A study on the use of discrete event data for prognostics and health management: discovery of association rules

Bin Liu[1], Phuc Do[2], Benoit Iung[2], Min Xie[3] , Flavien Peysson[4] and Mayank Jha[2]

[1]*Department of Civil and Environmental Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada,*
b274liu@uwaterloo.ca

[2]*Lorraine University, CRAN, CNRS UMR 7039, Campus Sciences BP70239, 54506 Vandoeuvre, France*
phuc.do@univ-lorraine.fr
benoit.iung@univ-lorraine.fr
mayank-shekhar.jha@univ-lorraine.fr

[3]*Department of Systems Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong*
minxie@cityu.edu.hk

[4]*PREDICT Company, 19, avenue de la Fort de Haye, CS 10508, 54519 Vandoeuvre-les-Nancy Cedex, France*
flavien.peysson@predict.fr

ABSTRACT

This study addresses prognostics and health management (PHM) for manufacturing machines. Different from previous researches where continuous monitoring is assumed for PHM, we investigate the issue with discrete event data. Various event data were recorded during system operation, which can provide useful information for fault diagnosis and failure prediction. We focus on discovery of association rules based on the industrial discrete event data. Events that occur together frequently are classified into event groups. Apriori algorithm is employed to discover the frequent event groups and identify strong association rules (occurrence of the events is highly dependent). To accommodate the algorithm, the initial event data is transformed into the form of transactional data. The obtained association rule estimates the occurrence probability of certain significant events within specified time interval. It is concluded through a case study that the number of frequent event groups and strong association rules increases with the time interval that the events are grouped as one transaction.

1. INTRODUCTION

With the fast development of information and sensing technology, prognostics and health management (PHM) has been widely used in modern systems to provide real time data management and processing. PHM predicts the future reliability and performance of a system based on the current and past condition monitoring data, which is usually collected via continuous monitoring (Tsui et al, 2015; Vogl et al, 2016). The collected real time measurements allow PHM to develop sophisticated models and predict system behavior accurately.

In literature, numerous degradation models have been established by taking advantage of the condition monitoring data, e.g., Wiener-process-based degradation models, Markov chain models, hidden Markov chain models and filter-based models (Moura et al, 2013; Si et al, 2013; Lee et al, 2014; Lin et al, 2015; Liu et al, 2016; Chandar & Panda, 2017). Si et al (2013) proposed a Wiener-process-based degradation model to estimate the remaining useful lifetime of a continuously monitored system. Vrignat et al (2015) developed a hidden Markov model to predict failure events, and they concluded that the performance of hidden Markov model is superior to traditional survival analysis. Belkacem et al (2017) investigated PHM of a hybrid dynamic system, where optimal maintenance policy and remaining useful life are evaluated under continuous monitoring.

Continuous monitoring is, however, costly or even impossible for certain complex systems, despite its prevalence and effectiveness in PHM. In cases where continuous monitoring is not available, an alternative would be to employ discrete event data for prognostic purposes. Compared with continuous monitoring, which requires a dedicated sensor, discrete event data can be readily recorded and transmitted during machine operation. The events are stored in logs, referred to as event/error logs (Oliner et al, 2012). Usually, the event log contains information of system operation, which can be used to monitor system condition and provide information for PHM. For example, in a manufacturing machine, there are thousands or even millions of events during operation of one week. The event log data consist of various events/errors during machine operation along a timeline. Various information is available with respect to the state of machine in operation, such as operation mode change, operating pressure/temperature unsatisfied, etc..

The logic behind event log analysis is that failure signatures/symptoms usually appear ahead of the final failure of a machine. The precedent failure signatures provide additional information for failure prediction. In practice, discrete event data were used manually to identify the failure signature based on area expert experience and physical mechanism of the operating machine, which, obviously, is time-consuming. Approaches that can adequately utilize the discrete event data are warranted for machine prognostics and health management.

Several techniques exist to deal with discrete event data in literature. They can be roughly classified into design-based approaches and data-driven-based methods (Hatonen et al, 1996; Mannila et al, 1997; Li et al, 2007; Allison, 2014). However, research on discrete event data for PHM is quite limited. Li et al (2007) proposed a failure prediction method based on a Cox proportional hazard model, where the frequent failure signatures are incorporated as the covariates of the proportional hazard model. Subsequently, Yuan et al (2011) extended the work of Li et al (2007) by fitting the Cox proportional hazard model with the event log data and evaluating the influence of covariance factors on the survival function. One deficiency of the above studies is that they assume the system failure rate follows Cox proportional hazard model, which, however, may not hold true in reality, due to the increasing complexity of modern systems. Fronza et al (2013) proposed a failure prediction method based on the event log data, where random indexing and support vector machine are employed to identify the sequence or pattern of events. Russo et al (2015) developed a novel method which integrates multiple machine learning techniques to address data brittleness and improve the robustness of the predicted results. Case study on a telemetry system is investigated to show the effectiveness of the proposed approach. Support vector machine is an effective tool to classify the events into failure signature or not. However, it fails to identify the casual links of the events.

In the present study, we aim to investigate the influence of discrete event data on PHM of manufacturing machines. An association rule learning approach is employed to identify the links between the events. Compared with the existing methods, the proposed method is more flexible in that it generates multiple association rules so that the manufacturers can select the rules of interest. Since significant rule may not necessarily be the most frequent events from the mathematical point of view, but can be the ones that have significant consequences in practice, our method allows for a comprehensive study of machine prognostic and health management.

The rest of the study is organized as follows. Section 2 briefly describes the industrial issue and the available dataset. Section 3 presents the association rule learning method for PHM analysis with the discrete data. Application on a manufacturing machine is performed in Section 4 to show the effectiveness of the data mining approach. Finally, conclusions and future directions are provided in Section 5.

2. PROBLEM DESCRIPTION

Our study is performed based on industrial discrete event data of a large manufacturing company in France, named as Predict, Inc. Predict, Inc used to monitor the manufacturing machines with dedicated sensors. However, despite the very effort of continuous monitoring, failures still occur with profound implications. In addition, continuous monitoring is expensive and sometimes is technically impossible for several special units. Therefore, use of discrete event data for PHM is fully of interest of Predict, Inc.

Table 1 presents the sample event logs of the manufacturing machine. As is shown in Table 1, the event data consists of the event code, description of the event, occurring date and the associated controller. The event data are recorded by Programmable Logic Controller (PLC) or Network Control Unit (NCU). Different from previous work where system failure or error is specified in the event logs, the present industrial data only contain various events, while system failure is not indicated along with the events. We aim to discover whether there exist significant patterns of the event that can be used for machine prognostics and health management.

3. ASSOCIATION RULE LEARNING WITH DISCRETE EVENT DATA

For a given discrete event data, the pattern of the events can be discovered via statistical models or data mining methods (e.g., sequence mining and association rule mining). In the present study, we will focus on the association relations among the discrete events. Association rule learning is a rule-based machine learning method that is used to identify the significant relations of the events in large database (Agrawal et al, 1993; Sarno et al, 2015). Several strong rules are discovered to provide insights of machine operation and further applied for decision making.

The formal description of association rule learning is stated as follows: Let $\mathbf{I}=\{I_1, I_2, ..., I_m\}$ be the set of items and $\mathbf{T}$ be the set of transactions. Each transaction is associated with a unique Transaction ID. A transaction T_i consists of a set of items, where $T_i \subseteq \mathbf{I}$. T_i is said to contain X, a set of items in $\mathbf{I}$, if $X \subseteq T$. An association rule of two item sets is denoted as $X \rightarrow Y$, where $X \subset \mathbf{I}$, $Y \subset \mathbf{I}$ and $X \cap Y = \varnothing$. Support and Confidence are two major measures of the association rule. Support measures the frequency of a certain item set in the dataset, which is defined as the proportion of transaction $t \in \mathbf{T}$ in the transaction set $\mathbf{T}$ which contains item set X,

$$\text{supp}(X) = \frac{\{t \in \mathbf{T}; X \subseteq t\}}{|\mathbf{T}|} \quad (1)$$

Table 1. Sample event logs of the manufacturing machine

Code	Event description	Date	Controller
510012	Non-compliant mode of operation	11.06.15 07:36:28	PLC
510219	Sprinkling filtered water tank maximum level	17.06.15 17:10:24	PLC
700135	Counterstock / counterbalanced spindle	18.06.15 00:58:18	PLC
6406	Channel 1 Acknowledgment AP missing for instruction 03	24.06.15 00:53:40	NCU
601114	Gear reduction of the main spindle is active	26.06.15 14:23:44	PLC

A rule $X \rightarrow Y$ is said to have support s if s% of the transactions $\mathbf{T}$ contains $X \cup Y$. Confidence measures the credibility of the rule. The confidence of a rule, $X \rightarrow Y$, is defined as the proportion of transactions that contain both item set X and Y in transactions that contain X,

$$\text{conf}(X \rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(X)} \quad (2)$$

If the transaction data is large enough, support can be interpreted as the probability that a certain item set appears in one transaction, while the confidence can be interpreted as the estimate of conditional probability $P(E_Y | E_X)$, where E_X (E_Y) denote the event that item set X (Y) is included in one transaction.

Given a set of transaction data, the goal of association rule learning is to find out all the association rules that have support and confidence larger than the user-specified thresholds. The result will typically appear in the following form:

If event A or B occurs during machine operation

Then event C will occur within the Time Interval with Confidence *c*%

The association rule learning typically consists of two parts: large item sets identification and association rule establishment. Many approaches have been developed to discover frequent item sets, e.g., Apriori algorithm, Eclat algorithm, and frequent-pattern-growth algorithm (Agrawal & Srikant,1994; Zaki, 2000; Mishra & Choubey, 2012). In the following, Apriori algorithm will be employed in our analysis, due to its easy implementation in practice (Agrawal & Srikant, 1994; Rudin et al, 2013). However, before implementing Apriori algorithm, the initial event data has to be pre-processed so as to fit the algorithm.

3.1. Data Pre-processing

One of the major challenges in event log analysis is to abstract the event structures, since the log data is usually unstructured. To make the event data suitable for association rule learning, we first pre-processed the initial data. Although the initial event data contains multiple dimensions, we were interested in the event code and the associated occurrence date. Note that each event was recorded at its occurrence, while for the association rule learning method, the input data should be in the form of transactions. As a first step, we treated the events that occur within certain time interval as one transaction. The interval was selected in terms of the timestamp of the events, such as 10 minutes, 30 minute, 1hour, etc.. Within the time interval, the identical events were merged to avoid repetition. In other words, the events are recorded only once even if they appear multiple times within the time interval. The sample event data after pre-processing is presented in Table 2.

Table 2. Sample event data for association rule learning

Transaction ID	Item sets (event code)
1	{'700137', '700136', '700135', '700454', '601011', '700146', '511311', '700143', '510011'}
2	{'700454', '67834', '700205', '6413'}
3	{'511311', '67834', '6413', '10208'}
4	{'600914', '700137', '700136', '700135', '6413', '700146', '700143', '510011'}
5	{'600914', '601011', '6413'}
6	{'300951', '700339', '700338', '510313', '700335', '600112', '700337', '700336', '27002', '700332', '27006', '701957', '700741', '700340', '700035'}

3.2. Discovering frequent item sets & strong association rules

Before arriving at association rule discovery, we first need to find out the frequent item sets from the discrete event dataset. Apriori algorithm is an efficient algorithm to mine frequent item sets and learn association rule over transactional databases. The discovered large item sets are further used to determine strong association rules. Apriori algorithm was first proposed by Agrawal and Srikant (1994) to operate on discrete data containing transactions.

Apriori algorithm consists of three steps. First the algorithm counts the event occurrence to discover all frequent 1-item set. Subsequently, at the *k*th step, the algorithm creates all candidate item sets C_k by extending the previous frequent item set L_{k-1} with Apriori Judgement. Next the candidate item sets are pruned in terms of the support of the item sets. The procedure of Apriori algorithm is shown as follows.

Apriori Algorithm to discover frequent item sets:

1. Generate all frequent 1-item sets, L_1.

2. Create candidate item sets C_k.

for ($k=2; L_{k-1} \neq \varnothing; k++$), do

$$C_k = creat\_C_k(L_{k-1})$$

3. Prune the candidate item sets to generate frequent item sets L_k

for all transactions $T_i \in \mathbf{T}$, do

$$L_k = generate\_L_k(C_k)$$

4. Output all the frequent item sets $\cup_k L_k$.

The $creat\_C_k(L_{k-1})$ function generates frequent candidate item sets according to Apriori property. The previous frequent item sets L_{k-1} is taken as input argument and the superset of L_{k-1} is generated as output. The function $generate\_L_k(C_k)$ is used to discover all the subset in C_k that exhibit a support larger than the minimal support threshold. Any subset that cannot satisfy the support criterion is deleted from the frequent item sets. Hash Tree is used to store the frequent candidate item sets C_k (Agrawal & Srikant, 1994).

Bottom-up approach is used to extend the frequent item sets by one item at each candidate generation step. The algorithm terminates when no frequent item sets can satisfy the minimal support threshold. Breadth-first search and a Hash tree structure is employed to count candidate item sets efficiently. Apriori algorithm has been widely applied in various industries, e.g., market basket analysis, medical image classification, and web usage mining (Hipp et al, 2000; Kumar & Rukmani, 2010).

After generating the frequent item sets L, the association rules of the events were determined. As a first step, we searched all the non-empty subsets of the frequent item sets L, and output the rule $A \rightarrow L-A$ if the confidence *c* (ratio of $\text{supp}(L)$ to $\text{supp}(A)$) is larger than the minimal confidence threshold. The result was interpreted as follow: given the events A occur, events $L-A$ will occur within the time interval with confidence *c*%. Since the frequent item sets were stored in hash tables, the support could be calculated efficiently. The procedure of association rules identification is as follows.

Algorithm to identify strong association rules:

for all frequent item sets L_k, do

for all subsets of L_k, $A \subset L_k$, do

$$\text{conf}(\mathrm{A} \rightarrow L_k - A) = \frac{\text{supp}(L_k)}{\text{supp}(A)}$$

if $\text{conf}(\mathrm{A} \rightarrow L_k - A) > minconf$

output rule $\mathrm{A} \rightarrow L_k - A$ with the confidence $\text{conf}(\mathrm{A} \rightarrow L_k - A)$ and support $\text{supp}(L_k)$

4. APPLICATION ON MANUFACTURING MACHINES

The industrial data in our analysis is collected from a manufacturing machine. The data are collected in one and a half month (from 11 June 2015 to 28 July 2015), which records various events during the machine operation. In total, 151 different types of events are recorded in the event log.

The industrial data are stored in an Excel file and handled by Python. To have an intuitive glance of the event data, we first counted the events according to their records in the database. Table 3 lists the 10 most frequent events, ranking from highest to lowest. The list of frequent events provides somewhat importance ranking of events and help managers and engineers to focus on the events that have significant impact on machine operation.

Based on the Apriori algorithm, the initial discrete event data is transformed into the type of transactional data. To achieve this, we treated the events within certain time interval as one transaction. The time interval is determined by the timestamp of the events or by area experts. For example, the time interval of interest should exceed the lead time of maintenance actions so that the engineers are able to intervene the machine in time. To illustrate the relations among the events, we set the time interval as 10 minutes. The data after pre-processing is referred to as 10-minute transactional data.

For the 10-minute transactional data, the frequent event groups are presented in Table 4 and the strong association rules are shown in Table 5. The minimal support threshold was set as 0.1 and the minimal confidence threshold 0.7. Note that we only list the 3-item event group in Table 4, as 3 items is the largest group. According to the down closure lemma, which states that any subset of frequent item sets is also frequent, other frequent event groups can be extracted from the 3-item groups. As can be observed in Table 4, the most frequent event group is {600914, 6413, 601011}, with support 0.246, which indicates that the events {600914, 6413, 601011} will occur within 10 minutes with probability 0.246. Table 5 presents the strong association rules that have confidence larger than 0.7. As can be observed in Table 5, the most significant rule is {601011, 6413} 600914, with confidence 0.946, which implies that given that event {601011, 6413} occur, event 600914 will occur within 10 minutes with the probability 0.946.

Table 3. List of 10 most frequent events.

Event code	Description
600914	Fluid feed coupling FS1 not tightened
6413	Channel 1 Tool
601114	Gear reduction of the main spindle is active
601011	Cutter spindle FS1 not in OFF state
700454	Tool changer holder on the non-closed machining space
27006	Axis external pulse erase test in progress
27002	Axis stop test in progress
700205	Hydraulic oil overtemperature weight balancing
510012	Non-compliant mode of operation
300951	Axis stop drive test in progress

Table 4. List of frequent event groups for 10-minute transactional data.

Event group	Support
700454, 600914, 6413	0.204
600914, 6413, 601011	0.246
700454, 600914, 601011	0.176
700454, 6413, 601011	0.159

In addition, it is interesting to find out how the number of frequent event groups and strong association rules vary with the time interval. Figure 1 shows the result for the case where the minimal support threshold and minimal confidence threshold is set as 0.1 and 0.9. It is quite obvious that both the number of frequent event groups and strong association rules increase with time interval. Moreover, we plot in Figure 2 the relationship between the number of strong association rules and the minimal support threshold. Clearly the number of strong association rules decreases with the minimal support threshold.

Table 5. List of strong association rules for 10-minute transactional data.

Antecedents	Consequents	Confidence
600914	601011	0.709
601011	600914	0.9
601011	6413	0.766
700454	600914	0.858
600914	6413	0.767
700454	6413	0.792
700454, 600914	6413	0.835
700454, 6413	600914	0.903
700454, 6413	6413, 600914	0.716
600914, 601011	6413	0.805
600914, 6413	601011	0.744
601011, 6413	600914	0.946
601011	6413, 600914	0.724
700454, 600914	601011	0.719
700454, 601011	600914	0.935
700454, 601011	6413	0.846
700454, 6413	601011	0.704

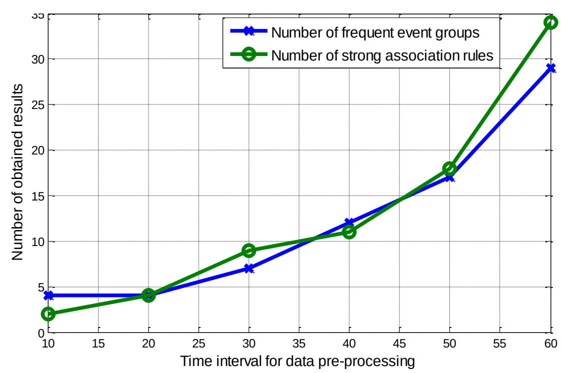

Figure 1. Number of frequent event groups and strong association rules with different time intervals.

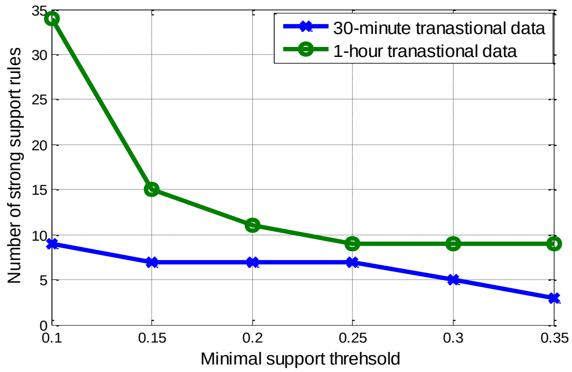

Figure 2. Number of frequent event groups and strong association rules with different time intervals.

5. CONCLUSION

In this study, we investigate the prognostic and health management of a manufacturing machine with discrete event data. Data mining approaches are applied to discover frequent event groups and strong association rules. The association rules are obtained with industrial data from a manufacturing company, which can serve as an precursor for abnornal events. Preventivre mainenance actions can be effectively implemented upon observing the precursor events so as to prevent system failure, which contributes to the machine prognostics and health management.

Future research can be conducted in the following directions. The discovered association rules can be incorporated into failure prediction, combined with continuous monitoring. A combination of continuous observation and discrete event data contributes to a more accurate failure prediction. The associated events can be treated as abrupt changes of the system states and carefully embedded within a degradation model (e.g., covariates in Cox proportional hazards model).

ACKNOWLEDGEMENT

Special thanks to Predict for the collaboration through this project, National Research Agency, the University of Lorraine, the National Center for Scientific Research for supporting and financing LabCOM PHM factory, our joint laboratory.

REFERENCES

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *In Acm sigmod record*, June.

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. *In Proc. 20th int. conf. very large data bases*, VLDB, September.

Allison, P. D. (2014). *Event history and survival analysis: Regression for longitudinal event data*. SAGE publications.

Belkacem, L., Simeu-Abazi, Z., Dhouibi, H., Gascard, E., & Messaoud, H. (2017). Diagnostic and prognostic of hybrid dynamic systems: Modeling and RUL evaluation for two maintenance policies. *Reliability Engineering & System Safety*, vol. 164, pp. 98-109.

Chandar, S., & Panda, S. (2017). Degradation Detection and Diagnosis of Inductors in LCL Filter Integrated with Active Front End Rectifier. *IEEE Transactions on Power Electronics*. In press.

Fronza, I., Sillitti, A., Succi, G., Terho, M., & Vlasenko, J. (2013). Failure prediction based on log files using random indexing and support vector machines. *Journal of Systems and Software*, vol. 86, pp. 2-11.

Hatonen, K., Klemettinen, M., Mannila, H., Ronkainen, P., & Toivonen, H. (1996). TASA: Telecommunication alarm sequence analyzer or how to enjoy faults in your network. *In Network Operations and Management Symposium*, April.

Hipp, J., Güntzer, U., & Nakhaeizadeh, G. (2000). Algorithms for association rule mining—a general survey and comparison. *ACM sigkdd explorations newsletter*, vol. 2, pp. 58-64.

Kumar, B. S., & Rukmani, K. V. (2010). Implementation of web usage mining using APRIORI and FP growth algorithms. *Int. J. of Advanced networking and Applications*, vol. 1, pp. 400-404.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., & Siegel, D. (2014). Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. *Mechanical systems and signal processing*, vol. 42, pp. 314-334.

Li, J., Shen, H., & Topor, R. (2002). Mining the optimal class association rule set. *Knowledge-Based Systems*, vol. 15, pp. 399-405.

Li, Z., Zhou, S., Choubey, S., & Sievenpiper, C. (2007). Failure event prediction using the Cox proportional hazard model driven by frequent failure signatures. *IIE Transactions*, vol. 39, pp. 303-315.

Lin, Y. H., Li, Y. F., & Zio, E. (2015). Integrating random shocks into multi-state physics models of degradation processes for component reliability assessment. *IEEE Transactions on Reliability*, vol. 64, pp. 154-166.

Liu, B., Xie, M., Xu, Z., & Kuo, W. (2016). An imperfect maintenance policy for mission-oriented systems subject to degradation and external shocks. *Computers & Industrial Engineering*, vol. 102, pp. 21-32.

Mannila, H., Toivonen, H., & Verkamo, A. I. (1997). Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery, vol. 1, pp. 259-289.

Mishra, R., & Choubey, A. (2012). Discovery of frequent patterns from web log data by using FP-growth algorithm for web usage mining. *International Journal* *of Advanced Research in Computer Science and Software Engineering*, vol. 2.

Moura, S. J., Stein, J. L., & Fathy, H. K. (2013). Battery-health conscious power management in plug-in hybrid electric vehicles via electrochemical modeling and stochastic control. *IEEE Transactions on Control Systems Technology*, vol. 21, pp. 679-694.

Oliner, A., Ganapathi, A., & Xu, W. (2012). Advances and challenges in log analysis. *Communications of the ACM*, vol. 55, pp. 55-61.

Rudin, C., Letham, B., & Madigan, D. (2013). Learning theory analysis for association rules and sequential event prediction. *The Journal of Machine Learning Research*, vol. 14, pp. 3441-3492.

Russo, B., Succi, G., & Pedrycz, W. (2015). Mining system logs to learn error predictors: a case study of a telemetry system. *Empirical Software Engineering*, vol. 20, pp. 879-927.

Sarno, R., Dewandono, R. D., Ahmad, T., Naufal, M. F., & Sinaga, F. (2015). Hybrid Association Rule Learning and Process Mining for Fraud Detection. *IAENG International Journal of Computer Science*, vol. 42.

Si, X. S., Wang, W., Hu, C. H., Chen, M. Y., & Zhou, D. H. (2013). A Wiener-process-based degradation model with a recursive filter algorithm for remaining useful life estimation. *Mechanical Systems and Signal Processing*, vol. 35, pp. 219-237.

Tsui, K. L., Chen, N., Zhou, Q., Hai, Y., & Wang, W. (2015). Prognostics and health management: A review on data driven approaches. *Mathematical Problems in Engineering*.

Vogl, G. W., Weiss, B. A., & Helu, M. (2016). A review of diagnostic and prognostic capabilities and best practices for manufacturing. *Journal of Intelligent Manufacturing*, pp. 1-17.

Vrignat, P., Avila, M., Duculty, F., & Kratz, F. (2015). Failure event prediction using hidden markov model approaches. *IEEE Transactions on Reliability*, vol. 64, pp. 1038-1048.

Yuan, Y., Zhou, S., Sievenpiper, C., Mannar, K., & Zheng, Y. (2011). Event log modeling and analysis for system failure prediction. *IIE Transactions*, vol. 43, pp. 647-660.

Zaki, M. J. (2000). Scalable algorithms for association mining. *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, pp. 372-390.