generated patterns (see *eq*. (3)) [Boros et al. (2000)].

The discriminant function is to classify a new observation covered by some positive and negative patterns. The value of $PP_1,\ldots,PP_r$ and $NP_1,\ldots,NP_s$ will be equal to 1 if the new observation is covered by the corresponding pattern, and 0, otherwise. A positive (negative) value of the discriminant function indicates that it is a positive (negative) observation. This way, any new observation can be classified unless the value of its discriminant function equals to zero.

$$class(obs) = \begin{cases} + & ;if \quad discriminant\ function\ > 0 \\ - & ;if \quad discriminant\ function\ < 0 \\ unclassified & ;if \quad discriminant\ function\ = 0 \end{cases} \tag{15}$$

Assuming equally normalized weights for both positive and negative patterns that were previously generated using the MILP method in the Table 8, the weights $H_1^+ = 0.5$, $H_2^+ = 0.5$, and $H_1^- = -1$ are respectively assigned to the patterns PP_1, PP_2, and NP_1. So the discriminant function is formulated as the following.

Discriminant Function = $0.5\ PP_1 + 0.5\ PP_2 - NP_1$

Table 14 shows a sample set of monitored data used for the diagnostic. It presents the monitored data of the equipment at four consecutive observation moments. The last observation moment of the equipment, referred to as the observation that will fail during the current period, is shown with the dark background.

Table 14. Sample Set of Monitored Data for Diagnostic or Prognostic

Observations		Attributes	
Equipment ID.	Observation Time	Age	Condition Indicator
1	0	0	14
1	1	1	16
1	2	2	20
1	3	3	22

Using binirization methods explained earlier, the dataset is transformed into the binary format, as shown in Table 15.

Table 15. Binary Transformation of Sample Set of Monitored Data for Diagnostic or Prognostic

Observations		Attributes				
Equipment ID.	Observation Time	Age			Condition Indicator	
		1.5	2.5	3.5	19	21
1	0	0	0	0	0	0
1	1	0	0	0	0	0
1	2	1	0	0	1	0
1	3	1	1	0	1	1

Each observation is tested with all the MILP patterns in the Table 8. The list of patterns covering each observation is shown in Table 16. The discriminant function is calculated based on the list of patterns covering each observation and the observations are classified based on the value of their discriminant function.

Table 16. Classification of Sample Set of Monitored Data based on MILP Patterns

Observations		Covering Patterns	Discriminant Function	Classification
Equipment ID.	Observation Time			
1	0	NP1	-1	-
1	1	NP1	-1	-
1	2	NP1	-1	-
1	3	PP1	0.5	+

As it is shown in the Table 16, the discriminant function correctly classified both positive and negative observations. If the discriminant function value is close to zero, there is a lack of certainty in the classification. On the other hand, if it is close to either 1 or -1, the classification is true with more certainty. Contrary to this example, a few patterns might not always classify all the observations accurately. The more generated patterns, the more characteristics of the classes are possessed by the pattern set, and consequently, the more accurate classification. In this example, only one negative pattern is generated which causes the discriminant function takes only the value -1 in the negative interval. If more negative patterns were generated, the discriminant function might take other values in the interval [-1,0). This way, the observations that are classified with less certainty would also be taken into consideration. If so, the gradual changes in the discriminant function,

from the negative (positive) values to the positive (negative) values, enable us to predict a change in the class of observations from a few observation moments before it occurs.

Similarly, applying generated patterns using the hybrid greedy in the Table 9, the discriminant function is formulated as following.

Discriminant Function = $0.5\ PP_1 + 0.5\ PP_2 - 0.5\ NP_1 - 0.5\ NP_2$

This calculation is based on $H_1^+ = 0.5$, $H_2^+ = 0.5$, $H_1^- = -0.5$ and $H_2^- = -0.5$ respectively for patterns PP_1, PP_2, NP_1, and NP_2. The classification of data, based on the hybrid greedy patterns in the Table 9, is shown in Table 17. In this example, two negative patterns are generated which enables the discriminant function to take different values in the interval [-1,0). As it is shown in the Table 17, both the negative patterns are reflected in the first two observations, which can be interpreted that the first two observations with absolute certainty are negative. But, the third observation only reflects one of the negative patterns, which means that the observation is classified with less certainty. This can be considered as an indication of an imminent failure. As it could be expected, the forth observation only reflects the positive patterns, and is classified as positive observation with absolute certainty. As it was previously mentioned, the gradual change in the discriminant function, from -1 to -0.5, enables us to predict an imminent failure before it occurs.

Table 17. Classification of Sample Set of Monitored Data based on Hybrid Greedy Patterns

Observations		Covering Patterns	Discriminant Function	Classification
Equipment ID.	Observation Time			
1	0	NP1 , NP2	-1	-
1	1	NP1 , NP2	-1	-
1	2	NP1	-0.5	-
1	3	PP1 , PP2	1	+

4.2. Failure Prognostic

Prognostic aims at the detection of the failure at certain moments in the future, which to the author's knowledge has been relatively untested. In the following section, we will introduce two methods to calculate the conditional survival probability of the equipment, based on the estimated survival functions using *Kaplan-Meier* (*KM*) estimation [Kaplan et al. (1958)].

For each positive or negative pattern generated using the hybrid greedy method, shown in the Table 9, the list of its covered observations is presented in Table 18.

Table 18. List of Observations from Train Set, Covered by Hybrid Greedy Patterns

+ Pattern	Covered Observations	- Pattern	Covered Observations
PP1	1-3 , 1-4 , 3-3	NP1	1-0 , 1-1 , 1-2 , 2-0 , 2-1 , 2-2 , 3-0 , 3-1 , 3-2
PP2	2-2 , 3-3	NP2	1-0 , 1-1 , 1-3 , 2-0 , 2-1 , 3-0 , 3-1

We associated to each pattern p, *Pattern Conditional Survival Probabilities* $SP_p(i)$ for $\forall i \in \{1,2,\ldots,T\}$, which represent the pattern's survival estimation of a piece of equipment for at least i periods, when the equipment's observation is covered by the pattern. T is the maximum available survival period within train set. Since LAD bases its pattern generation on the train set, its ability to prognosticate is limited to the attributes that it has observed in the train phase. In other words, T represents the LAD's maximum perception. KM estimation of pattern conditional survival probability is defined as the proportion of the number of observations covered by pattern P whose corresponding pieces of equipment survived at least i periods after being covered by the pattern, to the total number of observations covered by pattern P.

$$SP_P(i) = \frac{\#(P \cap S; \tau > \tau_0 + i\Delta)}{\#(P \cap S; \tau > \tau_0)} \quad (16)$$

Where S is the set of observations in the train set, and $P \cap S$ represents the subset of observations in the train set S that are covered by the pattern P. Function $\#(N)$ counts the number of members of the set N. τ is the actual failure time of the corresponding equipment, and τ_0 is the current age of the corresponding equipment at the observation moment when it is covered by pattern P. Δ is the observation period length.

Due to the fact that both age and condition of equipment are considered as the equipment's attributes in our study, the above-mentioned survival probability contains the prognostic information based on both age and condition of the equipment.

Table 19 shows KM estimation of conditional survival probability of the hybrid greedy patterns in the Table 9, based on their corresponding covered observations in the Table 18. For example, $SP_{PP1}(1)$ is equal to 0.333 because PP_1 covers observations 1-3 , 1-4 , 3-3 in the Table 18, but only observation 1-3 has corresponding equipment (i.e. equipment 1) that survives more than one period after being covered by PP_1. Both corresponding equipment of observation 1-4 and 3-3 have failed during next period as soon as they are covered by PP_1.

Table 19. KM Estimation of Conditional Survival Probability of Hybrid Greedy Patterns

$i\Delta$	1	2	3	4
PP1	0.333	0	0	0
PP2	0	0	0	0
NP1	0.889	0.667	0.333	0.111
NP2	1	0.714	0.428	0.143

We also defined the *Baseline Conditional Survival Probability* to indicate time-based survival function, regardless of the equipment's condition. This is taken into consideration for the probable case where no pattern covers an observation. This way, we will be able to calculate the conditional survival probability of the equipment at observation moments, which are not covered by any of the patterns. KM estimation of baseline conditional survival probability is calculated as the proportion of the number of pieces of equipment that survived at least i periods, to the number of all the pieces of equipment in train set.

$$SP_b(i) = \frac{\#(E;\tau>i\Delta)}{\#(E)} \tag{17}$$

Where E is the set of all pieces of equipment in the train set.

Table 20 shows KM estimation of baseline conditional survival probability based on all the observations in the train set. $SP_b(3)$ equal to 0.667 means that two out of three pieces of equipment in the train set have survived more than 3 periods.

Table 20. KM Estimation of Baseline Conditional Survival Probability

$i\Delta$	1	2	3	4
$SP_b(i)$	1	1	0.667	0.333

Considering the mentioned conditional survival probabilities, we introduce two methods to calculate the conditional survival probability of the equipment from which a new observation is collected.

The first method favors the Pattern Conditional Survival Probability (SP_p), while it takes into account the ones that were calculated for the equipment based on observations at previous observation moments (SP_{former}), less weightily. It also contains the Baseline Conditional Survival Probability (SP_b). Defining n as the number of patterns that cover an observation, the conditional survival probability of the equipment for i periods is calculated as follows:

$$SP_{obs}(i) = \begin{cases} \frac{\sum_{p=1}^{n} SP_p(i) + SP_b(i)}{n+1} & ; if \quad t = 0 \\ \frac{\sum_{p=1}^{n} SP_p(i) + SP_{former}(i+1)}{n+1} & ; if \quad t > 0 \end{cases} \tag{18}$$

Four consecutive observation records shown in Table 15 are covered by the hybrid greedy patterns as shown in the Table 17. Using the 1st method, as introduced in *eq*. (18), the conditional survival probabilities of the equipment at different observation moments are shown in Table 21. As previously mentioned, since all the train data failed before the fifth period, the fourth period is considered as the LAD's maximum perception, and as a result, the probability of survival more than four periods is equal to zero.

$SP_{obs}(2)$ for 1-0 is equal to 0.794 because the observation 1-0 is covered by patterns NP_1 and NP_2 for which $SP_{NP1}(2)$ and $SP_{NP2}(2)$ are equal to 0.667 and 0.714 respectively (see Table 19), and $SP_b(2)$ is equal to 1 (see Table 20). As a result $SP_{obs}(2)$ for 1-0 is equal to (0.667 + 0.714 + 1) / 3 = 0.794. $SP_{former}(1)$ for 1-1 is also equal to 0.794 because its corresponding equipment was formerly predicted to survive for at least 2 periods with the probability of 0.794 ($SP_{obs}(2)$ for 1-0 is 0.794). As mentioned earlier, since all the train data failed before the fifth period, the fourth period is considered as the LAD's maximum perception, and the probability of survival more than four periods is equal to zero.

Table 21. Conditional Survival Probabilities of Equipment at Different Observation Moments–1st Calculation Method

Obs	Covering Patterns	$\Sigma\ SP_p(t)$				$SP_b(t)$				$SP_{former}(t)$				$SP_{obs}(t)$			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1-0	NP1,NP2	1.89	1.38	0.76	0.25	1	1	0.67	0.33	-	-	-	-	0.96	0.79	0.48	0.19
1-1	NP1,NP2	1.89	1.38	0.76	0.25	-	-	-	-	0.79	0.48	0.19	0	0.89	0.62	0.32	0.08
1-2	NP1	0.89	0.67	0.33	0.11	-	-	-	-	0.62	0.32	0.08	0	0.76	0.5	0.21	0.06
1-3	PP1,PP2	0.33	0	0	0	-	-	-	-	0.5	0.21	0.06	0	0.28	0.07	0.02	0

The second method also prefers the latest observation to older observation. But, it considers equal weight for Pattern and Baseline Conditional Survival Probabilities. The conditional survival probability of the equipment at current observation moment is calculated as follows:

$$SP_{obs}(i) = \begin{cases} \dfrac{\frac{\sum_{p=1}^{n} SP_p(i)}{n} + SP_b(i)}{2} & ;if \quad t = 0 \\ \dfrac{\frac{\sum_{p=1}^{n} SP_p(i) + SP_{former}(i+1)}{n+1} + SP_b(i)}{2} & ;if \quad t > 0 \end{cases} \tag{19}$$

Using the 2nd method, as introduced in *eq*. (19), the conditional survival probabilities of the equipment at different observation moments are shown in Table 22. $SP_b(1)$ for 1-3 is equal to 0.5 because one out of two pieces of equipment that have survived more than 3 periods, has survived more than 4 periods.

Table 22. Conditional Survival Probabilities of Equipment at Different Observation Moments–2nd Calculation Method

Obs	Covering Patterns	$\Sigma\ SP_p(t)$				$SP_b(t)$				$SP_{former}(t)$				$SP_{obs}(t)$			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1-0	NP1,NP2	1.89	1.38	0.76	0.25	1	1	0.67	0.33	-	-	-	-	0.97	0.85	0.53	0.23
1-1	NP1,NP2	1.89	1.38	0.76	0.25	1	0.67	0.33	0	0.85	0.53	0.23	0	0.96	0.65	0.33	0.04
1-2	NP1	0.89	0.67	0.33	0.11	0.67	0.33	0	0	0.65	0.33	0.04	0	0.72	0.42	0.09	0.03
1-3	PP1,PP2	0.33	0	0	0	0.5	0	0	0	0.42	0.09	0.03	0	0.38	0.02	0	0

Figure 3 shows the conditional survival probability of the last observation using the patterns generated by both the hybrid greedy and MILP methods, based on both the survival probability calculation methods (*eq*. (18) and *eq*. (19)). As it can be inferred from the Figure 3, those provided by the first method gradually decrease over time while those provided by the second method severely decrease from the time 1 to 2, which emphasizes the probability of failure in the next coming period.

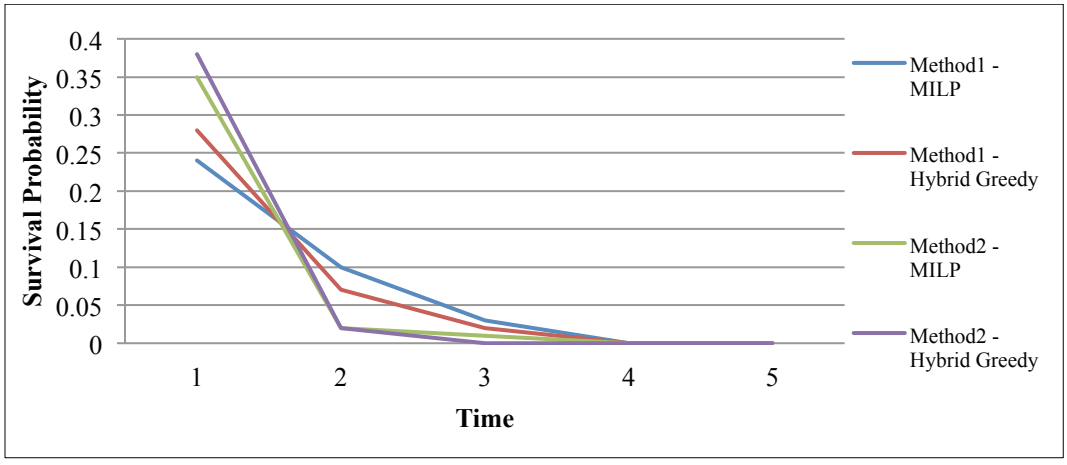


Figure 3. Conditional Survival Probability of the Last Observation using Different Methods

CHAPTER 3 **: EXPERIMENTS**

We applied the LAD methodology on *Prognostics and Health Management Challenge* dataset, a condition monitoring dataset provided by *NASA Ames Prognostics Data Repository*. The dataset consists of approximately 46,000 observations associated with 218 pieces of mechanical equipment. For each observation, 3 operational settings and 21 measurements associated with the equipment's attributes are provided.

1. DATA PREPARATION

In order to model the system in a reasonable time, we had to decrease the dataset size. To do so, we extracted every 10th observation and reduced the number of observations to about 4,600.

Table 23 shows the correlation between attributes. Cells with the dark background show values greater than 0.9. As it can be inferred, most of the attributes are highly correlated. Involving correlated attributes in the model is not appropriate due to two main reasons: First and foremost, correlated attributes do not provide any additional information. Second, the more number of attributes involved, the more modeling time is required. Our tests show that the modeling time grows exponentially by increasing the number of attributes.

Table 23. Correlation Matrix

Att.	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
F	1.00	0.94	0.87	0.90	0.99	0.99	0.97	0.57	0.86	0.83	0.70	0.97	0.16	0.34	-.54	0.80	0.87	0.57	0.16	0.98	0.98
G	0.94	1.00	0.98	0.98	0.92	0.94	0.97	0.81	0.98	0.91	0.89	0.97	0.47	0.62	-.78	0.81	0.98	0.81	0.47	0.96	0.96
H	0.87	0.98	1.00	0.99	0.84	0.88	0.93	0.89	1.00	0.93	0.96	0.93	0.62	0.75	-.87	0.81	1.00	0.89	0.62	0.92	0.92
I	0.90	0.98	0.99	1.00	0.88	0.92	0.96	0.84	0.99	0.96	0.94	0.96	0.54	0.71	-.85	0.86	0.99	0.84	0.54	0.95	0.95
J	0.99	0.92	0.84	0.88	1.00	1.00	0.98	0.52	0.83	0.84	0.67	0.98	0.11	0.32	-.52	0.83	0.84	0.52	0.11	0.99	0.99
K	0.99	0.94	0.88	0.92	1.00	1.00	0.99	0.59	0.87	0.88	0.73	0.99	0.19	0.40	-.59	0.84	0.89	0.59	0.19	1.00	1.00
L	0.97	0.97	0.93	0.96	0.98	0.99	1.00	0.68	0.92	0.92	0.80	1.00	0.30	0.50	-.68	0.86	0.93	0.68	0.30	1.00	1.00
M	0.57	0.81	0.89	0.84	0.52	0.59	0.68	1.00	0.90	0.78	0.97	0.68	0.90	0.93	-.97	0.60	0.89	1.00	0.90	0.65	0.65
N	0.86	0.98	1.00	0.99	0.83	0.87	0.92	0.90	1.00	0.93	0.96	0.92	0.63	0.77	-.88	0.80	1.00	0.90	0.63	0.91	0.91
O	0.83	0.91	0.93	0.96	0.84	0.88	0.92	0.78	0.93	1.00	0.89	0.92	0.51	0.72	-.85	0.91	0.93	0.78	0.50	0.91	0.91
P	0.70	0.89	0.96	0.94	0.67	0.73	0.80	0.97	0.96	0.89	1.00	0.80	0.80	0.89	-.97	0.73	0.96	0.97	0.80	0.78	0.78
Q	0.97	0.97	0.93	0.96	0.98	0.99	1.00	0.68	0.92	0.92	0.80	1.00	0.30	0.50	-.68	0.86	0.93	0.68	0.30	1.00	1.00
R	0.16	0.47	0.62	0.54	0.11	0.19	0.30	0.90	0.63	0.51	0.80	0.30	1.00	0.93	-.88	0.29	0.61	0.90	1.00	0.27	0.27
S	0.34	0.62	0.75	0.71	0.32	0.40	0.50	0.93	0.77	0.72	0.89	0.50	0.93	1.00	-.96	0.53	0.75	0.93	0.93	0.47	0.47
T	-.54	-.78	-.87	-.85	-.52	-.59	-.68	-.97	-.88	-.85	-.97	-.68	-.88	-.96	1.00	-.66	-.87	-.97	-.88	-.66	-.66
U	0.80	0.81	0.81	0.86	0.83	0.84	0.86	0.60	0.80	0.91	0.73	0.86	0.29	0.53	-.66	1.00	0.81	0.60	0.29	0.86	0.86
V	0.87	0.98	1.00	0.99	0.84	0.89	0.93	0.89	1.00	0.93	0.96	0.93	0.61	0.75	-.87	0.81	1.00	0.89	0.61	0.92	0.92
W	0.57	0.81	0.89	0.84	0.52	0.59	0.68	1.00	0.90	0.78	0.97	0.68	0.90	0.93	-.97	0.60	0.89	1.00	0.90	0.65	0.65
X	0.16	0.47	0.62	0.54	0.11	0.19	0.30	0.90	0.63	0.50	0.80	0.30	1.00	0.93	-.88	0.29	0.61	0.90	1.00	0.27	0.27
Y	0.98	0.96	0.92	0.95	0.99	1.00	1.00	0.65	0.91	0.91	0.78	1.00	0.27	0.47	-.66	0.86	0.92	0.65	0.27	1.00	1.00
Z	0.98	0.96	0.92	0.95	0.99	1.00	1.00	0.65	0.91	0.91	0.78	1.00	0.27	0.47	-.66	0.86	0.92	0.65	0.27	1.00	1.00

In order to remove the effect of involving trivial attributes, we applied the *Principal Component Analysis* (*PCA*). PCA [Pearson (1901)] is a mathematical method that

converts the attributes into a set of linearly uncorrelated attributes, using *Orthogonal Transformation*. The orthogonal transformation [Pearson (1901)] is performed in a way that the first principal component has the maximum variability in the available data, and the following principal components have decreasingly the next maximum variability while are uncorrelated with previous ones. One way to perform the orthogonal transformation is called *Eigen Decomposition* [Pearson (1901)]. In this procedure, the correlation matrix is factorized into a matrix represented in terms of the *Eigenvalues* and *Eigenvectors* [Pearson (1901)]. An eigenvector of a matrix is the vector that remains parallel if multiplied by the matrix. Corresponding to an eigenvector, an eigenvalue is the factor that scales the eigenvector during the multiplication. The more value of an eigenvalue, the more informative its corresponding eigenvector, and consequently, the better the eigenvector represents the variety in the dataset. As a result, a multi-attribute dataset can be reduced in the number of attributes by using only the most informative attributes (eigenvectors).

Eigenvalue, variability and cumulative variability percentage, associated with each newly defined attribute, are shown in Table 24.

Table 24. Eigenvalue, Variability, and Cumulative Variability

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11
Eigenvalue	16.86	3.62	0.37	0.09	0.04	0.02	0.01	0.00	0.00	0.00	0.00
Variability (%)	80.28	17.22	1.76	0.40	0.17	0.10	0.04	0.01	0.01	0.01	0.00
Cumulative (%)	80.28	97.50	99.26	99.66	99.83	99.93	99.97	99.98	99.99	99.99	100

As Table 24 shows, the first principal component has the maximum eigenvalue and variability percentage, 16.86 and 80.28% respectively. In the next level, the second component has the maximum eigenvalue and variability percentage, 3.62 and 17.22% respectively. The first two principal components convey more than 97% of characteristics of all the attributes before conversion. So, from this point on, we will construct the model based on these two attributes as a substitute for system's 21 attributes.

From the dataset of 218 pieces of equipment, a dataset of 15 pieces of equipment was extracted to test the performance of the model, and from the remaining pieces of equipment, 10 datasets of 70 randomly extracted pieces of equipment were generated to train the model. We constructed 10 models based on different train datasets, and for each model, we calculated the conditional survival probabilities based on the same test dataset.

This enables us to compare the conditional survival probability of each matched pair observation obtained by different models while eliminating the difference (error) due to randomness of different random test data. Finally, the prognostic results provided by all the 10 models are averaged over the models. The final analysis of the performances is performed using the averages.

2. SAMPLE PROGNOSTIC RESULTS

The proposed LAD prognostic model is entirely coded in the Python programming language, and all of the steps are carried out automatically. The inputs and outputs interface through the Excel spreadsheets. The MILP model is solved, using the CPLEX optimization software package module for Python.

Table 25 shows a set of monitored data for a test piece of equipment at 21 consecutive observation moments. Each row corresponds to an observation moment, for which the age and two condition indicators, all considered as the observation's attributes, are shown in the columns 3 to 5. The last observation moment of the equipment, referred to as the observation that will fail during the current period, is shown with the dark background. This is in fact the last period that record of data has been received for the equipment. Before getting to the next observation moment, the equipment has failed.

Table 25. Set of Monitored Data for a Test Equipment

Observations		Attributes		
Equipment ID.	Observation Time	Age	Condition Indicator 1	Condition Indicator 2
1	0	0	0.059	0.011
1	1	1	0.044	0.078
1	2	2	0.004	0.000
1	3	3	0.009	0.023
1	4	4	0.005	0.012
1	5	5	0.059	0.011
1	6	6	0.008	0.024
1	7	7	0.004	0.000
1	8	8	0.016	0.000
1	9	9	0.016	0.000
1	10	10	0.009	0.024
1	11	11	0.043	0.078
1	12	12	0.008	0.024
1	13	13	0.004	0.000
1	14	14	0.060	0.011
1	15	15	0.043	0.076
1	16	16	0.004	0.013
1	17	17	0.004	0.013
1	18	18	0.007	0.000
1	19	19	0.007	0.027
1	20	20	0.063	0.008

Table 26 shows the prognostic results for the data in Table 25, based on the second conditional survival probability calculation method introduced in the chapter 2, section 4.2. Each row corresponds to an observation moment, for which the conditional probabilities of survival up to 1 to 5 periods later, are respectively shown in the columns 3 to 7. For instance, the 16th row shows the conditional probabilities of survival up to the period 17th to 21st, based on the equipment's attributes at the 16th observation moment. The conditional probabilities of survival up to the period 17th to 21st are respectively equal to 0.923, 0.878, 0.799, 0.732, and 0.665.

Table 26. Prognostic Results for the Test Equipment

Observations		Conditional Survival Probability					Residual Life		
Equip. ID.	Obs. Time	X > 1	X > 2	X > 3	X > 4	X > 5	MRL	Actual RL	Diff.
1	0	0.998	0.990	0.978	0.961	0.939	14.80	20	- 5.20
1	1	0.996	0.986	0.973	0.953	0.933	13.87	19	- 5.13
1	2	0.995	0.984	0.965	0.948	0.917	13.46	18	- 4.54
1	3	0.996	0.983	0.970	0.948	0.924	13.43	17	- 3.57
1	4	0.995	0.986	0.972	0.955	0.931	13.56	16	- 2.44
1	5	0.996	0.986	0.971	0.951	0.924	13.49	15	- 1.51
1	6	0.994	0.978	0.961	0.929	0.902	12.96	14	- 1.04
1	7	0.994	0.982	0.960	0.942	0.909	13.27	13	0.27
1	8	0.995	0.983	0.969	0.950	0.924	13.41	12	1.41
1	9	0.995	0.984	0.970	0.953	0.926	13.43	11	2.43
1	10	0.995	0.982	0.969	0.946	0.923	13.33	10	3.33
1	11	0.995	0.983	0.968	0.947	0.924	13.45	9	4.45
1	12	0.993	0.976	0.958	0.925	0.897	12.81	8	4.81
1	13	0.991	0.975	0.946	0.923	0.884	12.65	7	5.65
1	14	0.990	0.967	0.946	0.913	0.870	12.14	6	6.14
1	15	0.981	0.950	0.913	0.862	0.815	10.91	5	5.91
1	16	0.923	0.878	0.799	0.732	0.665	8.22	4	4.22
1	17	0.962	0.890	0.793	0.702	0.609	7.37	3	4.37
1	18	0.859	0.740	0.647	0.569	0.472	6.24	2	4.24
1	19	0.823	0.656	0.540	0.429	0.367	4.82	1	3.82
1	20	0.878	0.667	0.524	0.403	0.301	4.17	0	4.17

For each observation moment, we calculate the set of conditional probabilities of survival for the future predictable periods (columns 3 to 7). However, this set is not meaningfully comparable to its matched pair set provided by other experiments. Therefore, we transform the information of the set into a single comparable value, *Mean Residual Life* (*MRL*), so that we can compare performance of different experiments. MRL represents the expected value of equipment residual life, and is formulated as following [Banjevic et al. (2006)]:

$$MRL = \sum_{i=1}^{\infty} i\Delta \times Probability(\tau > \tau_0 + i\Delta | \tau > \tau_0) \quad (20)$$

Where *Probability* ($\tau > \tau_0 + i\Delta | \tau > \tau_0$) shows the probability of survival for at least i periods, knowing that the equipment has not failed until τ_0. This conditional probability is identical with conditional survival probability $SP_{obs}(i)$, introduced in this work. So the MRL is formulated in terms of $SP_{obs}(i)$ as following:

$$MRL = \sum_{i=1}^{\infty} i\Delta \times SP_{obs}(i) \quad (21)$$

In Table 26, associated with each observation moment, the MRL and the actual *Residual Life* (*RL*) are calculated and shown in columns 8 and 9 respectively. The actual RL is not determined until the equipment failure moment. At this moment, the equipment lifetime is determined as the time difference between the failure moment and installation moment. In the above example, the equipment lifetime is equal to 20. Then, for each observation moment, the actual RL is calculated by subtracting the observation moment from the equipment lifetime. For instance, the actual RL associated with the 16th observation moment is equal to 4. At each observation moment, the difference between the MRL and the actual RL is calculated and shown in column 10. The lower the difference, the better the performance of the model. Figure 4 shows a comparison between the MRL and the actual RL of the equipment.

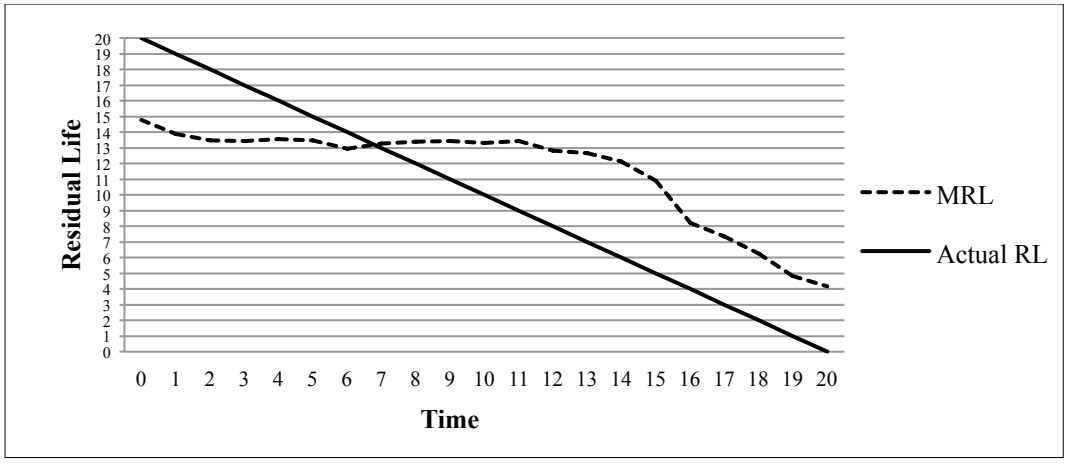


Figure 4. A Comparison Between MRL and Actual RL

Figure 4 shows that in early observation moments, the model underestimates the MRL. As time passes by, the MRL gets closer to the actual RL, and the model correctly estimates the MRL almost at the mid-age observation moment. Later, when getting closer to the actual failure moment, the model overestimates the MRL.

Conditional survival probabilities for the next 5 periods at each observation moment in the Table 26, are illustrated in Figure 5. The last five observations reveal severe decreases in the survival probability, which is consistent with the imminent failure in the next few periods.

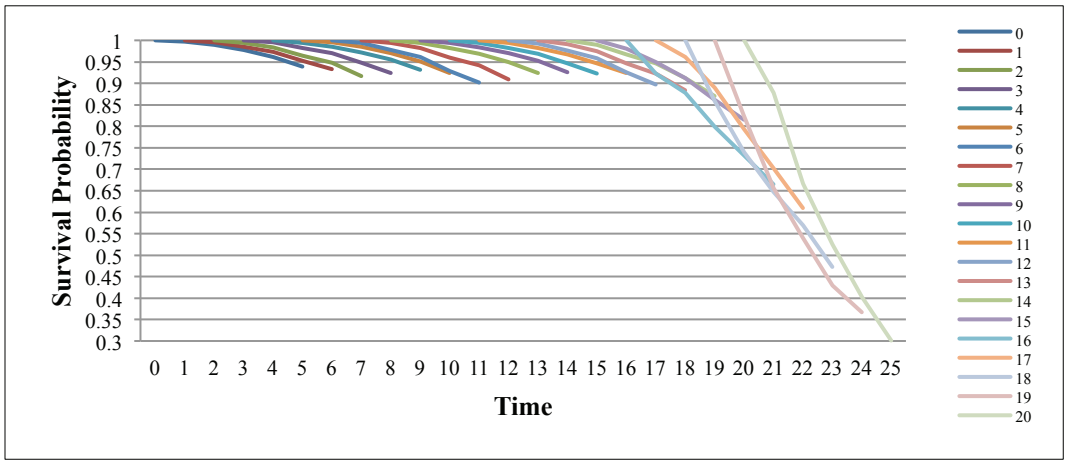


Figure 5. Survival Function for the Next 5 Periods at Each Observation Moment

3. DESIGN OF EXPERIMENT (DOE)

Our prognostics model formation generally has two phases: At the first phase, data binarization, using either the sensitive discriminating or the equipartitioning method, is performed. The performance of the equipartitioning method depends on a pre-defined number of cut-points. At the second phase, survival analysis is performed based on the patterns generated by either the MILP or the hybrid greedy method. The performance of the hybrid greedy method depends on the pre-defined degree, coverage and fuzziness parameters. Table 27 shows our *Design of Experiments* (*DOE*). We will examine 5 parameter settings for the data binarization phase, and 13 parameter settings for the survival analysis phase. We have also included the Weibull PHM [Banjevic et al. (2006)] to calculate conditional survival probabilities to compare different LAD settings performances with that of PHM. As a result, (5×14=) 70 experiments are designed in order to compare the performance of the model based on different parameters and methods. Each experiment is performed 10 times based on different train sets and the results are averaged over all the 10 runs.

Table 27. Design Of Experiments (DOE)

Data Binarization	Parameters
Sensitive Discriminating	-
Equipartitioning	# cut-points = 20
	# cut-points = 30
	# cut-points = 40
	# cut-points = 50

Survival Analysis	Parameters
PHM	-
MILP	-
Hybrid Greedy	d = 3 coverage > 10% fuzziness <=0
	d = 3 coverage > 10% fuzziness <=1
	d = 3 coverage > 10% fuzziness <=2
	d = 3 coverage > 20% fuzziness <=0
	d = 3 coverage > 20% fuzziness <=1
	d = 3 coverage > 20% fuzziness <=2
	d = 3 coverage > 30% fuzziness <=0
	d = 3 coverage > 30% fuzziness <=1
	d = 3 coverage > 30% fuzziness <=2
	d = 3 coverage > 40% fuzziness <=0
	d = 3 coverage > 40% fuzziness <=1
	d = 3 coverage > 40% fuzziness <=2

4. COMPARISONS

The absolute value of differences between the MRL and the actual RL indicates the accuracy of the experiment. The lower the difference, the more accurate the experiment. So, the measurement under study in the DOE is the absolute value of differences between the MRL and the actual RL.

Let $X=\{1,2,\ldots,U\}$ be defined as the test set, where U is the number of observations in the test set. In order to compare the performance of different experiments, first we associate the set $Z^e = (z_1^e, z_2^e,\ldots, z_U^e)$ with experiment e. The n^{th} member of the set, z_n^e is formulated as $|MRL_n^e - RL_n|$, where MRL_n^e is the estimated MRL by experiment e for observation n, and RL_n is the actual RL for observation n. Then, experiments $1,2,\ldots,m$ are compared based on the sets $Z^1, Z^2, \ldots, Z^m$. To do so, members of the sets are compared pair by pair, using the *Friedman Matched-pair Test* (*Dunn's Multiple Comparison Test*) [Friedman (1940)].

The comparison is structured as follows: First, we compare two conditional survival probability calculation methods introduced in the chapter 2, section 4.2 (*eq*. (18) and *eq*. (19)). Second, different hybrid greedy methods are compared. Third, the best hybrid greedy method is compared with the MILP and the PHM methods.

4.1. Method # 1 vs. Method # 2

Comparison of the two methods of conditional survival probability calculation reveals that the second method that equally prefers the baseline and pattern survival probability

(*eq*. (19)), statistically outperforms the first method that prefers the pattern survival probability (*eq*. (18)).

Table 28 shows the comparison between the two methods for a sample experiment. The mean of the differences in the second method (3.811) is significantly lower than that in the first method (4.534). Both lower and upper 95% confidence intervals of the second method (3.542,4.080) are extremely lower than that of the first one (4.205,4.864). The median of the second method (3.425) is also lower than that of the first one (3.844).

Table 28. Method #1 vs. Method #2 (sample experiment)

Minimum	0.4398	0.3149
25% Percentile	2.136	1.763
Median	3.844	3.425
75% Percentile	6.303	5.329
Maximum	12.36	10.20
Mean	4.534	3.811
Std. Deviation	3.008	2.458
Std. Error	0.1676	0.1370
Lower 95% CI	4.205	3.542
Upper 95% CI	4.864	4.080

4.2. Hybrid Greedy # 1 vs. … vs. Hybrid Greedy # 12

Comparison of the hybrid greedy methods reveals that the one with the parameters *coverage* ≥ 10% and *fuzziness* = 0 outperforms the other methods. Table 29 shows the comparison between the mean values of different hybrid greedy methods. The method with the parameters *coverage* ≥ 10% and *fuzziness* = 0 provides the lowest mean value although the differences are not statistically significant.

Table 29. Hybrid Greedy #1 vs. … vs. Hybrid Greedy #12

Par.	C>0.1 F <=0	C>0.1 F <=1	C>0.1 F <=2	C>0.2 F <=0	C>0.2 F <=1	C>0.2 F <=2	C>0.3 F <=0	C>0.3 F <=1	C>0.3 F <=2	C>0.4 F <=0	C>0.4 F <=1	C>0.4 F <=2
S.D.	3.751^{+}	3.774	3.78	3.783	3.789	3.758	3.788	3.775	3.783	3.796	3.775	3.787
E.20	3.748^{+}	3.762	3.826	3.778	3.816	3.835	3.812	3.839	3.822	3.833	3.842	3.857
E.30	3.731^{+}	3.753	3.754	3.77	3.792	3.835	3.778	3.778	3.829	3.786	3.795	3.785
E.40	3.723^{+}	3.727	3.751	3.738	3.749	3.753	3.758	3.731	3.732	3.779	3.738	3.775
E.50	3.728^{+}	3.739	3.788	3.75	3.734	3.818	3.751	3.766	3.754	3.763	3.772	3.753

$^{+}$ *Minimum mean value*

4.3. Hybrid Greedy vs. MILP vs. PHM

Comparison of the hybrid greedy, the MILP, and the PHM reveals that the PHM statistically outperforms the LAD methods. While the hybrid greedy and the MILP

methods are not statistically different. Table 30 shows the comparison between the mean values of the three methods.

Table 30. Hybrid Greedy vs. MILP vs. PHM

	Hybrid Greedy	MILP	PHM
Sensitive Discriminating	3.751	3.811	3.507[+]
Equipartitioning 20	3.748	3.867	3.51[+]
Equipartitioning 30	3.731	3.826	3.509[+]
Equipartitioning 40	3.723	3.801	3.51[+]
Equipartitioning 50	3.728	3.801	3.51[+]

[+] *Minimum mean value*

For the data in the Table 25, Figures 6-8 illustrate the survival functions for the next 5 periods at 21 consecutive observation moments, using the best models respectively provided by the hybrid greedy, the MILP, and the PHM methods.

As it can be inferred from the Figure 6, survival probabilities for the next 5 periods do not move below 0.95 for the early observation moments. They also do not move below 0.9 except for the last few observation moments. But as soon as it is warned about an imminent failure, the survival probabilities severely decrease even below 0.4.

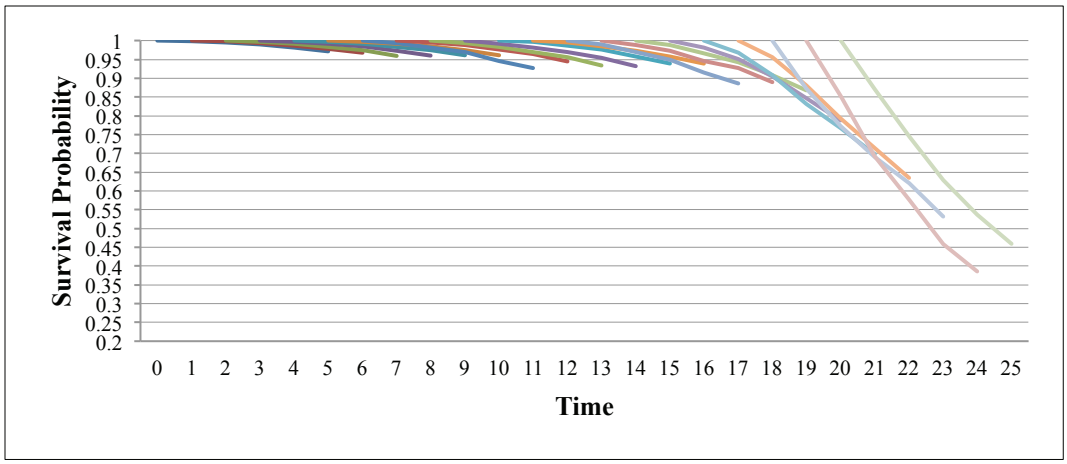


Figure 6. Survival Function using Hybrid Greedy Method

In comparison with the hybrid greedy method, Figure 7 shows that the MILP method has more pessimistic outlook about the future. The slopes of the survival functions in the Figure 7 are higher in comparison with that in the Figure 6. Survival probabilities for the next 5 periods usually move below 0.9 even at the early observation moments. But same as the hybrid greedy, the MILP survival probabilities also severely decrease as soon as it is warned about an imminent failure. Comparing the survival probabilities at the

observation moments 19 and 20 shows that the LAD methods are pessimistic about the future of the 19th observation while they are optimistic about the future of the 20th observation, which cannot be seen in the PHM method (Figure 8). The reason might be that a warning about a hidden failure was received at the 19th observation while the failure was not detectable at the 20th observation.

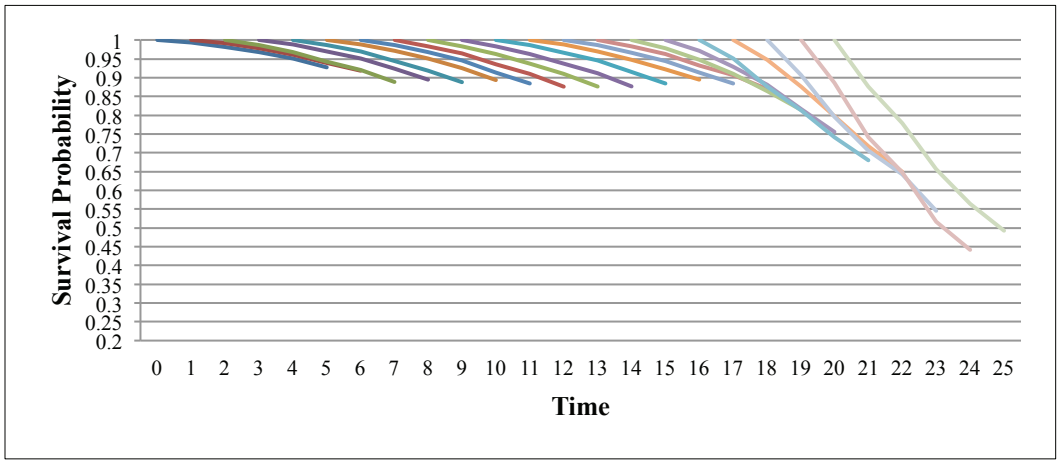


Figure 7. Survival Function using MILP Method

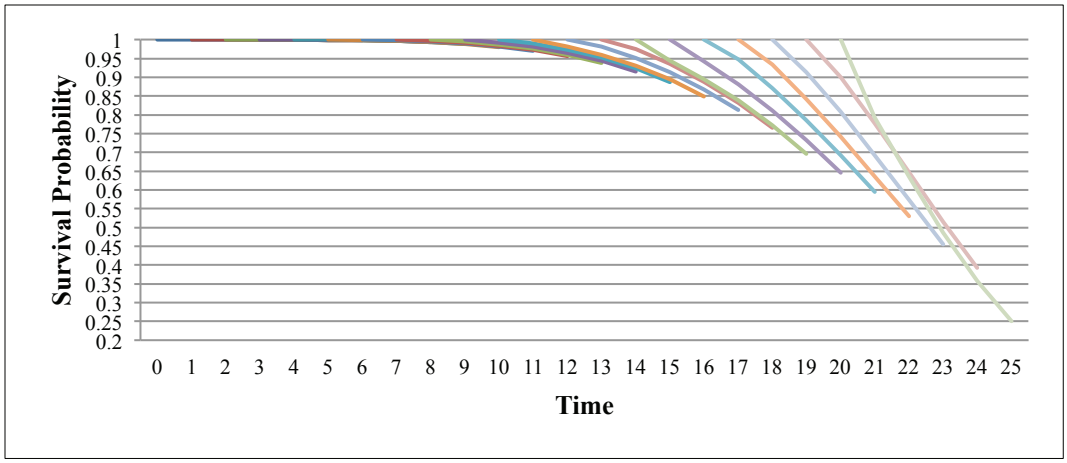


Figure 8. Survival Function using PHM Method

Figure 8 shows that the slopes of the survival functions gradually increase as the equipment health deteriorates. Survival probabilities at the last few observation moments severely decrease even as low as 0.25.

Figure 9 shows the difference between the MRL and the actual RL of the equipment in the Table 25, using the best models provided by the hybrid greedy, the MILP, and the PHM methods. It shows that the LAD methods underestimate the MRL at the early observation moments (pessimistic outlook about the equipment future). As time passes

by, the estimations get closer to the actual RL, and they correctly estimate the MRL between 4th and 7th observations. Later, when getting closer to the actual failure moment, they overestimate the MRL (optimistic outlook about the equipment future). It can be concluded that the LAD methods have neither a constant optimistic outlook nor a constant pessimistic outlook about the equipment future, whereas they adjust their outlook over the equipment lifetime. Contrary to the LAD methods, the PHM method always overestimates the MRL by at least one period (optimistic outlook about the equipment future). Interestingly, all the curves almost meet at the 18th and 19th observations. This means that all the methods estimate almost the same MRL at the last few observation moments before the actual failure moment. Figure 9 also shows that the PHM method is stable at the early observation moments (see zone A), while the LAD methods are stable when getting closer to the actual failure moment (see zone B).

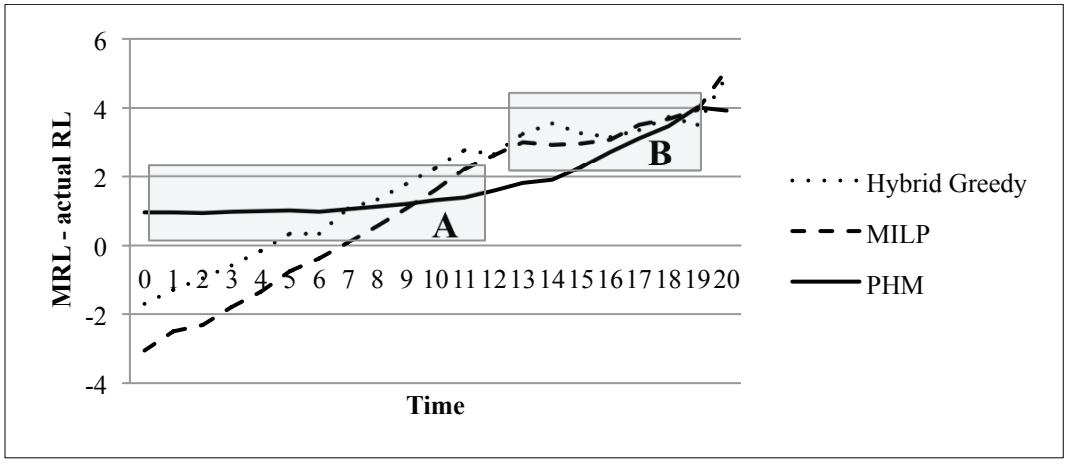
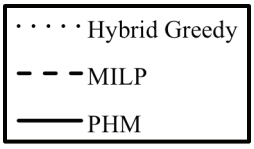


Figure 9. Difference Between MRL and Actual RL using Hybrid Greedy, MILP, and PHM Methods

Figures 10-a to 10-u illustrate the survival functions for the data in the Table 25, using the best models provided by the hybrid greedy, the MILP, and the PHM methods at the observation moments 0 to 20 respectively.



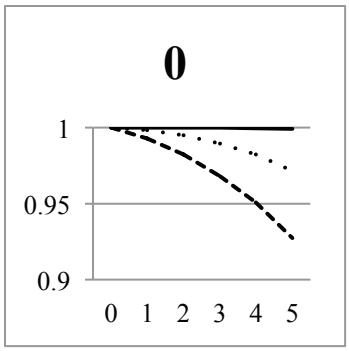


Figure 10-a. Observation Moment 0

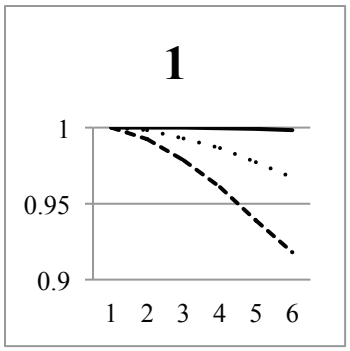


Figure 10-b. Observation Moment 1

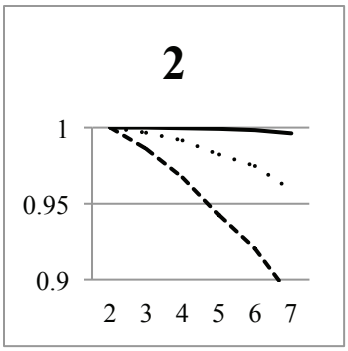


Figure 10-c. Observation Moment 2

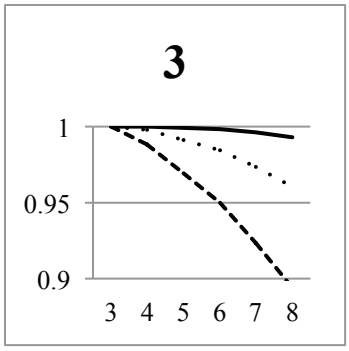


Figure 10-d. Observation Moment 3

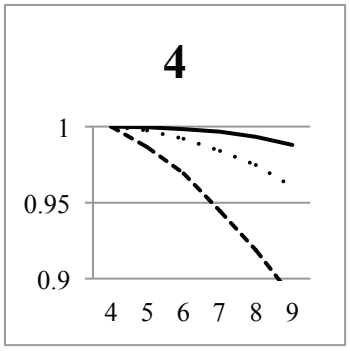


Figure 10-e. Observation Moment 4

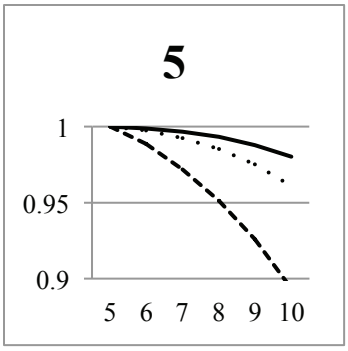


Figure 10-f. Observation Moment 5

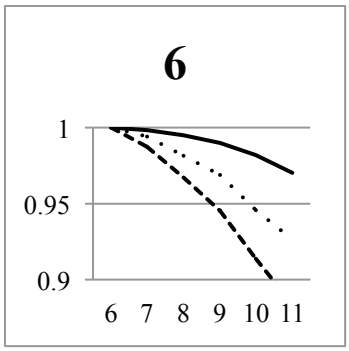


Figure 10-g. Observation Moment 6

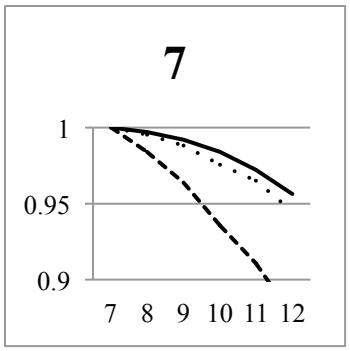


Figure 10-h. Observation Moment 7

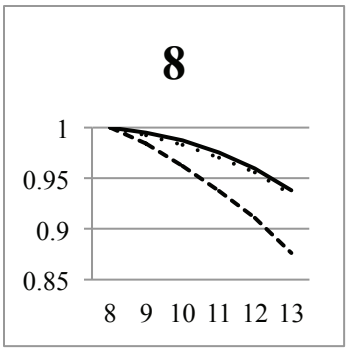


Figure 10-i. Observation Moment 8

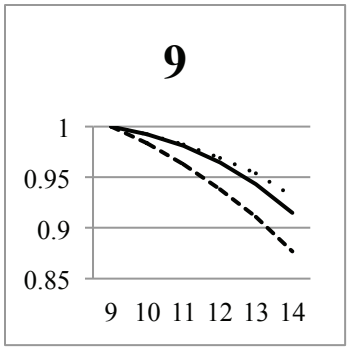


Figure 10-j. Observation Moment 9

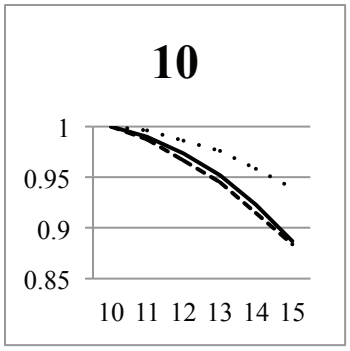


Figure 10-k. Observation Moment 10

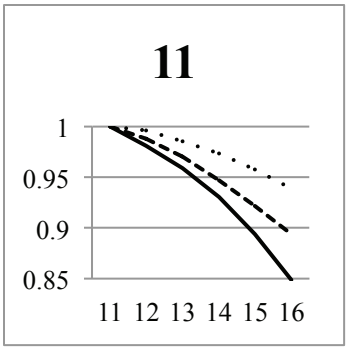


Figure 10-l. Observation Moment 11

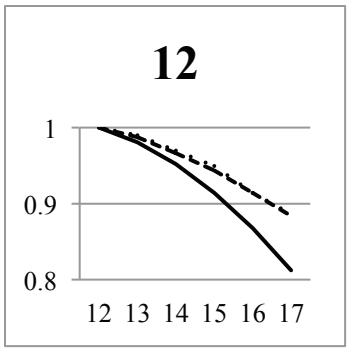


Figure 10-m. Observation Moment 12

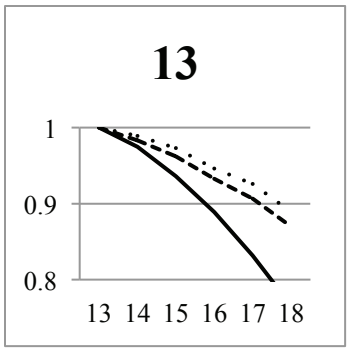


Figure 10-n. Observation Moment 13

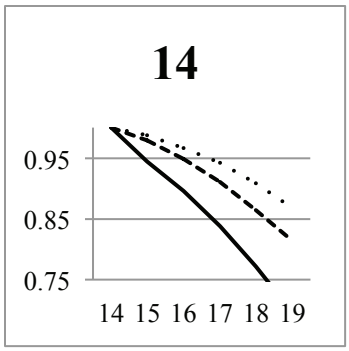


Figure 10-o. Observation Moment 14

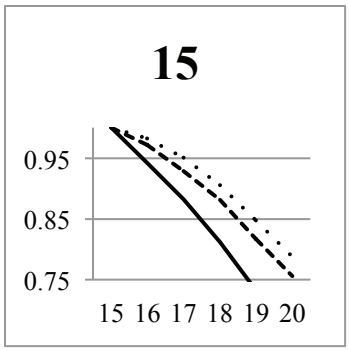


Figure 10-p. Observation Moment 15

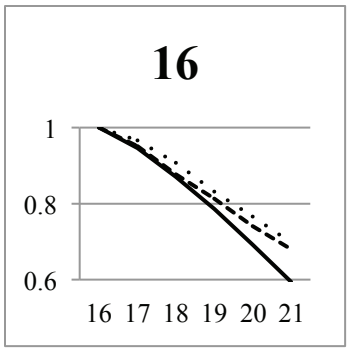


Figure 10-q. Observation Moment 16

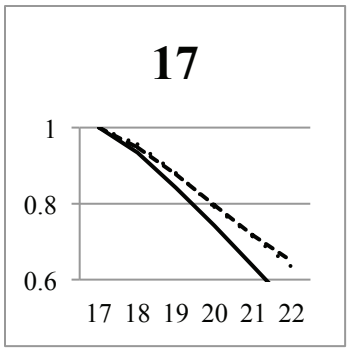


Figure 10-r. Observation Moment 17

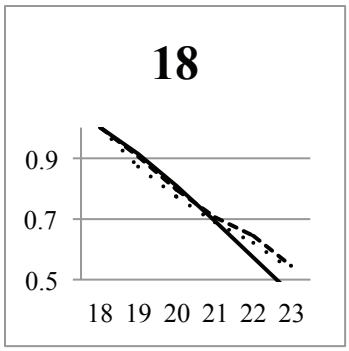


Figure 10-s. Observation Moment 18

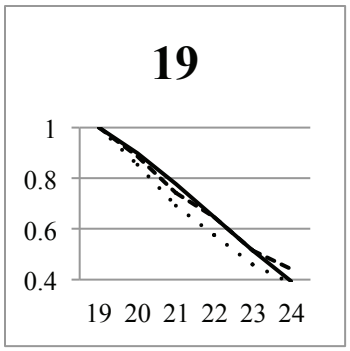


Figure 10-t. Observation Moment 19

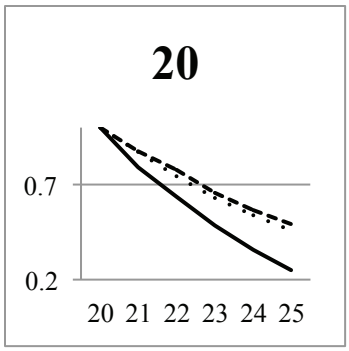


Figure 10-u. Observation Moment 20

The Figures 10-a to 10-i show that the PHM, hybrid greedy, and MILP methods have comparatively optimistic, moderate, and pessimistic outlooks from the early observation moments to the 8th observation. During these periods, the PHM gradually gets pessimistic and takes the place of the hybrid greedy at the 9th observation, the Figure 10-j. According to the Figures 10-j to 10-l, this continues up to the 11th observation when the PHM overtakes the MILP, the Figure 10-l. The Figures 10-l to 10-p show that: the hybrid

greedy, MILP, and PHM methods have comparatively optimistic, moderate, and pessimistic outlooks from the 11th to 15th observation. According to the Figures 10-q to 10-t, as the equipment gets closer to the actual failure moment, from the 16th to 19th observation, all the methods have similar outlooks about an imminent failure. The Figure 10-u shows that the PHM better warns about the failure at the last observation.

Table 31 shows the comparison between the average run-time of the three methods. It is shown that the run-time, as was expected, increases as the number of cut-points increases. In the train phase, the PHM method runs faster in comparison with the LAD methods. So, the PHM method is preferred in the cases where the train phase required to be performed frequently. In the test phase, the LAD methods run faster than the PHM method. So, the LAD methods are preferred in the cases where the test phase required to be performed frequently (online condition monitoring). In total, the PHM method comparatively runs faster in 4 out of 5 cases, while in the remaining case, the MILP method runs faster. Among the LAD methods, the MILP method runs, as was expected, much significantly faster than the hybrid greedy method.

Table 31. Run-Time: Hybrid Greedy vs. MILP vs. PHM

	Hybrid Greedy			MILP			PHM		
	Train	Test	Total	Train	Test	Total	Train	Test	Total
Sensitive Discriminating	2585.39	0.12*	2585.51	9.87	0.24	10.11	0.44+	3.07	3.51^
Equipartitioning 20	123.88	0.05*	123.93	2.18	0.38	2.56^	0.48+	2.66	3.14
Equipartitioning 30	670.14	0.08*	670.22	5.01	0.19	5.20	0.45+	2.79	3.24^
Equipartitioning 40	2293.49	0.10*	2293.59	8.56	0.26	8.82	0.44+	2.79	3.23^
Equipartitioning 50	5430.96	0.37	5431.32	11.29	0.33*	11.62	0.45+	2.84	3.29^

+ *Minimum mean value* (*Train*) * *Minimum mean value* (*Test*) ^ *Minimum mean value* (*Total*)

CHAPTER 4 **: CONCLUSION**

In this research, we developed an equipment failure prognostic model by employing the Logical Analysis of Data (LAD). We improved the LAD methodology to predict equipment's chance of survival at each observation moment when new data on the equipment health condition indicators is collected. The LAD model was applied on the Prognostics and Health Management Challenge dataset, a condition monitoring dataset provided by NASA Ames Prognostics Data Repository. Analysis of performance of the LAD model revealed that it provides comprehensible results that are greatly beneficial to maintenance practitioners. Prognostics results obtained using the LAD model, were compared with that using PHM model. Following result is only based on one example and need to be investigated further.

Comparison with respect to the accuracy of estimated MRL showed that: The conditional survival probability calculation method that equally favors the baseline and pattern survival probabilities statistically outperformed the one that prefers the pattern survival probability. The hybrid greedy method with the parameters *coverage* >10% and *fuzziness*= 0 statistically outperformed other hybrid greedy methods. The PHM statistically outperformed the both LAD methods. Also, it is noticed that the performances of the LAD model is highly sensitive to its defined survival function. However, the LAD model results are highly interpretable and easy to understand which is of great value for maintenance practitioners.

Comparison with respect to the run-time showed that: Fewer cut-points is preferred due to the fact that the accuracy of prognostics did not significantly depend much on the number of cut-points at the tested levels. In the train phase, the PHM method ran faster than the LAD methods, while in the test phase, the LAD methods ran faster than the PHM method. In 4 out of 5 cases, the PHM method ran faster in total. Among the LAD methods, the MILP method ran much significantly faster than the hybrid greedy method. Since the LAD methods were not statistically different, the MILP is preferred to the hybrid greedy, due to faster result achievement.

Our results also showed that the PHM method has an optimistic outlook about the equipment's survival. The LAD methods have neither constant optimistic nor constant pessimistic outlooks about the equipment's survival, whereas their outlooks change

gradually from pessimistic to optimistic, as the equipment health deteriorates over its lifetime. The PHM method is more stable at the early observation moments, while the LAD method stabilizes when the equipment gets older.

The LAD model has the advantage of not relying on any statistical theory, which enables it to overcome the conventional problems concerning the statistical properties of the datasets. Its main advantage is its straightforward process and self-explanatory results, which are greatly beneficial to maintenance practitioners.

Since the proposed LAD model is at its beginning phase, further research is required to improve the performance of the model. Due to the fact that the performances of the proposed calculation methods are highly sensitive to the defined survival function, a future research is to improve the survival function to reflect equipment's probable failure better. Due to the fact that the PH Model and the LAD model are stable at the early and the late observation moments, respectively, another future research direction is to investigate a hybrid LAD-PHM Model to benefit from both models' advantages. Another future research is to develop a technique to calibrate the LAD model to adjust for both underestimation and overestimation.

REFERENCE LIST

[1] S. Abramson, G. Alexe, P. L. Hammer, D. Knight and J. Kohn. A computational approach to predicting cell growth on polymeric biomaterials. *Journal of Biomedical Material Research 73A(1),* pp. 116-124. 2005.

[2] G. Alexe, S. Alexe, D. E. Axelrod, T. O. Bonates, I. Lozina, M. Reiss and P. L. Hammer. Breast cancer prognosis by combinatorial analysis of gene expression data. *Breast Cancer Research 8(4),* pp. R41. 2006-1.

[3] G. Alexe, S. Alexe, D. E. Axelrod, P. L. Hammer and D. Weissmann. Logical analysis of diffuse large B-cell lymphomas. *Artificial Intelligence in Medicine 34(3),* pp. 235-267. 2005.

[4] G. Alexe, S. Alexe and P. L. Hammer. Pattern-based clustering and attribute analysis. *Soft Computing 10(5),* pp. 442-452. 2006-2.

[5] G. Alexe, S. Alexe, P. L. Hammer and A. Kogan. Comprehensive vs. comprehensible classifiers in logical analysis of data. *Discrete Applied Mathematics 156(6),* pp. 870-882. 2008.

[6] G. Alexe, S. Alexe, P. L. Hammer and B. Vizvari. Pattern-based feature selection in genomics and proteomics. *Annals of Operations Research 148(1),* pp. 189-201. 2006-3.

[7] G. Alexe, S. Alexe, L. A. Liotta, E. Petricoin, M. Reiss and P. L. Hammer. Ovarian cancer detection by logical analysis of proteomic data. *Proteomics 4(3),* pp. 766-783. 2004.

[8] G. Alexe and P. L. Hammer. Spanned patterns for the logical analysis of data. *Discrete Applied Mathematics 154(7),* pp. 1039-1049. 2006-4.

[9] S. Alexe, E. Blackstone, P. L. Hammer, H. Ishwaran, M. S. Lauer and C. E. Pothier S. Coronary risk prediction by logical analysis of data. *Annals of Operations Research 119(1),* pp. 15-42. 2003.

[10] S. Alexe and P. L. Hammer. Pattern-based discriminants in the logical analysis of data. *Data Mining in Biomedicine 7(1),* pp. 3-23. 2007.

[11] S. Alexe and P. L. Hammer. Accelerated algorithm for pattern detection in logical analysis of data. *Discrete Applied Mathematics 154(7),* pp. 1050-1063. 2006.

[12] H. Almuallim and T. G. Dietterich. Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence 69(1–2),* pp. 279-305. 1994.

[13] H. Almuallim and T. G. Dietterich, "Learning with many irrelevant features," *Proceedings of the 9th National Conference on Artificial Intelligence(AAAI-91),* pp. 547-552, 1991.

[14] M. L. Araiza, R. Kent and R. Espinosa. Real-time, embedded diagnostics and prognostics in advanced artillery systems. *AUTOTESTCON Proceedings, 2002. IEEE* pp. 818-841. 2002.

[15] D. C. Baillie and J. Mathew. Comparison of autoregressive modeling techniques for fault diagnosis of rolling element bearings. *Mechanical Systems and Signal Processing 10(1),* pp. 1-17. 1996.

[16] D. Banjevic and A. K. S. Jardine. Calculation of reliability function and remaining useful life for a markov failure time process. *IMA Journal of Management Mathematics 17(2),* pp. 115-130. 2006.

[17] P. Baruah and R. B. Chinnam. HMMs for diagnostics and prognostics in machining processes. *International Journal of Production Research 43(6),* pp. 1275-1293. 2005.

[18] A. Bennane and S. Yacout. LAD-CBM; new data processing tool for diagnosis and prognosis in condition-based maintenance. *Journal of Intelligent Manufacturing 23(2),* pp. 265-275. 2012.

[19] T. O. Bonates, P. L. Hammer and A. Kogan. Maximum patterns in datasets. *Discrete Applied Mathematics 156(6),* pp. 846-861. 2008.

[20] E. Boros, P. L. Hammer, T. Ibaraki and A. Kogan. Logical analysis of numerical data. *Mathematical Programming 79(1),* pp. 163-190. 1997.

[21] E. Boros, P. L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz and I. Muchnik. An implementation of logical analysis of data. *Knowledge and Data Engineering, IEEE Transactions on 12(2),* pp. 292-306. 2000.

[22] E. Boros, T. Horiyama, T. Ibaraki and K. Makino. Finding essential attributes from binary data. *Annals of Mathematics and Artificial Intelligence 39(3),* pp. 223-257. 2003.

[23] E. Boros, T. Ibaraki and K. Makino. Logical analysis of binary data with missing bits. *Artificial Intelligence 107(2),* pp. 219-263. 1999.

[24] R. Bruni. Reformulation of the support set selection problem in the logical analysis of data. *Annals of Operations Research 150(1),* pp. 79-92. 2007.

[25] C. Bunks, D. McCarthy and T. Al-Ani. Condition-based maintenance of machines using hidden markov models. *Mechanical Systems and Signal Processing 14(4),* pp. 597-612. 2000.

[26] C. S. Byington, M. Watson and D. Edwards. Data-driven neural network methodology to remaining life predictions for aircraft actuator components. *Aerospace Conference, 2004. Proceedings. 2004 IEEE 6*pp. 3581-3589. 2004.

[27] S. S. Choi, K. S. Kang, H. G. Kim and S. H. Chang. Development of an on-line fuzzy expert system for integrated alarm processing in nuclear power plants. *IEEE Transactions on Nuclear Science 42(4),* pp. 1406-1418. 1995.

[28] V. Chvatal. A greedy heuristic for the set covering problem. *Mathematics of Operations Research 4(3),* pp. 233-235. 1979.

[29] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological) 34(2),* pp. 187-220. 1972.

[30] Y. Crama, P. L. Hammer and T. Ibaraki. Cause-effect relationships and partially defined boolean functions. *Annals of Operations Research 16(1),* pp. 299-326. 1988.

[31] M. Dong and D. He. Hidden semi-markov model-based methodology for multi-sensor equipment health diagnosis and prognosis. *European Journal of Operational Research 178(3),* pp. 858-878. 2007.

[32] J. Eckstein, P. L. Hammer, Y. Liu, M. Nediak and B. Simeone. The maximum box problem and its application to data analysis. *Computational Optimization and Applications 23(3),* pp. 285-298. 2002.

[33] Y. Fan and C. J. Li. Diagnostic rule extraction from trained feedforward neural networks. *Mechanical Systems and Signal Processing 16(6),* pp. 1073-1081. 2002.

[34] M. Friedman. A comparison of alternative tests of significance for the problem of *m* rankings. *The Annals of Mathematical Statistics 11(1),* pp. 86-92. 1940.

[35] M. L. Fugate, H. Sohn and C. R. Farrar. Vibration-based damage detection using statistical process control. *Mechanical Systems and Signal Processing 15(4),* pp. 707-721. 2001.

[36] K. B. Goode, J. Moore and B. J. Roylance. Plant machinery working life prediction method utilizing reliability and condition-monitoring data. *Proceedings of the Institution of Mechanical Engineers. 214(2),* pp. 109-122. 2000.

[37] A. B. Hammer, P. L. Hammer and I. Muchnik. Logical analysis of chinese labor productivity patterns. *Annals of Operations Research 87(0),* pp. 165-176. 1999.

[38] P. L. Hammer and T. O. Bonates, "Logical analysis of data - an overview: from combinatorial optimization to medical applications," *Annals of Operations Research,* vol. 148, pp. 203-225, 2006-1.

[39] P. L. Hammer, A. Kogan and M. A. Lejeune. Modeling country risk ratings using partial orders. *European Journal of Operational Research 175(2),* pp. 836-859. 2006-2.

[40] P. L. Hammer, A. Kogan, B. Simeone and S. Szedmak. Pareto-optimal patterns in logical analysis of data. *Discrete Applied Mathematics 144(1–2),* pp. 79-102. 2004-1.

[41] P. L. Hammer, Y. Liu, B. Simeone and S. Szedmak. Saturated systems of homogeneous boxes and the logical analysis of numerical data. *Discrete Applied Mathematics 144(1-2),* pp. 103-109. 2004-2.

[42] J. Han, N. Kim, M. K. Jeong and B. J. Yum. Comparisons of classification methods in the original and pattern spaces. *Expert Systems with Applications 38(10),* pp. 12432-12438. 2011-1.

[43] J. Han, N. Kim, B. J. Yum and M. K. Jeong. Pattern selection approaches for the logical analysis of data considering the outliers and the coverage of a pattern. *Expert Systems with Applications 38(11),* pp. 13857-13862. 2011-2.

[44] A. K. S. Jardine, D. Lin and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing 20(7),* pp. 1483-1510. 2006.

[45] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering,* 1960.

[46] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association 53(282),* pp. 457-481. 1958.

[47] K. Kim and H. S. Ryoo. A LAD-based method for selecting short oligo probes for genotyping applications. *OR-Spectrum 30(2),* pp. 249-268. 2008.

[48] S. Kotsiantis and D. Kanellopoulos, "Discretization techniques: a recent survey," *GESTS International Transaction Computer Science Engineering,* vol. 32, pp. 47-58, 2006.

[49] L. P. Kronek and A. Reddy. Logical analysis of survival data: Prognostic survival models by detecting high-degree interactions in right-censored data. *Bioinformatics 24(16),* pp. i248-i253. 2008.

[50] M. S. Lauer, S. Alexe, P. Snader Claire E., E. H. Blackstone, H. Ishwaran and P. L. Hammer. Use of the logical analysis of data method for assessing long-term mortality risk after exercise electrocardiography. *Circulation 106(6),* pp. 685-690. 2002.

[51] C. J. Li and T. Y. Huang. Automatic structure and parameter training methods for modeling of mechanical systems by recurrent neural networks. *Applied Mathematical Modelling 23(12),* pp. 933-944. 1999.

[52] Y. Li, S. Billington, C. Zhang, T. Kurfess, S. Danyluk and S. Liang. Adaptive prognostics for rolling element bearing condition. *Mechanical Systems and Signal Processing 13(1),* pp. 103-113. 1999.

[53] D. Lin and V. Makis. Filters and parameter estimation for a partially observable system subject to random failure with continuous-range observations. *Advances in Applied Probability 36(4),* pp. 1212-1230. 2004.

[54] H. Liu, F. Hussain, C. L. Tan and M. Dash. Discretization: An enabling technique. *Data Mining and Knowledge Discovery 6(4),* pp. 393-423. 2002.

[55] J. Luo, A. Bixby, K. Pattipati, L. Qiao, M. Kawamoto and S. Chigusa. An interacting multiple model approach to model-based prognostics. *Systems, Man and Cybernetics, 2003. IEEE International Conference on 1*pp. 189-194. 2003.

[56] E. Mayoraz and M. Moreira. Combinatorial approach for data binarization. *Principles of Data Mining and Knowledge Discovery 1704*pp. 442-447. 1999.

[57] L. M. Moreira, "The use of Boolean concepts in general classification contexts," *Doctoral Thesis, École Polytechnique Fédérale De Lausanne,* 2000.

[58] M. A. Mortada, "Applicability and interpretability of logical analysis of data in condition based maintenance," *Doctoral Thesis, École Polytechnique De Montréal,* 2010.

[59] M. A. Mortada, T. Carroll, S. Yacout and A. Lakis. Rogue components: Their effect and control using logical analysis of data. *Journal of Intelligent Manufacturing 23(2),* pp. 289-302. 2012.

[60] M. A. Mortada, S. Yacout and A. Lakis. Diagnosis of rotor bearings using logical analysis of data. *Journal of Quality in Maintenance Engineering 17(4),* pp. 371-397. 2011.

[61] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6 2(11),* pp. 559-572. 1901.

[62] Y. Peng, M. Dong and M. J. Zuo, "Current status of machine prognostics in condition-based maintenance: a review," *The International Journal of Advanced Manufacturing Technology,* vol. 50, pp. 297-313, 2010.

[63] K. W. Przytula and A. Choi. Reasoning framework for diagnosis and prognosis. *Aerospace Conference, 2007 IEEE* pp. 1-10. 2007.

[64] H. S. Ryoo. MILP approach to pattern generation in logical analysis of data. *Discrete Applied Mathematics 157(4),* pp. 749-761. 2009.

[65] S. Sampath, S. Ogaji, R. Singh and D. Probert. Engine-fault diagnostics: An optimisation procedure. *Applied Energy 73(1),* pp. 47-70. 2002.

[66] J. W. Sheppard and M. A. Kaufman. Bayesian diagnosis and prognosis using instrument uncertainty. *Autotestcon, 2005. IEEE* pp. 417-423. 2005.

[67] V. A. Skormin, L. J. Popyack, V. I. Gorodetski, M. L. Araiza and J. D. Michel. Applications of cluster analysis in diagnostics-related problems. *Aerospace Conference, 1999. Proceedings. 1999 IEEE 3*pp. 161-168. 1999.

[68] J. K. Spoerre. Application of the cascade correlation algorithm (CCA) to bearing fault classification problems. *Computers in Industry 32(3),* pp. 295-304. 1997.

[69] M. Stanek, M. Morari and K. Frohlich. Model-aided diagnosis: An inexpensive combination of model-based and case-based condition assessment. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 31(2),* pp. 137-145. 2001.

[70] Z. Tian. An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring. *Journal of Intelligent Manufacturing 23(2),* pp. 227-237. 2012.

[71] W. Wang. A model to predict the residual life of rolling element bearings given monitored condition information to date. *IMA Journal of Management Mathematics 13(1),* pp. 3-16. 2002.

[72] Z. Wen, J. Crossman, J. Cardillo and Y. L. Murphey. Case-base reasoning in vehicle fault diagnostics. *Proceedings of the International Joint Conference on Neural Networks. 4*pp. 2679-2684. 2003.

[73] S. Yacout. Fault detection and diagnosis for condition based maintenance using the logical analysis of data. Presented at Computers and Industrial Engineering (CIE), 2010 40th International Conference on. 2010, .

[74] R. C. M. Yam, P. W. Tse, L. Li and P. Tu. Intelligent predictive decision support system for condition-based maintenance. *The International Journal of Advanced Manufacturing Technology 17(5),* pp. 383-391. 2001.

[75] J. Yan, M. Koc and J. Lee. A prognostic algorithm for machine performance assessment and its application. *Production Planning Control 15(8),* pp. 796-801. 2004.

[76] J. Ying, T. Kirubarajan, K. R. Pattipati and S. Deb. A hidden markov model-based algorithm for online fault diagnosis with partial and imperfect tests. *AUTOTESTCON '99. IEEE Systems Readiness Technology Conference, 1999. IEEE* pp. 355-366. 1999.

[77] X. Zhang, R. Xu, C. Kwan, S. Y. Liang, Q. Xie and L. Haynes. An integrated approach to bearing fault diagnostics and prognostics. *Proceedings of the American Control Conference 4*pp. 2750-2755. 2005.