1-1-2017

Evolving Clustering Algorithms And Their Application For Condition Monitoring, Diagnostics, & Prognostics

Fling Finn Tseng
Wayne State University,

EVOLVING CLUSTERING ALGORITHMS AND THEIR APPLICATION FOR CONDITION MONITORING, DIAGNOSTICS, & PROGNOSTICS

by

FLING FINN TSENG

DISSERTATION

Submitted to the Graduate School

of Wayne State University

Detroit, Michigan

in partial fulfillment of the requirements

for degree of

DOCTOR OF PHILOSOPHY

2017

MAJOR: INDUSTRIAL ENGINEERING

Approved By:

_____

Advisor Date

_____

_____

_____

_____

_____

DEDICATION

To my parents for their hard work and supports of my post-graduate education abroad,

To my advisor, Dr. Ratna Babu Chinnam, for his inputs through the whole process,

To my advisor Dr. Dimitar Filev for his inspirational pioneering work,

To my wife, Hsinyi Lee for her kindness and patience,

and

To my son, Ian Tseng.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I INTRODUCTION

This chapter provides the motivation for the research and concludes with a statement of research. We start first with a brief introduction to common maintenance engineering practices before describing Condition-based Maintenance (CBM) and associated benefits. We also discuss the domain of evolving models and associated benefits. These sections form the foundation for discussing the specific research objectives at the end of the chapter.

1.1 CONDITION-BASED MAINTENANCE

Maintenance plays an important role to ensure the capacity of a system is being kept at a high level by keeping critical equipment and components' overall availability, reliability and safety in check. Traditionally, maintenance activities can be classified into three categories, improvement maintenance (IM), preventive maintenance (PM) and corrective maintenance (CM) [Pat02]. IM aims at reducing the need of any maintenance activities in the design and operation phase of a system and as a result is highly cost prohibitive and impractical for most equipment or systems [Yan02]. CM deals with restoring a machine's functionality after occurrence of failures. Often times, these failures cause line stoppage in production facilities and damage not only to the system itself but other systems as well. For certain equipment such as special purpose military vehicles or airplanes, any occurrence of a catastrophic failure is unacceptable due to potential loss of life and other losses.

Preventive maintenance focuses on activities that will prevent or minimize the occurrences of a failure. In turn, PM activities are considered effective when the levels of desired asset availability and reliability are high [Rao92]. Typical practices of PM are time or cumulative usage based where certain maintenance activities are prescribed as the asset reaches certain time or cumulative based milestone. Effectively, PM activities trade-off more frequent maintenance activities to unplanned (and at times catastrophic) failures that require CM. It improves total uptime by reducing downtime incurred by failures at the cost of some downtime associated with

PM activities. In PM, one of the most difficult decisions is related to the determination of the maintenance period. Since PM activities often involves change or recondition several components or subsystem, total time, labor and money invested in PM typically increases as maintenance period decreases. For example, if one were to employ the mean-time-between-failures (MTBF) as the maintenance period, one would expect half the assets to experience some type of a failure while the others will experience none. To reduce the number of expected failures, one can reduce the maintenance period albeit at a potentially higher overall cost of maintenance [LT97]. Figure 1 illustrates this trade-off of maintenance frequency vs. total cost.

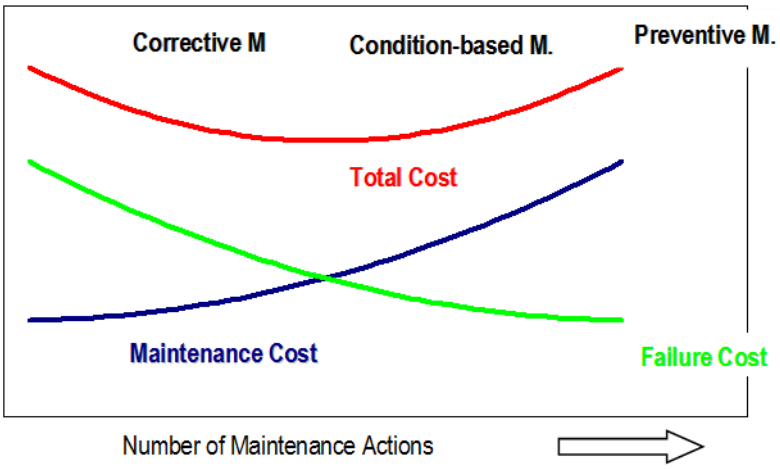


Figure 1: Illustration of the concept of optimal maintenance with failure and maintenance costs [F05]

Alternative to PM or CM, one can also consider the concept of Condition-based Maintenance (CBM) or predictive maintenance [Yan02]. CBM focuses on monitoring the condition of the asset and warrants maintenance activities only if impending failures are predicted or some indicative signatures suggesting unacceptable level of performance deteriorations have been observed [BG01, OSL02].

Traditionally, diagnostics and prognostics functionalities are the foundation of such CBM practices. “CBM is a maintenance program that recommends maintenance decision based on the information collected through condition monitoring” [JL06]. While the extremely vast extant literature reports good success in developing highly effective diagnostic algorithms for certain

classes of components and equipment (such as bearings, pumps, and motors), leading to so called “point solutions,” most of these successes are based on decades of academic and industrial research involving characterization and modeling of equipment behavior through extensive, expensive and involved physics based modeling that requires in-depth knowledge of the system (e.g., physics driven models) [BMT99], [KR02a], [HZ09] and [LOS09]. It is understandable that such approaches are warranted in dealing with mission-critical systems (that might involve loss of life or incur large financial costs due to its malfunction or breakage). However, methods that are more cost-effective, produce accurate results and with ease of calibration yet mass-deployable are also desirable for various reasons. Mainly, they provide the means for cost effectively monitoring a wide-array of equipment and systems that are not considered to be non-mission-critical. When properly designed and implemented effectively, unnecessary maintenance activities and cost can be reduced through accurate diagnostics and prognostics information [JL06].

1.2 PROGNOSTICS

Equipment prognostics provide valuable insight into a meaningful future time or usage horizon regarding the status of the health and remaining useful life of a system. Most widely used definition of prognostics is to predict how much (usage) time is remaining before a failure occurs [JL06]. Alternatively, it can be also defined as to the probability that an asset will operate fault and failure free up to a meaningful future time [LM030], [SF03]. In turn, it provides opportunities for better planning and scheduling of both operation and maintenance activities with more informed decisions as to the contingency plan for affected sites, personnel and facilities and timely acquisition of maintenance parts and other necessary resources for maintenance activities.

Prognostics models often built upon diagnostics and it is the process of estimating the Remaining Useful Life (RUL) of a machine by predicting the progression of a diagnosed failure [F05]. Statistical methods such as autoregressive integrated moving average (ARIMA) models and Hidden Markov Models (HMM) [KT08], machine learning methods such as Artificial Neural

Networks (ANN), Support Vector Machines (SVM) and Relevance Vector Machines (RVM) and Monte Carlo methods such as Particle Filter (PF) have been studied as the engine for prognostics modeling purposes [ZL11]. Not to be confused with "*mean time to failure*" (MTTF) or life expectancy that has to do with "*a population*" of the same or at least similar machine or component's "*average operational life*", RUL is specifically defined as the time to the next failure for "*one particular machine"* or component being monitored [EGBH00].

Different from a diagnostics task that typically determines type and severity of a fault and whether the machine is "*presently*" in a known faulty state, carrying out a prognostics task is much more difficult in practice due to the need to forecast information into the future. Since the underlying degradation phenomenon is stochastic in nature, accuracy, precision and confidence of a model need to be considered. Accuracy is the measure of closeness between estimated and actual time of failure. Precision is the model's ability to remain accurate as the time horizon increases. Confidence is the probability that the actual RUL falls within the estimated RUL interval. As a result, high accuracy, narrow precision and high confidence values are desirable [EGBH00], [F05].

In the aircraft industry, systems comprised of diagnostics and prognostics capabilities are commonly referred to as Integrated Vehicle Health Management (IVHM). Information generated from IVHM is highly desirable for not only OEMs but also for product maintenance and service providers [EJJ13]. There is a growing trend in the aviation industry of moving away from traditional "producer only" role to a more service oriented business model. Due to the effectiveness of IVHM based maintenance strategies, the demand has been substantial from the airplane operators to retrofit their legacy fleet (that do not come with necessary hardware) with OEM IVHM hardware to take advantage of the added benefits of IVHM based prognostics technologies.

1.3 EVOLVABLE MODELS

Real-world engineering and manufacturing problems come with high levels of complexity and complex underlying dynamics requiring sophisticated methods and tools for building intelligent systems (ISs) that can learn and adapt online [KS02]. These types of systems are capable of increasing structural complexity, perform model refinement and update its knowledge base through interaction with the environment with learning mechanism that may be supervisory or non-supervisory by design [AK98 and YM98]. Clustering analysis, a discipline related to data-mining (focus on knowledge discovery) and machine learning (focus on prediction & incremental learning), can form a building block for such systems. Recently, there has been a renewed focus on online clustering methods applied in various problem domains including image processing, classification and system identification. Classical clustering methods include Fuzzy C-Means (FCM) [BE84], K-Nearest Neighbor (KNN) [A92], Mountain Clustering [YF94] and its variant the Subtractive Clustering algorithm assume the availability of all data, operate in batch mode and create crisp or fuzzy spheroid shaped clusters. Online clustering methods have been proposed for performing clustering tasks dealing with streaming type of data [FG10, SR12, TK12, DS11, AM11] with single pass operations.

The Gustafson-Kessel (GK) algorithm characterizes each cluster with a centroid and a covariance matrix as an extension to the original Fuzzy C-mean clustering method [GK]. GK employs the inner product distance norm formally known as the Mahalanobis distance [MD], which is calculated from a positive definite symmetric matrix A:

$d_{ik}^2 = \|x_k - v_i\|_{s_i}^2 = (x_k - v_i)^T A_i (x_k - v_i)$, the norm inducing matrix *Ai* is calculated from p_i, the cluster volume, and F_i, the fuzzy covariance matrix of the *i*th cluster and n is the dimensionality of the space where clustering takes place:

$$A_i = [p_i \det(F_i)]^{1/n} F_i^{-1} \quad (1)$$

By doing so, GK algorithm rids of the restrictions being imposed on other clustering methods by allowing the creation of ellipsoidal shaped clusters with varying orientations and sizes and is relatively insensitive to data measurement scales. Extended versions of GK capable to dealing with data streams with single pass operation have been proposed by [FG10, GF09, DS11, SR12]. In [AF04], online data clustering mechanism was applied to identify Takagi-Sugeno (TS) type models consist of If-Then rules as its knowledgebase. Evolving Takagi-Sugeno (ETS) model is capable of utilizing its evolvable multi-model structure and parameters to decompose a non-linear system into simpler ones. Effectively, for a complex problem the clustering procedure acts as a mean to partition the whole operational space into crisp (non-overlapping) or fuzzy (allowing some overlapping) into smaller sets while a local (cluster specific) model is responsible to capture the dynamics within the proximity of the corresponding cluster. Various applications including diagnostics and prognostics, motion tracking, non-linear system identification and controls have been discussed in literature [FCT10, LC13, IA12, MB15].

1.4 PROBLEM STATEMENT

The development of generic data-driven diagnostic and prognostic methods with evolvable structure and parameters will be the focus of this research. Specifically, system prognostics functionalities including degradation and RUL estimation will be the focal point and the main exemplary application. Built upon a mutual information based distance measure applied to the extended version of the Gustafson-Kessel-like (MIRGKL) algorithm, a new evolving clustering method with built-in pruning mechanism and embedded local models will be proposed and applied to problems of different natures.

Sensor selection, feature extraction, data fusion and physics based or historical data based analysis, while all important aspects of the development of a capable monitoring system, they are beyond the scope of this research. In addition, there is no strict assumption as to the underlying distribution of the data except for the assumption of Gaussian. In other words, the focus of the

research will be mainly on the evolving data driven model and less so on the derivation of physical model based information extraction and associated signal processing procedures.

The research will conclude with reports based on proposed MIRGKL applied to degradation monitoring (with local adaptive models) and data compression (without local adaptive models). Proposed models will be validated by real-world data collected under different experimental scenarios and goals. For process degradation and RUL estimation, we will specifically address issues when the degradation or RUL signal is available less frequently when compared to other information channels.

1.5 ORGANIZATION OF THE DISSERTATION

Existing literature on evolvable models and fault and degradation monitoring is reviewed in Chapter 2. In Chapter 3, proposed evolving clustering method, Mutual Information based Extended version of Gustafson-Kessel-like Clustering (MIRGKL) will be discussed with a validation on GPS data. Chapter 4 is dedicated to MIRGKL with evolvable local models and is validated with lab collected (drill bit) data and field collected (brake wear) data. Both of the datasets we studied are with limited observability in terms of system degradation requiring special treatment during model training (inference through normalization and interpolation). Chapter 5 is the final chapter where conclusions, comments and directions of future work will be given.

Evolving clustering methods have advantage to deal with stream type of data when compared to traditional clustering methods that requires all data to be available at the time of operation. Gustafson-Kessel algorithm uses Mahalanobis distance measure where orientation and size of a cluster becomes adaptable due to its clusters being defined as sets of centroids and covariance matrices. Jensen-Shannon (JS) divergence, a branch of the more generic Bregman divergence, is dependent on the choice of a convex function and can lead to different divergence and distance measures. Examples include the squared Euclidean distance (ED), Mahalanobis distance (MD) and the well-known Kullback–Leibler divergence (KL) just to name a few. The novelty in this part of the research presented in Chapter 3 is the use of the mutual information

formulation originated from JS to MD where the similarity or divergence between two distributions is measured as a weighted quantity obtained through the creation and use of a mixture of distributions obtained from known ones. In addition, a pruning mechanism is added as a way to regulate the growth of the overall structural complexity. Without any local model attached to each cluster, we demonstrate the proposed new clustering algorithm and benchmarked it against other variants of the GK algorithm utilizing real-world GPS data. This part of the work effectively utilized the proposed MIRGKL as an engine for data compression and favorable results are obtained.

In Chapter 4, MIRGKL is extended with cluster specific local models (MIRGKL-plus) and such a design is inspired by the pioneering work on online identification and incremental refinement of Takagi-Sugeno rule base [YF94, AF04 and L08]. General difficulties include not being able to handle non-stationary processes, requiring assumptions regarding underlying data disruptions, assumption of data independence and lack of online incremental changes to the modeling mechanism are common in equipment and system prognostics literature. In addition, costly and infrequent observations for RUL or degradation signal is often assumed to be a non-issue where in the real-world remains to be one of the most daunting obstacles during the development of a prognostics model. By design, proposed model's supervisory and unsupervisory learning mechanisms are decoupled, and as a result, alleviates many of the difficulties mentioned above where degradation signal may come much less frequently when compared to other data channels. MIRGKL with local models will be validated with lab collected drill bit data with no degradation measurement available in this chapter. Novelty detection is a broader definition of the process monitoring problem, and hence, will be used frequently in this chapter. In Chapter 5, we will discuss novelty detection based on data evolution characterized by the evolving cluster process. We will summarize some notions discussed in our published papers and extend the framework with newly proposed concepts.

Conclusions, comments and directions of future work will be given in Chapter 6, which is the last chapter of the dissertation.

CHAPTER 2: LITERATURE REVIEW

This chapter discusses previous work on evolvable models including evolving clustering and equipment prognostics in sections 2.1 and 2.2, respectively.

2.1 EVOLVABLE MODELS

An evolvable model (EM) performs incremental construction of an increasingly more sophisticated model structure and adaptation of relevant parameters as part of a growing knowledge base. Different from most adaptive methodologies that only adapt model parameters without modification to the overall model structure, EM typically operates through an optional unsupervised data clustering algorithm where partitioning of the overall operational space is carried. A partition of the operational space may be represented in the form of centroid and influencing radius pairs that may take the form of cluster center (focal point) and radius (radius of influence of a local model) pairs, cluster centers with the same fixed radius or clusters and covariance pairs where each cluster's size and orientation may differ. In the case where the partition is set to be fixed, it is equivalent of multiple local models being activated based on different inputs and only local models' parameters will be evolving through interaction with data. This type of model is attractive in practice in following situations: 1) System is not yet or too costly and time consuming to be fully characterized. 2) System with multiple operating modes. 3) Systems dynamics will be non-stationary during its interaction with the environment and through its usage life. 4) Piece to piece variations of the same system will be significant due to natural variations in combination with the effects as described in 2 and 3. It should be noted that piece to piece variations may be caused not only by a production process but driven by the application's intended use such as a software based personal digital assistant that need to adapt and learn from different users that will exhibit unique usage patterns and usage conditions.

The Takagi-Sugeno (TS) type fuzzy model is a powerful yet practical method for model and control of complex systems. It uses the notion of linearization in fuzzily defined regions in the state

space which effectively decomposes the original nonlinear model into multiple linear ones covered by fuzzy partitions [YF94]. The original fuzzy inference system makes embedding human or expert knowledge to the initial knowledge base possible; learning of TS model from data represents the idea of structure and parameter identification through: 1. Incremental adaptation of existing focal points and their parameters. 2. Addition of new focal points and associated rule bases and 3. The consolidation of focal points that are deemed too similar [AF04]. In [F01], the principle of evolving models is applied to a process control problem where a rule base of models (model bank) was proposed. The model bank learning algorithm initially started with randomly initialized clusters. When a normalized measure for the winning adaptive model meet the update criteria $\frac{\|\theta_j(t)-\theta_k\|}{\|\theta_t\|} < \varepsilon$, t denotes time, k denotes an unique existing adaptive model and $\boldsymbol{\varepsilon}$ denotes the threshold for the normalized similarity measure, the k'th model will be updated with current system observations; otherwise, a new adaptive model will be initialized because none of the existing adaptive models share enough similarity with the current system observables that warrants the assignment of current data to an existing and subsequent use it as evidence to adapt the model. While this method takes the "winner takes all" approach when it comes to model adaptation and utilization, control actions governed by a committee of (top) models utilize normalized similarities as weights is possible.

Another type of evolving system utilizes predetermined state space partitions or granulation where local models are subject to adaptation during its learning phase. In the estimation or prediction phase, various granulation approaches (crisp or fuzzy) may be applied to further generalize the performance of the model where multiple local models may have their outputs taken into account in the final prediction. Markov models can be applied for control purposes where forecasting mechanism include projection of future states and use of corresponding local adaptive models. In [FT11], the state space is spanned by time of day and day of the week with local adaptive Markov model for driver destination predictions. The granulation of the state space

was designed to capture prototypical decision models in terms of location transitions. It should be note that different encoding methods such as fuzzy encoding may be applicable to not only generalize, smooth out model outcome it is beneficial to adopt such information granulation mechanism to overcome lack of training of certain local models in the initial phase of deployment of the model. Since the underlying learning mechanism is equivalent of imposing a moving window with a set of recursive procedures, the outcome is a "*LIVING*" database that contextually matches the user's recent decisions for location transitions. Another aspect of applying information encoding with multiple local models is the improved consistency, usability and predictability of the outcomes of the model which is a crucial aspect for the design of systems that will have certain level of interactions with human.

Except for the case when the space granulation or partitioning has been pre-determined, at the center of an evolving model is typically to some extent a clustering method [MG14, LL01, ZD14]. The reason being that the design of the mechanism the enables self-adaptation shall be effective in its predictive performance yet sufficiently compact in its overall structural complexity is essentially the same task a clustering method performs.

2.1.1. Evolving Clustering

Clustering, synonymous to cluster analysis, classification, numerical taxonomy and typological analysis represents the task of grouping object (with descriptive data) such that object in the same group share more similarity when compared to objects in other groups [XW05]. Any clustering method is either supervised or unsupervised depending on the availability of the truth labels that consist of finite number of discrete assignable classes given some inputs [B95]. Most well-known clustering methods include Fuzzy C-Means (FCM) [BE84], K-Nearest Neighbor (KNN) [A92], Mountain Clustering [YF94], Support Vector Machine (SVM) [CV95] and Relevance Vector Machine (RVM) [T01]. Most clustering algorithms assume the availability of all data, operate in batch mode and create crisp or fuzzy spheroid shaped clusters. They have been applied in extensively in exploratory data mining and visualization, object detection, novelty detection,

knowledge discovery and acquisition, statistical data analysis, machine learning, pattern recognition and data representation and compression.

Online clustering methods have been proposed in [FG10, SR12, TK12, TS11, and AM11] with single pass operations where proposed models are evolvable both structurally and parametrically. Compared with batch type methods, they are specifically suitable for problems with following characteristics:

A. *Stream Type of Data and Very Large Data (VLD)*: Mining and extraction of knowledge from existing dataset that is extremely large and cumulative data streams that, over time, grows significantly in its sheer size poses great challenging to most data modeling methods [GZ09]. For example, the number searches Google performs increased from 60 million during 2000 to 5.7 billion during 2014 on a daily basis. In modern vehicles, there are thousands of signals in its CAN (controller area network) updated at a speed at milliseconds. Even with down sampling, the amount of data grows very rapidly not only for storage but also for the identification of applicable means to analyze them. Tackling the task of modeling and extracting useful information from stream type of data or extremely large data requires a fundamental shift in the frame of mind of data miners as well as development of methodologies to be more applicable in these situations [DH01].

B. *Dynamic Operating Conditions:* A common challenge for real-life applications of clustering algorithm has to be with non-stationary data. Facial and voice recognition, metabolic pathways in biological cells, object tracking in computation vision and monitoring of equipment operating under distinct modes under the influence of random noises are just few examples [LA14, FC04a and FC04b]. Clustering methods operate in batch mode, assuming all data to be observable during structure and parameter identification phase, often requires the model to be re-trained if certain limits on the range input space have been exceeded rendering existing model to be invalid or facing

the risk of significantly degraded performance. On the other hand, evolvable clustering method are designed to take into account such conditions by evaluate the degree of deviation at both the overall model level as well as individual cluster level. Dependent on the level of deviation, either incremental adjustment can be made to existing clusters or a new cluster will be created to specifically account for new and emerging patterns.

C. *Uncertainty in Acquired Information:* The ability to make subtle adjustment to existing clusters to deal with signals embedded with varying level of drifts imposed by either the physical attributes of the underlying sensing hardware (i.e. GPS signals) along with varying noises types and impacts they have on the fidelity of the signal when interacting with the environment [FT11]. In some applications, this involves piece to piece variations as well as similarly purposed equipment equipped with different classes of sensors due to hardware advancement and for the purpose of differentiation in their portfolio accommodating different target industries and customers.

D. *Building of Intelligent Systems:* An intelligent system improves system performance through learning while preserving useful previously gained experience. It is a highly interdisciplinary field involves control theory, artificial and computational intelligence, biologically and life science inspired systems and various engineering disciplines [A12 and A13]. These systems rely on statistical and data driven mechanisms to adapt their flexible yet evolvable model structure and parameters temporally to accommodate both abrupt and slow changing dynamics [L08 and L15]. At their cores, these algorithms represent the development of goal-oriented self-learning machines engaging in extraction of new knowledge and conduct evidence based refinement to existing ones when opportunities present themselves.

Participatory learning (PL), based on unsupervised learning, provides a mean to identify structures of a rule-based system. It enables the learning process through compatibility evaluations between currently known belief and observations made in the learning environment

[R90]. Naturally, several unsupervised dynamic clustering algorithms have been proposed as part of a wave of adaptive systems [LG06, AF05 and OL04]. The original Evolving Takagi-Sugeno (ETS) model and its modified version Simpl_eTS model combine the supervisory and unsupervisory learning to dynamically identify new rule bases and modify existing ones. The unsupervisory part of learning mainly addressed the core issue of dividing the state space into sub-partitions (through recursive clustering) and the supervisory learning procedure establishes a local linear model for each sub-partition (cluster specific) [AF04 and AF05] for approximations of an overall non-linear system's behavior.

One critical element in any clustering method is the distance (similarity) measure employed to determine: 1) Similarity between a new data to existing clusters for data assignment and cluster update purposes and 2) Similarity between existing clusters to evaluate the necessity of structure pruning. In addition, the choice of the distance measure will have a direct relationship as to how each identified cluster will be characterized and subsequent procedures for adaptation according to new evidence [AG01, BB91, CG91, HB00, M67, FT11 and MJ94]. The Mahalanobis distance is an unit-less and scale invariant distance measure generalized for the multi-dimensional space and enables the creation of ellipsoidal shaped clusters with variable orientation and sizes [M36].

$$D_M(X) = \sqrt{(x-\mu)^T S^{-1}(x-\mu)} \quad (2)$$

x: data point to be evaluated

u: cluster specific mean vector

S: cluster specific covariance matrix

The recursive procedures to identify clusters using the Mahalanobis distance is presented in [FT11] where cluster specific local models were trained in a least square fashion and the evolution of the clustering process is tracked as serve as indicators for incipient and abrupt failures of a machine.

2.1.1. Gustafson-Kessel Clustering Algorithm and Bregman Divergence

The Gustafson-Kessel (GK) algorithm characterizes each cluster with a centroid and a covariance matrix and employs the inner product distance norm formally known as the Mahalanobis distance [MD] which is calculated from a positive definite symmetric matrix A:

$d_{ik}^2 = \|x_k - v_i\|_{s_i}^2 = (x_k - v_i)^T A_i (x_k - v_i)$, the norm inducing matrix A_i is calculated from p_i, the cluster volume, and F_i, the fuzzy covariance matrix of the ith cluster and n is the dimensionality of the space in which the clustering procedure takes place:

$$A_i = [p_i \det(F_i)]^{1/n} F_i^{-1}$$

Extended version of GK capable to dealing with data streams with single pass operation has been proposed by [FG10, GF09, DS11, SR12]. It is worth to note that the single pass operation proposed in these methods, in contrast of iterative batch procedures, represents significant improvement in computational resources and memory requirement when dealing with very large or stream type of data where the size of the whole dataset may be undefined initially.

Bregman divergence [B67] represents divergences generated from some convex function. Examples include well-known Euclidean distance, Itakura-Saito distance, Kullbak-Leibler (KL) divergence and the Mahalanobis distance [BM05b]. A few examples of convex function of choice $\Phi(X)$ and corresponding distance measure $d_\Phi(X,Y)$ derived from definition of Bregman divergence are shown in Table 1 as follows:

Table 1:*Bregman divergences generated from convex functions [BM05b]*

$\Phi(X)$	**$d_\Phi(X,Y)$**	**Divergence**
x^2	$(x-y)^2$	Squared loss
-logx	x/y-log(x/y)-1	Itakura-Saito distance
$\|\|x\|\|^2$	$\|\|x-y\|\|^2$	Squared Euclidean distance
$x^T Ax$	$(x-y)^T A(x-y)$	Mahalanobis distance

$\sum_{j=1}^{d} x_j log_2 x_j$	$\sum_{j=1}^{d} x_j log_2 (\frac{x_j}{y_j})$	Kullbak-Leibler (KL) divergence

It was shown in [BM05a] that clustering problems can be solved using Kmeans type of iterative schemes [HW79 and WC01] only if the underlying distortion applied as a distance measure is from the family of Bregman divergence. Furthermore, it was also proven that the minimization between data points to identified cluster centroids is equivalent to the loss in Bregman information therefore the minimizing either produced the same optimal clustering result [BM05b]. Jenson-Shannon divergence (JSD), a variant of KL divergence provides an inspiring formulation that is applicable to other Bregman family of divergences that yields a smoothed, symmetrized and bounded form when compared to its original forms [L91].

2.2 EQUIPMENT PROGNOSTICS

Diagnostics tasks are defined as the ability to detect, localize and determine the severity of a fault in a system that is “currently” in a failure state; prognostics tasks are often much more difficult to be carried out in practice because the need to perform forecasting into the future. As a result, it is often referred to as the Achilles’ heel of CBM systems. Prognostics research historically has much less available literature when compared to diagnostics methods [VWK99] due to difficulties surrounding elements to be forecasted are relevant to a stochastic process and specifically the availability of usable features and enabling techniques to support such development of prognostics functionalities could be difficult to obtain [BM99]. So far, center of the attention of such efforts often focus on systems with slow and progressive degradations pattern leading to eventual failure including mechanical and structural systems [Mar01]. While prognostics often built upon diagnostics functionalities, it has a stronger emphasis on predicting the Remaining Useful Life (RUL) [WC01] and the progression of a diagnosable failure [F05] through prognosis assessment. In the literature, statistical methods (i.e. Autoregressive Moving Average or ARMA model), stochastic methods (i.e. Hidden Markov Model or HMM) [KT08] and

machine learning methods (i.e. Artificial Neural Network or ANN, Support Vector Machines (SVM) and Relevance Vector Machines (RVM)) [WC01, CW10 and HL07] all have been applied as means to perform prognostics tasks. One should not confuse the RUL estimation with the notion of “Mean Time to Failure” (MTTF), the prior (RUL) has to do with a specific machine or system’s predicted remaining life with respect to time or other cumulative usage unit while the latter is more relevant to a “population wise” attribute in an average sense [EGBH00].

Aside from data-driven methods mentioned in the previous paragraph, physics or model-based methods have been developed for various types of equipment with much success. However, these types of efforts are mostly reserved for mission critical systems and equipment due to extensive efforts and dedicated experiments required to establish the model [BM99, KRM02]. Application of the same model to similarly purposed, updated equipment with design differences or equipment under completely different usage patterns in most cases are not feasible. In those cases, development and calibration procedure typically need to be repeated to be able to achieve good performance in the field [RK00, TV01 and HH01]. In recent Condition Based Monitoring (CBM) literature, data-driven prognostics started to utilize methods developed for survival analysis and specifically the reliability function such as Weibull distribution [GH09].

2.1.1. Data-Driven Methods for Prognostics

Traditional reliability models such as Weibull, Poisson and Log-Normal distribution based estimation methods characterizes population performance degradation patterns to enable longer-term forecast of RUL [SR03 and G00]. However, this type of model often provides predicted RUL information not accurate enough for individual units [HZ09]. As a result, Machine Learning (ML) based method and hybrid models utilizing ML techniques have been proposed in [CC10, FT08, B14, J07, JR06, SG09, DT12 and NA14] that can deal with stochastic degradation signatures and operation environment, make few to no assumption as the underlying signal distributions yet integratable with physical understanding of a system and some of them with online adaptable model structures and parameters [FT06a and FT06b]. In [CC05 and KT08], Hidden Markov Model

applied through utilizing a hierarchical structure and as a mean for sequential clustering for the purpose of drill bit health diagnostics. Though RUL information was not explicitly expressed, they indirectly obtained equipment RUL information in the form of cumulative equipment usage as total number of tasks conducted so far through knowledge of each drill bit started to be brand new and End of Life (EOL) was reached until each of them can no longer carry out the prescribed drilling tasks. They represent a case where indirect inference of RUL information can be inferred for the purpose of diagnostic model but could also be applicable for the development of prognostics models.

2.1.2. Usage of Prognostics based Information

In contrary to the vehicle industry, the aircraft and military industry have been moving at a much faster pace to apply Integrate Vehicle Health Maintenance systems (IVHM) or Prognostics and Health Management (PHM) system to fully utilize diagnostics and prognostics information produced by such systems. To move away from the of being sole manufacturers, IVHM has become highly desirable as a way for OEM's to enhance maintenance programs, improve logistics efficiency and being able to offer services at a more customized level to their customers[EJJ13]. Due to the effectiveness of IVHM based maintenance strategies, the demand has been substantial from the airplane operators to retrofit their legacy fleet (that do not come with necessary hardware) with OEM IVHM hardware to take advantage of the added benefits of advanced diagnostics and prognostics technologies. With various industries' renewed focus on reducing downtime and critical failures [KT06], maintaining system's operational safety and ongoing maintainability [SZ10] and to able to address reliability issues arise from more complex design and ever more demanding operation conditions during lifecycle of a produce, PHM and IVHM systems with integrated prognostics functions will play a crucial role in effective fulfillment of Produce-service system (PSS) contracts [DA11 and LT12]

CHAPTER 3: MUTUAL INFORMATION BASED RECURSIVE GUSTAFSON-KESSEL-LIKE CLUSTERING ALGORITHM (MIRGKL)

The notion of a “cluster” cannot be precisely because its meaning and property varies depending on the applications or algorithms [E02]. Nevertheless, clustering is generally synonymous to cluster analysis, classification, numerical taxonomy and typological analysis and represents the task of grouping object (with descriptive data) in a manner such that object in the same group share more similarity when compared to objects in other groups [Jain 1999 and XW05]. Different from classification, which is typically a supervised task where some predefined labels consist of finite number of distinct classes is available; clustering is mostly an unsupervised task where there exists no true label for each piece of information. Over the years, methods include Fuzzy C-Means (FCM) [BE84], K-Nearest Neighbor (KNN) [A92], Mountain Clustering [YF94] and their early variants operate in batch mode and create crisp or fuzzy spheroid shaped clusters with some predefined parameters including cluster radius or number of clusters to be identified. Originally proposed and applied for anthropology [DK32] and psychology [T39] research, clustering algorithms been applied extensively in exploratory data mining and visualization, object detection, novelty detection, knowledge discovery and acquisition, statistical data analysis, machine learning, pattern recognition and data representation and compression.

Online clustering methods have been proposed for performing the same clustering tasks dealing with stream type of data or very large data (VLD) [FG10, SR12, TK12, TS11, AM11] with single pass operations. These online evolvable clustering algorithms allow incremental changes to be made both structurally and parametrically through respective data-driven mechanisms. More recently, online clustering methods have been heavily researched and applications in objective detection, diagnostics and prognostics and system identification due to their much improved capability through embedded adaptable model or function specific to each identified cluster [AF10]. In [L08], development has been made where prediction performance of such model is comparable to a physics based model for a rather complex system.

Gustafson-Kessel (GK) clustering characterizes a cluster with a centroid and a covariance matrix and represents a major improvement over traditional methods with its ability to better adapt to real-world data that comes in different shapes and forms [GK78]. The distance measure GK employs what is formally known as the Mahalanobis distance. Alternatively, it can be expressed as the following:

$$d_{ik}^2 = \|x_k - v_i\|_{s_i}^2 = (x_k - v_i)^T A_i (x_k - v_i) \quad (3)$$

The norm inducing matrix A_i is calculated from p_i, the cluster volume, and F_i, the fuzzy covariance matrix of the *i*th cluster and n is the dimensionality of the space in which the clustering procedure takes place:

$$A_i = [p_i \det(F_i)]^{1/n} F_i^{-1}$$

It has been shown in that clustering problems can be solved using K-means type of iterative schemes [HW79 and WC01] only if the underlying distances measure are from the family of Bregman divergence [BM05a]. Furthermore, it was also proven that the minimization between data points to identified cluster centroids is equivalent to the loss in Bregman information therefore the minimizing either produced the same optimal clustering result [BM05b]. Bregman divergence [B67] represents divergences generated from some convex functions and examples include well-known Euclidean distance, Itakura-Saito distance, Kullback-Leibler (KL) divergence and the Mahalanobis distance [BM05b]. Jenson-Shannon divergence (JSD), a variant of KL divergence provides an inspiring formulation that is applicable to other Bregman family of divergences that yields a smoothed, symmetrized and bounded properties when compared to its originating divergence the KL divergence [L91].

Extended version of GK operates recursively (single pass operation) has been proposed by [FG10, GF09, DS11, SR12]. Single pass operation proposed in these methods, in contrast of iterative batch procedures, represents significant improvement in the reduction of computational resources and memory requirement. Similar to other online clustering methods such as [QW08,

TK12, BL05, BZ06 and AF04] they are designed to deal with very large data (VLD) or stream type of data where the size of the whole dataset may be very large or initially undefined. Online clustering methods are critical in the development of intelligent systems that are goal-oriented self-learning machines engaging in extraction of new knowledge and conduct evidence based refinement to existing ones when opportunities present themselves [KS02, A12 and A13].

This article proposes a novel Mutual Information based Recursive Gustafson-Kessel Clustering Algorithm (MIRGK). We will briefly describe the original version of the extended Gustafson-Kessel algorithm in section 2, describe in detail the proposed mutual information based version in section 3, Experiment and validation using real-world GPS data will be given in section 4, and finally we will conclude the paper with summary and future work in section 5.

3.1 EXTENDED VERSION OF THE GUSTAFSON-KESSEL (GK) ALGORITHM

In real-life applications, we are often encountered with data exhibiting varying non-stationary behaviors, generated from noisy operating conditions and tend to grow in its amount as usage being accrued [SR12]. While it is not uncommon to restart the clustering algorithm by using only a fraction of data, it is not a very efficient way of carrying out such task and by doing so also defeat the purpose of the intention of extracting useful information from all the data. As a result, online clustering is of particular interest recently in fields such as pattern recognition, machine learning and non-linear system identification [DH12]. The extended version of Gustafson-Kessel algorithm [FG10 and DS11] operates recursively to update cluster specific parameters including the centroid, the inverse covariance matrix and the determinant of the original covariance matrix. In [FG10], a detailed and more generic version of the updating procedure based on Exponentially Weighted Moving Average (EWMA) was derived:

$$V_{i,t} = (1-\alpha) * V_{i,t-1} + \alpha * X_t = V_{i,t} + (1-\alpha) * (X_t - V_{i,t-1}) \quad (4)$$

$$F_{i,t}^{-1} = (I - G_i * (X_t - V_{i,t-1})) * F_{i,t-1}^{-1} * \frac{1}{1-\alpha} \quad (5)$$

where

$$G_i = F_{i,t-1}^{-1} * (X_t - V_{i,t-1})^T * \frac{\alpha}{1-\alpha+\alpha*(X_t-V_{i,t-1})*F_{i,t-1}^{-1}*(X_t-V_{i,t-1})^T} \quad (6)$$

$$detF_{i,t} = (1-\alpha)^{n-1} * F_{i,t-1} * (1 - \alpha + \alpha * (X_t - V_{i,t-1}) * F_{i,t-1}^{-1} * (X_t - V_{i,t-1})^T) \quad (7)$$

It should be noted that i denotes the i'th cluster, t denotes time instant, *V*, *F* and *detF* are cluster specific centroid, covariance matrix and determinant of the covariance matrix. The procedure employs *Sherman-Morrison-Woodbury* or simply *Woodbury formula* [W50] to derive the recursion of the inverse covariance matrix. Additional cluster specific credibility value M increments whenever a data falls into the coverage of a cluster. This parameter functions as a relaxation mechanism in the cluster algorithm's data assignment and cluster update procedure. Effectively, this allows additional incremental adjustment to clusters that have been receiving little evidence for subsequent refinement since creation and increase the coverage of all available space occupied by all clusters to be maximized. [FG10] also suggested the use of negative updates to non-receiving clusters to mitigate issues with overlapping clusters common in GK derived clustering algorithms.

The application of (4-7) supports at least 3 flavors of online versions of GK derived clustering algorithms including the recursive evolving version GK(eGK), evolving GK-Like (eGKL) algorithm [FG10]. While the main elements are quite similar in notion, eGK and eGKL algorithm are different in a few key steps. First of all, eGKL may start from scratch with either some clusters already been identified (either from a initialization step or from the previous execution) while eGK relies on the existence of some clusters. In addition, the credibility of an existing cluster in eGK is estimated at each step to determine enclosure of points while in eGKL this is done through increment a cluster specific value in the data assignment process as a result a more relaxed and computationally less taxing procedure. While sharing the same updating procedure of the winning cluster (4-7), eGKL performs an additional update to the centroids of non-winning clusters adopting a negative learning rate to move them slightly away from the current data. Lastly, the distance or similarity measure in eGK is the original Gutafson-Kessel formulation where the norm

induction includes a fuzzification term $det(F)^{1/n}$; while in the eGKL the provision was made to have the flexibility of choosing between some unit-less measures including the Gustafson-Kessel norm inducing formulation or the Mahalanobis distance. The main steps of eGK and eGKL clustering algorithms are organized side by side and summarize the following table.

Table 2: Comparison of *Key Steps in eGK v.s. eGKL Clustering Algorithm*

Clustering Steps	Evolving GK Algorithm	Evolving GK-Like Algorithm
1. Initialization	Batch GK or other clustering algorithms	Optional; may start from scratch
2. Distance / Similarity Measure	$d_{i,k} = \sqrt{(X_k - V_i)[(\det(F_i))^{1/n} F_i^{-1}](X_k - V_i)^T}$	$D_{i,k}^2 = (X_k - V_i)F_i^{-1}(X_k - V_i)^T$ or $d_{i,k}$
3. Cluster Assignment Criteria	$d_{p,k} \leq r_p$	$D_{p,k}^2 < \mathcal{X}_{n,\beta}^2$ *or* $M_p < M_{min}$
4. Winning Cluster Updates	(4 - 7)	(4 – 7)
5. Non-Winning Cluster Update	N/A	Centroids moving away from sample with a negative learning rate

Evidential evolving GK (E2GK) [SR12] was inspired by eGK and Evidential Fuzzy C-means Algorithm (ECM) [MD08]. In [SR12], E2GK creates credal partitions (similar in concept to fuzzy but more genera) using belief function theory and adopts the same cluster adaptations using (4 - 7).

3.2 Mutual Information based Recursive Gustafson-Kessel Clustering Algorithm

Post cluster assignment, eGK and eGKL maintain cluster credibility parameters that are calculated differently but both serve as a mean to control the total number of clusters being created. In eGK, such parameter for a new cluster is assessed to determine if a meaningful amount of number of existing data points are within its coverage. Failure to meet such criteria results in removal of the newly added cluster and the overall cluster reverted to the one prior to processing the current data point. On the other hand, in eGKL the credibility parameter provides opportunistic additional updates to clusters that are not yet considered to be well defined. It is

worth noting in eGKL the provision of having “infant” or “emerging” stage clusters’ absorption of new data fall into a proximity region reduces the propensity in eGK where non-consecutive yet similar patterns may not be properly captured.

The structure complexity regulation mechanisms described above focus on evaluation of new or relative new clusters’ coverage to known data points by calculations of cluster specific credibility scores. Another avenue to ensure conditions led to creation of a new cluster is truly warranted is study the more fundamental aspects of a clustering algorithm which is the similarity measure of choice.

3.2.1 Mutual Information based Mahalanobis Distance

In [D07], the well-known Kullback-Leibler (KL) divergence for two distributions $\aleph_0$ and $\aleph_1$with mean,μ_0 and μ_1, and covariance matrices, $\sum_0$ and $\sum_1$, of the same dimension is expressed in the following matrix form:

$$D_{KL}(\aleph_0||\aleph_1) = \frac{1}{2}(tr(\sum_1^{-1}\sum_0) + (\mu_1 - \mu_0)^T\sum_1^{-1}(\mu_1 - \mu_0) - k + \ln(\frac{det\sum_1}{det\sum_0})) \quad (8)$$

The terms calculating the trace of the ratio between $\sum_0$ and $\sum_1$ and the natural log of the ratio between the determinant of $\sum_0$ and $\sum_1$ are related to the comparison of the size and shape of the distributions as defined in their respective covariance matrices. The remaining terms include the dimensionality expressed in k and $(\mu_1 - \mu_0)^T\sum_1^{-1}(\mu_1 - \mu_0)$ which is nothing but the original definition of the Mahalanobis Distance [M36]. As a result, one can consider the Mahalanobis distance as a special case for the more generic definition as defined by the KL-divergence.

Bregman divergence represents a more generic divergences generated from some convex functions [B67]. It was shown in [BM05a] that clustering problems can be solved using K-means type of iterative schemes [HW79 and WC01] only if the underlying distortion applied as a distance measure is from the family of Bregman divergence. Examples of divergences derived from the Bregman divergence, not surprisingly, include well-known Euclidean distance, Itakura-Saito

distance, Kullback-Leibler (KL) divergence and the Mahalanobis distance [BM05b]. A few examples of convex function of choice $\Phi(X)$ and corresponding distance measure $d_\Phi(X, Y)$ derived from definition of Bregman divergence are shown in Table 1 as follows:

Table 3: *Bregman divergences generated from convex functions [BM05b]*

$\Phi(X)$	**$d_\Phi(X,Y)$**	**Divergence**
x^2	$(x-y)^2$	Squared loss
-logx	x/y-log(x/y)-1	Itakura-Saito distance
$\|\|x\|\|^2$	$\|\|x-y\|\|^2$	Squared Euclidean distance
$x^T Ax$	$(x-y)^T A(x-y)$	Mahalanobis distance
$\sum_{j=1}^{d} x_j log_2 x_j$	$\sum_{j=1}^{d} x_j log_2(\frac{x_j}{y_j})$	Kullbak-Leibler (KL) divergence

Furthermore, it was also proven that the minimization between data points to identified cluster centroids is equivalent to the loss in Bregman information therefore the minimizing either produced the same optimal clustering result [BM05b]. Jenson-Shannon divergence (JSD), a variant of KL-divergence provides an inspiring formulation that is applicable to other Bregman family of divergences that yields a smoothed, symmetrized and bounded form when compared to its original forms [L91] defined as the following:

$$D_{JS}(\aleph_0||\aleph_1) = \frac{1}{2}D_{KL}(\aleph_0||\aleph_M) + \frac{1}{2}D_{KL}(\aleph_1||\aleph_M) \quad (9)$$

Where $\aleph_M = \frac{1}{2}\aleph_0 + \frac{1}{2}\aleph_1$

The Jensen-Shannon divergence is the mutual information between a random variable to a mixture distribution N_M between the two distributions of interest, namely N_0 and N_1. D_{KL} as shown in (9) represents the KL-divergence value between two distributions and $D_{KL}(N_0||N_1)$ is different from $D_{KL}(N_1||N_0)$. A special case for (9) where the shape and size information contained in the

covariance matrix not to be taken into account separately other than as a normalizer would become the following:

$$D_{MM}(\aleph_0||\aleph_1) = \frac{1}{2}D_M(\aleph_0||\aleph_M) + \frac{1}{2}D_M(\aleph_1||\aleph_M) \quad (10)$$

Where $\aleph_M = \frac{1}{2}\aleph_0 + \frac{1}{2}\aleph_1$

And D_M is the original Mahalanobis Distance:

$$D_M(\aleph_0||\aleph_M) = \sqrt{(\mu_0 - \mu_M)^T \sum_M^{-1} (\mu_0 - \mu_M)} \quad (11)$$

In essence, the formulation presented by the Johan Jensen and Claude Shannon as shown in [L91] applied the original KL-divergence calculations from the perspective of a mixture distribution denoted by N_M against the two distributions of interest, N_0 and N_1. As a result, the KL-divergence with the Jensen-Shannon formulation can be understood as the weighted average of N_i with respect to N_M where Ni are the distributions that their difference need to be quantifiably measured and N_M is a linear combination of N_i. The fact that the Jensen-Shannon formulation transformed the originally unbounded KL-divergence to be bounded and symmetrized the originally asymmetric similarity measure brings in significant improvement in numerical stability when utilizing such measures. Since (11) is simply partial formulation of the original (8), such property will be useful when applied in a clustering algorithm to subdue the likelihood of creation of new clusters and stabilize the cluster assignment process simultaneously.

3.2.2 Threshold Setting in Cluster Assignment, New Cluster Creation and Merging

In [FG10, FT06a and FT06b], the similarity measure of the most similar (or winning) cluster to the current data is subjected to a threshold to determine if the level of closeness warrants the assignment (of the current to the winning cluster) and subsequent updates made to its parameters:

$$D^2 < \mathcal{X}^2_{n,\beta} \quad (12)$$

This is based on the interpretation of the Mahalanobis distance as the summation of the squares of random variable where n denotes the number of random variables involved (or

dimensionality) and β is the threshold associated with a probability value. Given a problem need to be tackled by a clustering algorithm, the value of n is typically fixed and the choice of β controls the propensity of new cluster creation and the total number of clusters to be obtained. The reason for this is quite straight forward because a higher value of β will result in a larger value of $\mathcal{X}^2_{n,\beta}$ as it represents that for this particular cluster the likelihood of seeing the similarity measure exceeding $\mathcal{X}^2_{n,\beta}$ will be (1-β). In other words, the higher β (therefore a higher value of $\mathcal{X}^2_{n,\beta}$) given that dimensionality and the definition of similarity are fixed, the radius of influence of the same cluster will be increased and eventually lead to the final resulting number of clusters to be lower to represents the same set data. While we modified the similarity measure in the original eGKL to (10, 11), the Kohonen rule [K12] to update the winning cluster. When any assignment is not possible, a new cluster will be created with a default covariance matrix.

The evaluation of the merging existing clusters focusses on (C-1) pairs of clusters consists of the current active cluster and one of the remainder of the clusters where C denotes the total number of clusters. The merging of a pair of clusters, each represents different set of data points and is a result of incremental refinement based on these points, involves the combination of their respective parameters and shall be treated with more caution. Due to this reason, we propose to use (Ch-2 1) but with a different β denoted as β_M where M represents merging:

$$D^2_M < \mathcal{X}^2_{n,\beta_M} \tag{13}$$

The parameter β_M, specifically for the purpose of cluster merging, shall be a number smaller than the original β which is specifically for cluster assignment. This treatment in the evaluation of merging a pair of clusters represents a more restrictive and conservative approach as to combining two existing clusters. Another way to understand that β_M should be strictly smaller than β is as this parameter decreases; it is equivalent of imposing a tighter control limit during similarity evaluation. In other words, a random variable meeting the threshold established

by a smaller β give us a higher certainty it belongs to the same class (distribution) that it is being compared to.

Prior to merging, parameters associated with each cluster were learned separately. Alternatively, one may also determine the value of D_M^2 directly. By doing so, the probability associated with D_M^2 can still be derived with the inverse Chi-square distribution calculation.

Once it is determined that the current active cluster's closest existing cluster meet the criteria defined by (13), the centroids and covariance matrices are combined as the following:

$$\mu_M = w_a\mu_a + w_m\mu_{pm} \tag{14}$$

$$\sum_M = w_a^2\sum_a + w_m^2\sum_{pm} \tag{15}$$

Where $w_a = \frac{n_a}{n_a+n_{pm}}, w_{pm} = \frac{n_{pm}}{n_a+n_{pm}}$ and n_a and n_{pm} are the number of samples assigned to the active cluster (*a*) and the winning cluster (*pm*) in the merging process. (14, 15) using the reproductive property of normal distributions:

Assuming Y is the linear combination of independent normal random variables *Xi*:

$$Y = a_0 + a_1X_1 + a_2X_2 + \cdots a_nX_n \tag{16}$$

Y's mean, μ, and variance, σ, can be expressed as the following:

$$\mu_Y = a_0 + \sum_{i=1}^{n} a_iX_i \tag{17}$$

$$\sigma_Y^2 = \sum_{i=1}^{n} a_i^2\sigma_i^2 \tag{18}$$

Note that in (14, 15) M denotes the cluster index for the combined cluster which is the product from merging the active cluster (*a*) and the winning cluster (*pm*) during this merging evaluation process. Internally, the newly created cluster M will take the place of the currently active cluster's index and the cluster absorbed by the active cluster's index will be become available for the next newly created cluster.

3.3 SUMMARY OF MUTUAL INFORMATION BASED RECURSIVE GUSTAFSON-KESSEL-LIKE CLUSTERING ALGORITHM (MIRGKL)

Inspired eGK and eGKL clustering methods as proposed in by the research of [FG10 and DS11], we proposed the use of Mutual Information formulation applied to the Mahalanobis distance as the core similarity measure for the proposed new clustering algorithm. The treatment for determining thresholds for degree of similarity evaluation in the cluster assignment, creation and merging has been unified with inverse of Chi-square distribution with a fixed dimensionality but different levels of probability values. The major steps for the algorithm are summarized as the following:

1. Determining the thresholds $\mathcal{X}^2_{n,\beta}$ and $\mathcal{X}^2_{n,\beta_M}$, inverse of the Chi-square cumulative distribution function (CDF), that are applied during the decision making process for cluster assignment or creation and cluster merging. In the following Table 4, we show the values with fixed probability value,β, set to 0.95 and n ranges from 2 to 10. At this step, the dimensionality of the data, n, should also be specified since procedures for data pre-processing should have already been identified and that portion of the calculation is out of scope of the overall clustering algorithm.

Table 4: $\mathcal{X}^2_{n,\beta}$ Values for dimensionality 2 – 10 with β = 0.95

n	2	3	4	5	6	7	8	9	10
$\mathcal{X}^2_{n,\beta}$	5.992	7.815	9.488	11.07	12.592	14.067	15.507	16.919	18.307

2. Determine the default initial inverse covariance matrix $\sum_0^{-1}$by specifying parameter γ (i.e.: consider default value of γ = 100):

$$\sum_0^{-1} = \gamma I \tag{19}$$

Depending on whether the data is treated with a predefined normalization procedure or not, the value of γ need to be carefully chosen. If the default γ is chosen to be too small (large values in the diagonal of the original covariance matrix), the algorithm may not produce any additional clusters due to the initial coverage of a newly created

cluster (starting from the 1st cluster) will have a coverage that is overly extensive. On the other hand, if the default value of γ is chosen to be too big hence resulting a newly created cluster (starting from the 1st cluster) to be overly tight, in the worst case scenario every data point will result in a new cluster.

3. Set C = 0 and read 1st data x_1. C is the total number of clusters generated so far. Before the 1st data is process, this number should be zero.
4. Initialize the 1st cluster using the 1st data and default initial inverse covariance matrix

   $\mu_1 = x_1;\ \sum_1^{-1} = \sum_0^{-1} = \gamma I;\ C = C + 1; det\sum_1 = \det(\sum_0);\ N_1 = 1;$

   Note that N_i represents the number of data assigned to the *i*'th so far. Since the 1st data is used to initialize the 1st cluster, N_1 at this step is set to 1 as part of the initialization.
5. Read next data x_k and calculate similarity measure using Mutual Information Formulation of the Mahalanobis Distance. This is done by assigning a temporary inverse covariance matrix to the current data, $\sum_{k,temp}^{-1} = \gamma I$, to facilitate the calculations described in (MM1 and 2). In other words, given cluster i, $\aleph_i$, and the temporary cluster, $\aleph_{k,temp}$, we can compute the mutual information based Mahalanobis distance with a temporary mixture distribution $\aleph_m$. For the temporary cluster associated with x_k, the centroid is simply x_k and the inverse covariance set to γI.

   $$\mathrm{D_{i,k}}(\aleph_\mathrm{i}||\aleph_\mathrm{k}) = \frac{1}{2}\mathrm{D_M}(\aleph_\mathrm{i}||\aleph_\mathrm{M}) + \frac{1}{2}\mathrm{D_M}(\aleph_\mathrm{k}||\aleph_\mathrm{M}), \mathrm{i} = [1, \mathrm{C}]$$

   Where $\aleph_\mathrm{M} = \frac{1}{2}\aleph_\mathrm{i} + \frac{1}{2}\aleph_\mathrm{k}$

   And the formula to compute DM is the original Mahalanobis Distance:

   $$\mathrm{D_M}(\aleph_0||\aleph_\mathrm{M}) = \sqrt{(\mathrm{u}_0 - \mathrm{u}_\mathrm{M})^\mathrm{T}\sum_\mathrm{M}^{-1}(\mathrm{u}_0 - \mathrm{u}_\mathrm{M})}$$

6. Identify active (closest) cluster with respect to current data x_k

   $p = \arg\min(D_{i,k})\,, \mathrm{i} = [1, \mathrm{C}]$

7. Determine whether current data, x_k, should be assigned to the active cluster and trigger subsequent updates. Evaluation of whether the p'th cluster obtained from step warrants the assignment takes the following form:

   $\mathrm{D_{p,k}} < \mathcal{X}^2_{n,\beta}$

   7.a. Update the active cluster if the x_k is successfully assigned to p'th cluster

   The update of the p'th cluster is straightforward following (4 - 7). Go to Step 8.

   7.b. Create a new cluster if x_k is not assigned to p'th cluster

   The creation of a new cluster is identical to the description in Step 4 with the new cluster centroid set to be x_k, initial inverse covariance matrix and determinant of the covariance matrix set to default values, initial cluster size set to 1 and the increment of C to C + 1 due the newly added cluster. After this step, Step 8 will be skipped and the algorithm continues to Step 9.

8. Determine whether the active cluster shall be merged with an existing cluster. This process involves computing the similarities between the active cluster to the remainder of existing clusters.

   $\mathrm{D_{q,p}}(\aleph_\mathrm{q}||\aleph_\mathrm{p}) = \frac{1}{2}\mathrm{D_M}(\aleph_\mathrm{q}||\aleph_\mathrm{M}) + \frac{1}{2}\mathrm{D_M}(\aleph_\mathrm{p}||\aleph_\mathrm{M}), \mathrm{q} = [1, \mathrm{C}], \mathrm{q} \neq \mathrm{p}$

   Where $\aleph_\mathrm{M} = \frac{1}{2}\aleph_\mathrm{q} + \frac{1}{2}\aleph_\mathrm{p}$

   And the formula to compute DM is the original Mahalanobis Distance:

   $\mathrm{D_M}(\aleph_0||\aleph_\mathrm{M}) = \sqrt{(\mu_0 - \mu_\mathrm{M})^\mathrm{T}\sum_\mathrm{M}^{-1}(\mu_0 - \mu_\mathrm{M})}$

   The top candidate to be considered to be merged with active cluster is determined with the following criteria:

   $g = \arg\min(D_{q,p}), \mathrm{q} = [1, \mathrm{C}], \mathrm{q} \neq \mathrm{p}$

   Subsequent evaluation of whether merging shall take place by comparing the similarity value between the closest cluster *g* and the active cluster *p* with the following:

$\mathrm{D}_{\mathrm{g,p}} < \mathcal{X}^2_{n,\beta_M}$

8.a. If the above criteria is met, the active cluster p and the closest cluster g will be combined using formula (NRV 2 – 3). Since this will result in a reduction of the total clusters to be reduced by 1. C should be modified accordingly.

$C = C - 1$

If the merged cluster takes the index previously used by the active cluster *p*, following modifications to this merged cluster's data count shall also be adjusted.

$N_{p,new} = N_{p,old} + N_g$

9. At this step, the processing of the current data x_k is complete and the procedure will continue by looking at the next data. Therefore, go to step 5

3.4 EXPERIMENT

3.4.1 Data Description and Collection Process

Ford's SYNC infotainment system, through the years, have become one of most noted one on the market integrating functions such as music, navigation and in-vehicle feature controls (i.e. a/c, heated seat … etc.). As a result, Ford vehicles equipped with SYNC by default will have the vehicle's GPS signals broadcast in the high speed CAN (Controller Area Network) due to the GPS hardware is standard on those vehicles even if the customer chose to not have the navigation function added.

Ongoing efforts from Ford Research have been focused on methods to automatically extract, summarize and populate predictive models based on GPS information. Example of such activities involves learning and maintenance of the decision models for a driver associated with frequent trips and destinations [FT11, MF12]. Further development of such system involves the automatic compression, learning and prediction of sets of possible road segments to be traversed in the upcoming trip with varying probabilities [TF16, YT15].

In the development process, several employee personal vehicles with SYNC system were equipped with data logger to retrieve and store GPS information populated on the CAN network through an off-the-shelf OBD dongle at 1 HZ rate. Periodically, the dongle will be unplugged and data being downloaded and converted into Matlab file format for further process. The use of the GPS data stream will focus on the longitude and latitude coordinates but not the altitude. Context based trip learning requires the recognition of frequent locations and a reliable mean to compress GPS data stream into road segment to enable subsequent learning procedures. Both of those tasks will be detailed in Chapter 4. In subsequent paragraphs, we will demonstrate MIRGKL's performance for the location recognition task compared with variants of GK clustering algorithm. Also, we will show how MIRGKL as an evolving clustering algorithm can be applied to tackle the problem of online compression of GPS data stream into re-usable road segment information. After that, we will conclude this chapter.

3.4.2 MIRGKL for Location Recognition Using GPS Stream Data

Identification of frequent destinations with GPS stream data requires filtering out data corresponds to vehicle being driven and occasional stops for traffic and traffic controls. This can be done by assess signals related to ignition on, vehicle speed and the total stop duration. The end result of the above is a set of "*stop positions*" that corresponds to prolonged stop durations (> 30 minutes) that need to be summarized in a compact representation of clusters of locations without the true label making it an unsupervised problem suitable for MIRGKL clustering algorithm. The reason being most of our customer's usage of their vehicles concentrated on certain highly frequent locations (i.e. home, work, school, super markets … etc.) being either as a start location or the destination of trip. In addition, the parking position for visiting the same location will vary due to customer choice, parking availability (i.e. available slot on the street may vary from visit to visit) and the fact that depending on the number of satellites the GPS unit can make use of at the moment of operation the position reading will also have certain level of uncertainty and visit-to-visit variability even for the same exact position.

Variants of evolving clustering algorithms we experimented with include the following:

1. Evolving Gustafson-Kessel-Like Clustering Algorithm (eGKL)
2. Evolving Gustafson-Kessel-Like Clustering Algorithm with Merging Mechanism (eGKL w. merge)
3. MIRGKL Clustering Algorithm (MIRGKL)
4. MIRGKL Clustering Algorithm with Merging Mechanism (MIRGKL w. merge)
5. Evolving Gustafson-Kessel-Like Clustering Algorithm with Jensen-Shannon divergence (mutual KL divergence) as the similarity measure (JS or Mutual KL)

When the true labels are available for a clustering algorithm, ROC curve would be a good measure of a clustering (classification) performance in a general sense. However, since the true labels are not available we resort to other alternative measures to collectively assess the quality of the outcomes of abovementioned cluster algorithm.

1. Total Cluster Coverage Area

   This criterion involves the summation of the coverage of all cluster created after the stop position data passes through the algorithm in an online fashion (each data will be observed and processed for one time only). For the same location with repeated observations (stop position data readings), the cluster's center is incrementally adjusted and its coverage shall decrease with readings with low variability (high certainty). When a clustering algorithm creates more one and off clusters frequently with limited number of subsequent assignment of data and updates, the total coverage of all clusters tends to be large indicating relatively poorer ability to generalize data.

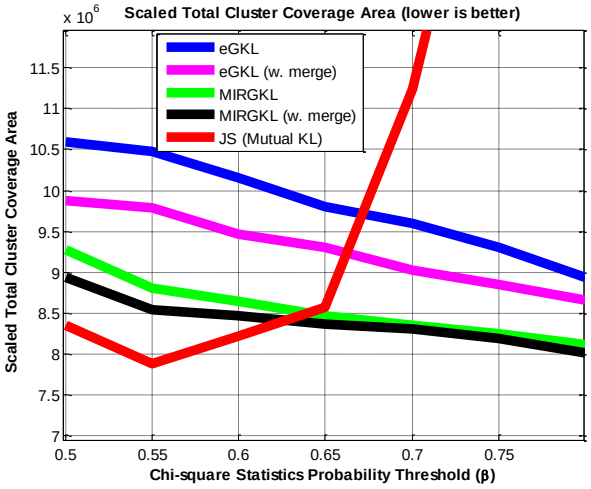


Figure 2: Total Coverage of Different Clustering Algorithms vs Proposed MIRGKL with Merging Mechanism.

In Figure 1, variants of eGKL cluster algorithms were compared across different probability threshold β. The version using Jensen-Shannon divergence (JS distance or mutual KL divergence) as the similarity measure performed best between β to be 0.5 to 0.6 but its performance quickly degrades as β increases which increases the size of the default cluster coverage. The issue with JS distance applied to GPS data which consist of spatial information is that the 1st and 3rd terms in (8) may force clusters of similar shape and sizes to be combined which are not desirable for this type of data. All other variants of eGKL clustering methods yield slow decreasing value over increasing probability threshold, β, and indicative of improvement in their clustering results. This is because a smaller total coverage (overall tighter) equates to a more precise representation of all data being captured by the clusters.

2. Total Number of Clusters

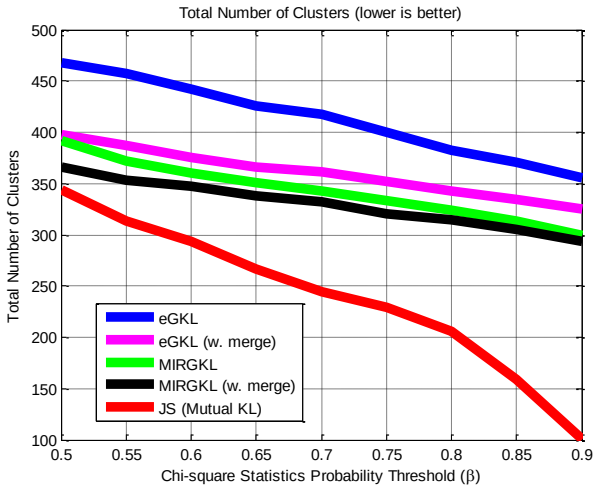


Figure 3: Total Number of Clusters Generated by Different Clustering Algorithms vs Proposed MIRGKL with Merging Mechanism.

Due to the fact that true labels of stop locations are not available, the actual number of total clusters (each represents a location) cannot be obtained. Even if the true labels are available, there are situations associated with a single location correspond to multiple locations. For example, in the case of a large parking area, due to the current version of the MIRGKL algorithm started each new cluster with the same covariance, even if the initial cluster may expand over time it may still require multiple clusters to capture a parking area much larger than the one defined by default covariance matrix.

Nevertheless, from Figure 2, similar decreasing trend can be observed for eGKL, eGKL with merge, MIRGKL and MIRGKL with merge cluster algorithms that as probability threshold increases. The reason of such pattern is that larger probability threshold effectively increases the radius of each cluster rendering more points to be assigned to existing clusters. In turn, it subdues the likelihood of new cluster creation resulting less clusters as the parameter β increases.

3. Entropy of Obtained Cluster Centroids

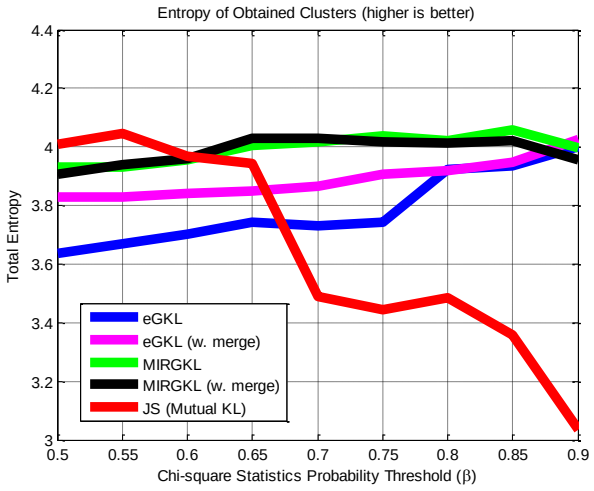


Figure 4: Entropy of the Centroids of Clusters Generated by Different Algorithms vs Proposed MIRGKL with Merging Mechanism

The ability of the KL-divergence (even in JS form) to absorb data simply due to shape similarities degrades its performance very rapidly from this angle as it generates much smaller amount of clusters covering undesirably very large areas. This may not be an issue for image based abnormal cell detection in the medical field of research but in our case of utilizing it for GPS data modeling may not be an appropriate choice.

In information theory, entropy represents a quantifiable measure of the unpredictability, diversity or the overall randomness of the content. When the content is expressed as a probability distribution with probability mass function P(X), the entropy H(X) can be expressed as the following:

$$H(x) = E[-\ln(P(X)] \tag{20}$$

For a clustering algorithm in a particular setting, the higher the entropy value (estimated from the cluster centroids) the more diversity (less similarity) among the clusters generated from data. As a result, the higher the value typically indicates clustering result with better quality. In Figure 3, the MIRGKL and MIRGKL with merging algorithm's overall perform are better than its peers. Although their advantage decrease with respect to eGKL and its variants as the probability threshold increased to over 0.8. Also to be note is that the

JS version of eGKL showed most erratic patterns in its performance which degrades rapidly after the probability threshold goes beyond 0.65.

4. Davies-Bouldin Index

Davies-Bouldin index was proposed in [DB79] where Si is the cluster specific scatter measure which leads to the definition of another cluster specific parameter, Ri and the mean of all the clusters generated by a clustering algorithm denoted by DB. DB defines a quantitative (not in an absolute sense) measure associated clustering algorithm where a clustering algorithm with a lower DB value is generally considered to have relatively better clustering quality when compared with one with a higher DB value.

$$DB = \frac{1}{N}\sum_{i_1=1}^{N} D_{i_1} \tag{21}$$

$$D_{i_1} = maxR_{i_1,i_2}; i_1 \neq i_2 \tag{22}$$

$$R_{i_1,i_2} = \frac{S_{i_1}+S_{i_2}}{M_{i_1,i_2}} \tag{23}$$

$$S_i = \left(\frac{1}{T}\sum_{j=1}^{T_i}\left\|X_j - A_i\right\|_k\right)^{1/k} \tag{24}$$

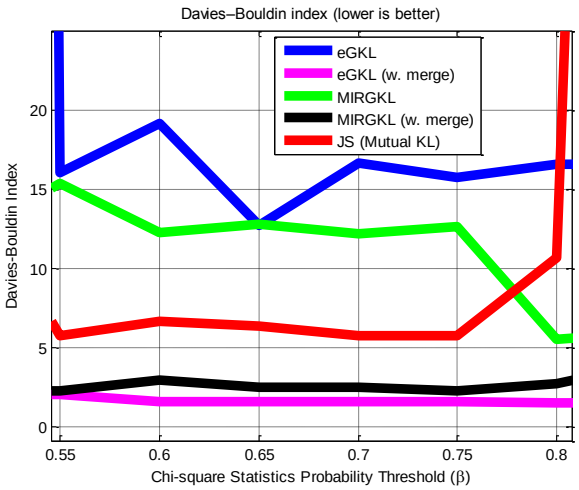


Figure 5: Davies-Bouldin Indices of the Centroids of Clusters Generated by Different Algorithms vs Proposed MIRGKL with Merging Mechanism

From Figure 4, one can observe that eGKL and MIRGKL with merge mechanisms are the best performing clustering algorithm we tested across the probability threshold

values from 0.55 – 0.8. Next to them would be JS (Mutual KL divergence) based eGKL, however, similar to other previous figures this similarity is sensitive to the probability threshold and its performance degrades rapidly when probability threshold is set to be larger than 0.8.

Summarize the observation of Figure 1 – 4, proposed MIRGKL clustering algorithm yields favorable performance among different quantitative cluster quality measures applied. In most cases, across specified probability threshold, it is either the best or 2nd best performing clustering method with smooth performance variations.

3.5.3 MIRGKL for Road Automatic Segmentation

In this section, the MIRGKL clustering algorithm is applied for online information (GPS trace) compression. The reasons to perform such task include the following:

1. Reduce requirement for local storage of GPS traces from different trips.
2. Similar to 1, when GPS trace is sent with a compressed form (data reduction) it also reduces the time needed to transfer such data out of the vehicle (if such transfer is needed)
3. Reduction in variables for state transition modeling. Vehicle traversing from road segment to road segment can be modeled as a Markov model. While most road segments have only one possible transition, some segments will have multiple possible states to transition to (i.e. cross road where left, right, going straight are all possible next states or on the highway near an exit there exist two possible next states of either staying on the highway to take the road segment associated with the exit ramp). Road segment definitions from map providers typically imposed a maximum limit on how long a segment should be and also their definitions of road segment are for the purpose of not only preserving longitudinal and latitude position information but also other road segment attributes such as curvature, grade, road class … etc. As a result, using an online cluster algorithm to compress GPS trace for the purpose of preserving only

longitude and latitude GPS position information result in much compact representation for subsequent model building tasks.

Applying MIRGKL clustering algorithm to compress GPS trace involves a key modification to the original method. That is, only the previous cluster in involved in the decision making of creation of a new cluster. As a result, road segments can be obtained as A. Results from the modified MIRGKL clustering algorithm. Or B, A separate class of clusters where their centroids are placed at the position where new clusters being created (cluster creation represents high value in novelty index). In general, alternative B preserves the GPS trace much closer to the actual one because it approximately captures the trace with segments separated within proximity of relatively large turns.

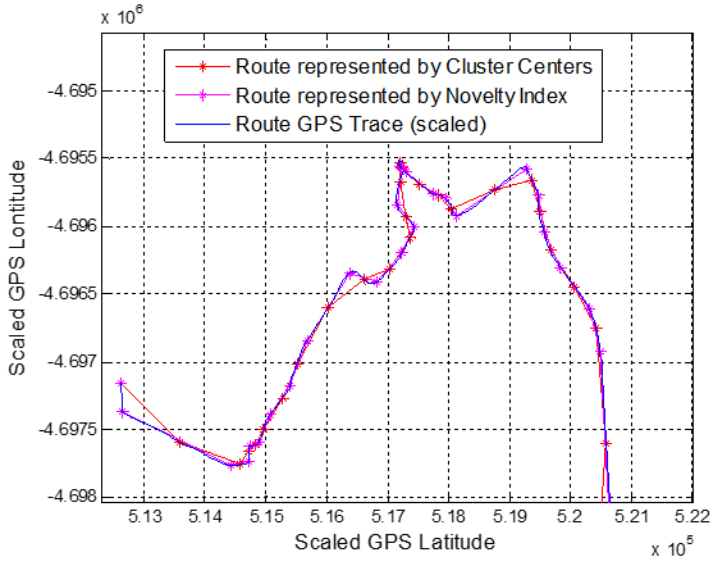


Figure 6: Compress GPS Trace with Modified MIRGKL Online Clustering Algorithm

The varying capabilities of alternative A and B are illustrated in Figure 5. Compared to the original GPS trip trace marked by dotted blue line, one can clearly observe the close resemblance of the reconstructed trace use alternative B where occurrence of a new cluster is used to mark the road segments. The reason alternative A, using the cluster centers from MIRGKL clustering

algorithm, doesn't work as well is due to the fact these cluster centers tends to located at or near center of a linear segment instead of at or near the break points between segments.

3.5.4 Comparison between Original RGKL and MIRGKL with GPS Trip Data

With the same data as shown in 3.5.3, we compare side by side the original RGKL and proposed MIRGKL clustering algorithm as engines for data compression. Since the data consist of 1HZ GPS readings from vehicle driven on a predetermine route, such task is equivalent to the segmentation of the full route in a more compact representation. For both clustering algorithms, we executed the task under different probability settings ranging from 0.65 ~ 0.99 where lower a probability value produces lower threshold for cluster creation yielding more clusters when compared with a higher probability threshold that for the same data and clustering method will result in relatively less clusters.

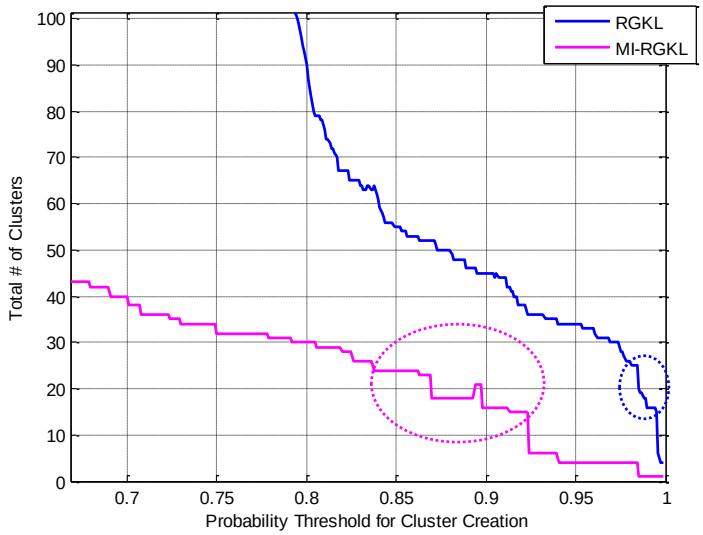


Figure 7: RGKL vs MIRGKL with different probability thresholds with GPS trip trace data

From Figure 6, note that in the data there are total of 15 ~ 25 major and minor turns that should be identified as proximity points corresponds to the creation of a new clusters. As a result, when the probability threshold is chosen appropriately the total of number of clusters identified should approximately fall into that range for both RGKL and MIRGKL clustering algorithm. For

MIRGKL, we noticed that such probability threshold ranges from 0.83 ~ 0.925 which is much wider than 0.975 ~ 0.994. The more linear relationship between probability threshold and total number of clusters obtained from MIRGKL when compared to RGKL is also visible when observing the whole pink line (MIRGKL) vs the blue line (RGKL). It is especially obvious when RGKL's creates rapidly increasing number of clusters when the threshold goes below 0.85; on the other hand, the MIRGKL's performance throughout the range between 0.65 ~1 exhibited an approximately linear increment in the number of clusters created as the threshold decrease.

In summary, MIRGKL, when compared with the original RGKL algorithm, its clustering outcome exhibited a pattern less sensitive to the probability threshold. This is mainly the effect of the mutual information based formulation which bounded and symmetrized the underlying Mahalanobis distance measure thus making the outcome of the clustering algorithm more controllable and such property is beneficial for parameter tuning when dealing data generated from different processes, phenomenon and nature.

3.5 Conclusion and Next Steps

We have described a novel clustering method from Evolving Gustafson-Kessel-Like Clustering algorithm where the similarity measure is modified to take the form of Mutual Information as proposed in the Jensen-Shannon Divergence as a smoothed, symmetric and bounded version of the KL-divergence. In the proposed similarity measure, we preserved only the Mahalanobis distance measure computations and rid of the portions that focused solely on the size and orientation of the clusters. In addition, we unified cluster assignment vs creation (more relaxed) and cluster merging (stricter) criteria with two probability thresholds. The method was tested and validated with GPS stream data where the evolving nature and 1-pass processing capability of an online clustering algorithm is highlighted to 1st determine and summarize frequent locations and 2ndly to compress the GPS trace into a more compact form. These two examples are crucial elements in building context aware intelligent navigation systems and will discussed more in detain in another paper.

Future research on evolving clustering method will focus on derivation of adaptive mechanism to generalize the weighting of the terms in the matrix form of KL divergence (8). This is because while most applications are suitable to use only the Mahalanobis term, some applications in the medical and biological imaging research demonstrates promising results while keeping the shape and orientation terms. Other future research topics will include the application of MIRGKL to embed its clusters with local cluster specific functions for non-linear function approximation and prediction. This is an especially difficult problem in data-driven system prognostics when 1st principle degradation model is not available and multiple degradation modes exist.

CHAPTER 4: MACHINE AND SYSTEM PROGNOSTICS WITH EVOLVING CLUSTERS EMBEDDED WITH LOCAL SURVIVAL FUNCTIONS

Remaining useful life (RUL) is defined as the useful life of an asset at a particular time of operation and it is of critical importance to condition based maintenance and prognostics functionalities of a health management system [SW11]. Majority of literature typically refer to prognostics as the capability to produce RUL estimate and involves methods that can be grouped into four categories: Model based methods, Knowledge based methods, Data driven methods and hybrid methods [GMM10]. Recently, there has been a renewed focus on this field of research due to demand of safer aviation and ground vehicles and relatively newly developed Integrated Vehicle Health Management (IVHM) systems [EJJ13]. Prognostics information is critical to the success of IVHM as the procurement of parts, scheduling of both jobs and maintenance of fleet vehicles and the scheduling of maintenance personnel and equipment all dependent on such information. Furthermore, prognostics information has been increasingly become a new viable source of revenue for companies want to rid them self of the role of a pure producer but also a service provider throughout the lifespan of their products.

Clustering, synonymous to cluster analysis, classification, numerical taxonomy and typological analysis represents the task of grouping object (with descriptive data) such that object in the same group share more similarity when compared to objects in other groups [XW05]. Online clustering methods have been proposed in [FG10, SR12, TK12, TS11, and AM11]. The reasons to applying evolving clustering method to RUL estimation problems have to do with its ability to 1. Deal with data with indefinitely size (data stream) 2. Dynamic operating conditions and 3 Operate under uncertainty in acquired information. Each identified cluster will also be associated with a local RUL approximation function which will be updated opportunistically.

In summary, proposed method aims at incrementally identify and refine multiple operational modes extracted with Mutual Information based Recursive Gustafson-Kessel-Like clustering

algorithm (MIRGKL). The learning of RUL curves is achieved from observing inferred degradation signal that arrives asynchronously and triggers the update of cluster specific degradation models. The remainder of the paper is organized as the following. Chapter 4.2 describes MIRGKL-plus where the original MIRGKL clustering algorithm is extended with embedded local functions to learn and estimate the RUL. Chapter 4.3 – 4.4 summarizes the setup of the experiment, procedures for the test, signals acquired in the process and how inferred RUL information was obtained. In Chapter 4.5, results for RUL estimation of the drill-bit data will be presented in detail. Chapter 4.6 will focus on the 2nd prognostics type of problem with the vehicle braking system. We will describe in detail a novel feature applied with MIRGKL-plus to predict the brake wear. Finally, in Chapter 4.7 we will draw conclusion for the research and identify future work and next steps.

4.1 REMAININNG USUFUL LIFE (RUL) LEARNING AND ESITMIATION USING MIRGKL-plus Algorithm

4.1.1 Mutual Information based Gustafson-Kessel-Like (MIRGKL) Clustering Algorithm and MIRGKL-plus

Jenson-Shannon divergence (JSD), a variant of KL-divergence and is a member of the Bregman divergence family [BM05b] and produced a smoothed, symmetrized and bounded formulation when compared to its original forms [L91] defined as the following:

$$D_{JS}(\aleph_0||\aleph_1) = \frac{1}{2}D_{KL}(\aleph_0||\aleph_M) + \frac{1}{2}D_{KL}(\aleph_1||\aleph_M) \qquad \text{same as (9)}$$

Where $\aleph_M = \frac{1}{2}\aleph_0 + \frac{1}{2}\aleph_1$

Proposed in Chapter 3 is a special case of the above where the shape and size information contained in the covariance matrix are not to be taken into account separately other than as a normalizer, which becomes the following:

$$D_{MM}(\aleph_0||\aleph_1) = \frac{1}{2}D_M(\aleph_0||\aleph_M) + \frac{1}{2}D_M(\aleph_1||\aleph_M) \qquad \text{same as (10)}$$

Where $\aleph_M = \frac{1}{2}\aleph_0 + \frac{1}{2}\aleph_1$

And D_M is the original Mahalanobis Distance:

$$D_M(\aleph_0||\aleph_M) = \sqrt{(\mu_0 - \mu_M)^T \sum_M^{-1} (\mu_0 - \mu_M)} \qquad \text{same as (11)}$$

During the clustering procedure, (MM1 and MM2) are applied 1st to compare the new piece of information contained in the new data is to be assigned to an existing cluster or to trigger the initialization of a new one. Secondly, it is also applied to evaluate whether the active cluster, after update, should be combined with another existing cluster and as a result reduce to complexity of the overall model structure. Using the Chi square statistics, $\mathcal{X}^2_{n,\beta}$, while the dimensionality value n is kept constant through the process the value of β may of different values during cluster assignment and cluster consolidation process.

4.1.2 Weibull Survival Function as A Local Approximation Function

When local models are attached to existing clusters, they collectively can perform as function approximators [K94] activated in accordance to varying inputs. In addition, they are also suitable for online modeling and control of complex process with additional implication in fields such as fault detection, performance analysis, forecasting and knowledge extraction. Multiple clusters embedded with its own specific model is similar in notion of a knowledge base consists of fuzzy IF-THEN rules extracted from data capable of case based reasoning and predictions [AF94]. This type of model is especially suitable for the application of interest where multiple degradation modes may exist and a detailed physics or model based degradation model is not available.

A hazard function (HZF) represents operation of a population of devices and can viewed to have 3 distinct phases including A: Infant mortality (burn-in), B: Random failure (useful life) and C: Wear-out period [KK03]. A HZF has a generic form of the following:

$$h(t) = \frac{f(t)}{1-F(t)} \tag{25}$$

where f(t) and F(t) are the failure distribution function and the cumulative failure distribution respectively.

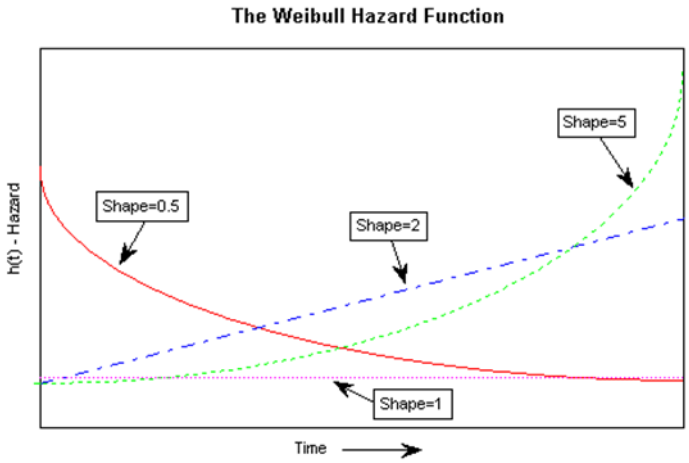


Figure 8: Weibull Hazard Function with different shape parameters

Weibull distribution is a probability distribution first described in detail by Waloddi Weibull in 1951 [W51]. When the random variable x represents "time-to-failure" (TTF), it gives a distribution of failure rates proportional to the power of time. Shape parameter k when smaller, equal to or larger than 1, represent cases where a failure rate decreases with time, constant or increases with time.

$$f(x;\lambda,k) = \begin{cases} \frac{k}{\lambda}(\frac{x}{\lambda})^{k-1} e^{-(\frac{x}{\lambda})^k}, & x \geq 0 \\ 0 & , x < 0 \end{cases} \quad (26)$$

Weibull distribution, as a viable choice of the failure distribution, has been extensively used in modeling of reliability, failure and fatality analysis, engineering and biological studies [CL13, AN14, NA15, UG15, BL12]. It is found to be useful in practice for representing a HZF due to its generic bathtub shape, high flexibility and its statistical moments' relationship to the interpretation of failure rates. Complementary cumulative distribution function of the Weibull is often associated with the modeling of the survival probability. The probability distribution function, f, cumulative distribution function, F, and the complementary of F are shown in (26, 27, 28)

CDF of Weibull Distribution is often associated with modeling of a hazard function:

$$F(x,\lambda,k) = 1 - e^{-(x/\lambda)^k} \quad (27)$$

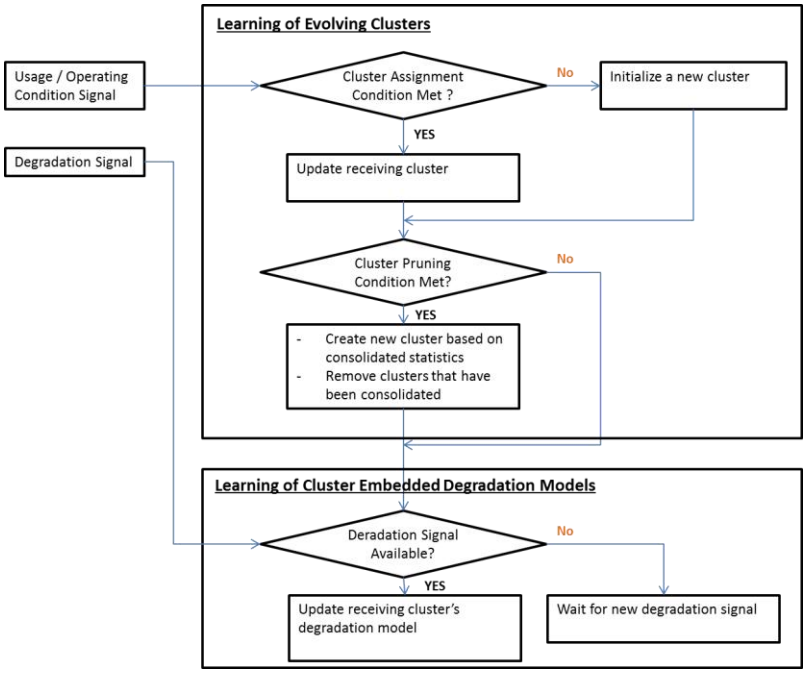
Complementary CDF of Weibull often associated with modeling of a survival function:

$$S(x, \lambda, k) = e^{-(x/\lambda)^k} \tag{28}$$

Furthermore, linearization can be performed to (28):

$$\ln(-\ln S(x, \lambda, k)) = k * \ln(\mathrm{x}) - k * \ln(\lambda) \tag{29}$$

Formula (29) represents another reason why we chose to use the survival function derived from the Weibull distribution as the local function of a cluster. While the MIRGKL clustering is capable of online operation, online identification of cluster specific function's parameters becomes critical to ensure the overall method still maintains its capability for online operation. The fact that (28) has a linearized form of (29) makes recursive parameter identification methods such as recursive least square or Kalman filter to be applicable. For certain applications where the choice of cluster specific local function cannot be linearized or recursive (online) updates is not possible, the learning of associated parameters becomes more trivial as all or a portion of the data associated with the cluster will have to be buffered to facilitate the latter parameter identification procedure.



Flowchart 4.2: Flowchart of Evolving Clustering with Embedded Local Functions for System Prognostics

In Flowchart 4.2, we present the overall method's major procedures. The whole method can be broken down into two parts: The upper part (A) of the flowchart has to do with the usage and operation condition identification through MIRGKL clustering algorithm. The lower part (B) of the flow charted deals with the learning of cluster specific degradation models.

For most systems, data required for part A of Flowchart 4.2 is typically more readily available as part of the control strategy or generic operating condition sensing. For a machine with rotational parts, they could include things like the RPM readings, loadings and sometimes thermocouple signals from which statistics, model based or experienced based features can be calculated. When combined with other state information such as the On/Off or predefined operation modes, conditional usage patterns can be retained by processing incoming signals separately for each state. Such predefined state can be understood as an alternative to the clusters identified by MIRGKL algorithm.

Lower part B of Flowchart 4.2 depicts the learning procedure of cluster specific local models. Unlike part A of the Flowchart, the degradation signal is often not available without some definition of the progression of degradation being defined and measured by added sensor(s). Different alternatives exist as to obtaining indirectly inferred degradation signals. For example, the survival probability within a predefined time window can be computed from machine maintenance records where Operation Time to Failure (OTTF) and Time to Repair (TTR) information are available. Another example to infer degradation information would be to associate brand new part or machine (inspected) to have zero or close to zero degradation and full degradation when the part or machine is determined to be non-functional to its designated task. If feasible in implementation, intermediate degradation levels can be determined with the assistance of certain indirect physical measurements. In the case when part A and B are highly asynchronous, the training in part B can be either inactive or to create inferred degradation information through interpolation to facilitate

the learning process (require buffer or storage of information). Dependent on how the degradation data is obtained, part B may be executed in a delayed fashion. For example, if RUL is applied as the degradation signal and such information won't be available until EOL (End of Life) of part to actually take place. In this situation, part B won't be executed until occurrence of an EOL event. In addition, cluster assignment sequence from part A need to be buffered such that identified clusters' local degradation models can be updated in accordance to that sequence.

4.2 EXPERIMENT: DRILL-BIT RUL ESTIMATION

4.2.1 Experimental Setup

Drill process is one of the most widely applied machining processes and is chosen as the test bed for validation of proposed autonomous prognostics framework. HAAS VF-1 CNC Machining Center with Kistler 9257B piezo-dynamometer (250 HZ) to drill holes in 1/4-inch stainless steel bars. High-speed twist drill-bits with two flutes were operated at a feed rate of 4.5 inch/minute and spindle speed at 800 rpm without coolant. Each new drill-bit were visually inspected to ensure no visible defect present on the unit. Once the drill process starts, it continues until a physical failure was detected. These physical failures are typically associated with excessive wear and/or deformation result from high temperature during its operation. During each drilling task, dynamometer captures thrust force and torque signal from a torque sensor in the form of time series data. While we demonstrate the effectiveness of proposed method utilizing the drill-bit data, it should be straightforward that it is applicable to a wide array of machine and systems in a cooperative learning setup. Dependent on the nature of the machine or system and signal availability, modules involving signal processing, feature extraction and degradation signal inference need to be adjusted in a case by case fashion.

4.2.2 Feature Extraction and Inferred RUL Information

4.2.2.1 Dynamic Time Warping (DTW) Algorithm

In real-world problems, time series data from repeated measurements or data from different channels may have different sample rates, varying sampling rate(s) or both that make them to be

out of phase when pairwise comparisons need to be performed. In those situations, widely applied Euclidean distance's [KK03], due to its known weakness of sensitivity to distortion in time axis [KR05], may not be the best choice for such tasks. Dynamic Time Warping (DTW) is a distance (similarity) measure originally developed in the speed recognition community [SC78] which includes a non-linear mapping procedure to minimize the difference of the time series pair of interest before compute their differences. Despite its computational complexity, the method's ability to deal with local out of phase signals, improve signal alignment and dealing with signals of different lengths make it the best known algorithm to compare and derive a quantifiable measure for time series patterns in the data mining [KP00, DT08, and RC12], bioinformatics [AC01], medicine and engineering community.

Suppose we want to compute the similarity between two time series, one sequence Q with length of n and another sequence C with length m, where

$$Q = q_1, q_2, \cdots, q_i, \cdots, q_n \tag{30}$$

$$C = c_1, c_2, \cdots, c_j, \cdots, c_m \tag{31}$$

The warping cost is expressed as the following

$$DTW(Q, C) = min\left\{\sqrt{\sum_{k=1}^{K} w_k}\right\} \tag{32}$$

A n-by-m matrix with the squared (Euclidean) distance between q_i and c_j are computed through $(q_i\text{-}c_j)^2$ and w_k is the matrix element $(i,j)_k$ that belongs to the kth element of a warping path W. The optimal path is typically found by dynamic programming and it is out the scope of the present paper itself.

4.2.2.2 Feature Extraction of Drill-bit Data with DTW

To demonstrate the capability of MIRGKL embedded with survival function for prognostics purpose, we resort to conceptually simple feature computed using DTW. For each drilling task of a drill-bit, the DTW computation will compute the distances of thrust force (F) and torque (Q) with respect to a pair of nominal thrust force (TF_n) and nominal torque (TQ_n) profiles conducting the

same task. Essentially, the functioning features we are using are the deviations between the most current thrust force and torque profiles compared with some nominal ones. The 2-D feature vector as a result can be expressed as the following:

$$X_{T=1,2,\cdots,t} = x_1, x_2, \cdots, x_t \tag{33}$$

$$x_t = [DTW(TF_n, TF_t), DTW(TQ_n, TQ_t)]\ ) \tag{34}$$

It should be noted that T is the cumulative count of number of drilling tasks conducted for the current drill-bit. The feature calculation as described in (33) and (34) takes place after the completion of a drilling task when the thrust force profile and torque profile become available to compute $DTW(TF_n, TF_t)$ and $DTW(TQ_n, TQ_t)$ according to (34). Nominal thrust force, TF_n, and nominal torque, TQ_n, are typically the determined from the first few fully completed drills. For this dataset we've determined that the 2nd drilling tasks for each drill-bit to be used as the nominal profiles due to large variability among the 1st drilling profiles between drill-bits.

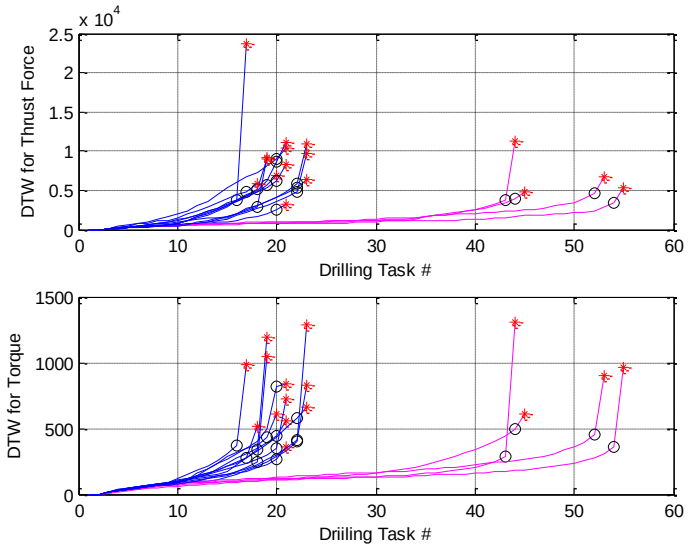


Figure 9: Feature calculation using Dynamic Time Warping (DTW) with Thrust Force (upper figure) and Torque (lower figure) with multiple drill-bits. Blue lines represent drill-bits lasted less than 30 drilling tasks and pink line represent those lasted more than 30 drilling tasks

In Figure 4.2, Feature calculation using DTW with Thrust Force (upper figure) and Torque (lower figure) with multiple drill-bits are shown. Using data from 2nd drills as the nominal profiles,

both DTW based features trended up as the drill-bit degrades and eventually reaches a high point after which the drill-bit no longer usable. The last feature values for each drill bit are marked with a red star marker and the prior value marked with a black circle. To make the outcome of the prognostics prediction from proposed algorithm more useful, we determine that the RUL of the drill-bit to be close to zero for the drilling task prior to the last one. The reason being if the last drilling were to be started, it is highly likely the finished drill would be deemed unsatisfactory. Therefore, the algorithm shall be able to predict RUL is reached or fairly close after the drill prior to the last one was completed to avoid defective parts being produced as a result of a bad drill-bit.

The two dimensional DTW based feature expressed in (R) is applied to the MIRGKL clustering algorithm as shown in the upper portion of Flowchart 4.2 to identify major operating patterns as clusters that are reasonably unique when compared to one another. The following paragraph will describe in detail the determination of RUL based normalized degradation information for the training of cluster specific degradation models.

4.2.2.3 Determine Inferred RUL Information

As described in 4.2.2, we obtain nominal profiles from the 2nd drilling task of each drill-bit to perform DTW based feature calculation. In addition, we define the drill prior to the last one for each drill-bit as RUL being reached to avoid additional drilling tasks being done. As a result, the normalized RUL's for the 3rd drilling task and the drilling task prior to the last one for each drill-bit was set to be:

$$RUL_N(D, T = 3) = 1 - \varepsilon; \tag{35}$$

$$RUL_N(D, T = T_D - 1) = \varepsilon; \tag{36}$$

Where D represents the drill-bit number, T_D represents the total number of drills a drill-bit performed and ε represents a small number to avoid numerical difficulties in the learning procedure of cluster specific local models as described in (M). For the RUL_N of drills between T=3 and $T=T_D-2$, we linearly interpolates their values according to $RUL_N(D, T = 3)$ and

$RUL_N(D, T = T_D - 1)$. This interpolation procedure enables cluster specific local models to undergo more updates rather than a total of only two updates amongst all existing clusters for during a drill-bit's life cycle. Intermediate drills' inferred normalized RUL through interpolation can be expressed:

$$RUL_N(D,T) = 1 - \varepsilon - \frac{1-2\varepsilon}{T_D-3} * (T_D - T); \quad 2 < T < T_D \tag{37}$$

In summary, inferred normalized RUL for the nominal drill (3rd drill) is assigned to have a value close to 1 and the drill before the last for each drill-bit to be close to zero. From these two initial and final inferred normalized RUL values, intermediate ones were generated by linear interpolation. Since T_D will not be known until the drill-bit actually reaches end of life, $RUL_N(D,T)$ information and subsequent cluster specific degradation models' updates will occur at a delayed fashion with obtained cluster sequence buffered throughout the drilling process.

4.2.3 RESULTS

4.2.3.1 Initial Data Analysis

With the setup as described in 4.2.1, total of 16 drill-bits were used in the experiment where each piece was in brand new condition with no noticeable visual imperfections. After the test during which each drill-bit perform identical drill task until EOL, we obtained following information as to how many drills each drill-bit completed.

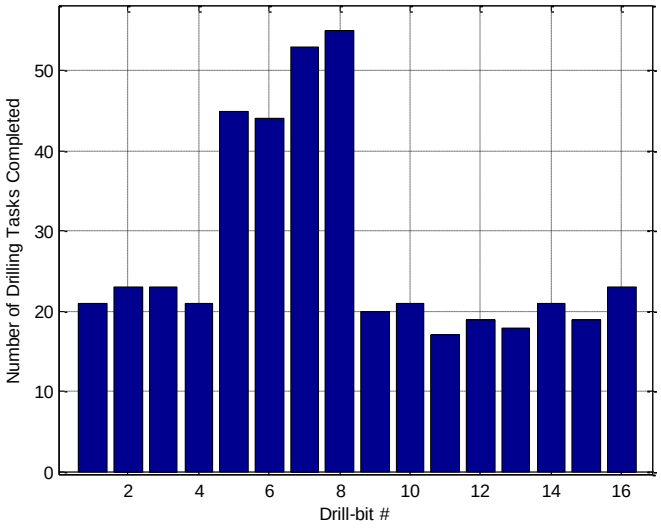


Figure 10: Number of drilling tasks completed for all 16 drill-bit in the experiment

In the experiment, drill-bit # 5, 6, 7 and 8 lasted almost twice the number of total drilling tasks as all other samples. Given that the subjects of the drilling, stainless steel bars, were of the same spec, we assume this large variance in the useful life may be results from:

- Parts (drill-bit and stainless bar) may have non-uniformity that is not easily detectable from visual inspections
- Drilling process controls (programmed thrust force and / or torque profiles) may be close to the operational limit of the setup

However, such large variation in their degradation profiles also makes this particular set of experimental data suitable for data-driven prognostics method based on evolvable models such as the one we are proposing. The rationale is that the MIRGKL will online identify evolving clusters matching different groups of repeated operational patterns with local degradation models learned in a delayed on-line fashion matching uniquely different degradation profiles.

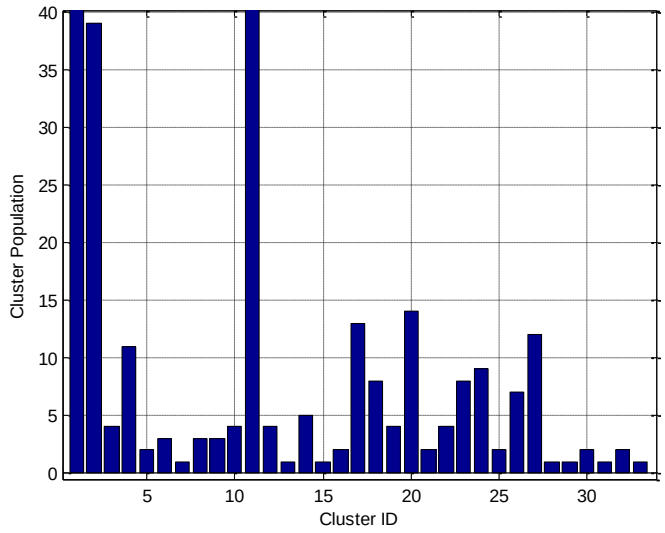


Figure 11: Population of each cluster identified by MIRGKL Algorithm

With total of 16 similarly conditioned new drill-bit, total of 443 drilling tasks were carried out. As described in 4.2.2, the data first drillings of each drill-bit was skipped due to excessive noise and as a result the DTW based feature calculation using the 2nd drilling task of each drill-bit as the nominal pattern starts from the 3rd drilling task (self-comparison is skipped). Adding the definition of the drilling task before the last one to have the lowest assigned normalized RUL value, the number of each drill-bit's drilling tasks being used for modeling will be reduced by 3. Therefore, original 443 drilling tasks from 16 drill-bits were reduced to 443-48 = 395 to validate the method. Using MIRGKL online clustering algorithm, total of 33 clusters were identified with the top 3 most utilized cluster being #1, #2 and #11 each experienced more than 39 (or 10% of all populations) updates from drilling patterns being assigned to them. Cluster #3, #5 ~ #10, #12, #13, #15, #16, #21, #25 and #28 ~ #33 each experienced less than 5 (or 1% of all populations) updates making them candidates to be dropped off during prediction phase due to lack of amount of training / significance.

4.2.3.2 Setup #1 – Fully Online Mode (Perfect Information)

As previously described, the inferred normalized RUL information won't become available until the drill-bit in the current experiment has reached RUL. As a result, the training of cluster specific local model in reality would take place in a delayed manner relative to MIRGKL clustering algorithm. However, to demonstrate the model's performance in cases when (inferred) RUL may indeed be available enabling the local degradation model to be updated subsequent to cluster assignment, we present the DTW pattern and inferred normalized RUL information in a fashion to simulate such a scenario. In other words, for each drilling task, the DTW based patterns first is applied to determine a cluster with the best match to predict the remaining useful life (prediction phase). After that, the cluster learns from the pattern by either update an existing cluster or creates a new one. Finally, assuming the availability of inferred RUL information, the active cluster's local degradation model learns from that In a recursive least fashion. Since the overall model is updated essentially at a one sample delay, the overall performance in this setup is quite good. In Figure 12, 13 and 14, we observe the predictions to track the inferred RUL values very closely after a few observations even between major pattern shifts where the inferred RUL doubled between the 4^{th} and the 5^{th} drill-bits (end of Figure 12 to the beginning of Figure 13) then reduced by half between the 8^{th} and the 9^{th} drill-bit (end of Figure 13 to the beginning of Figure 14). Throughout the whole process as shown from Figures 12 through 14, the MIRGKL algorithm operates in a fully online mode and we separate the plots into 3 only for visualization purpose.

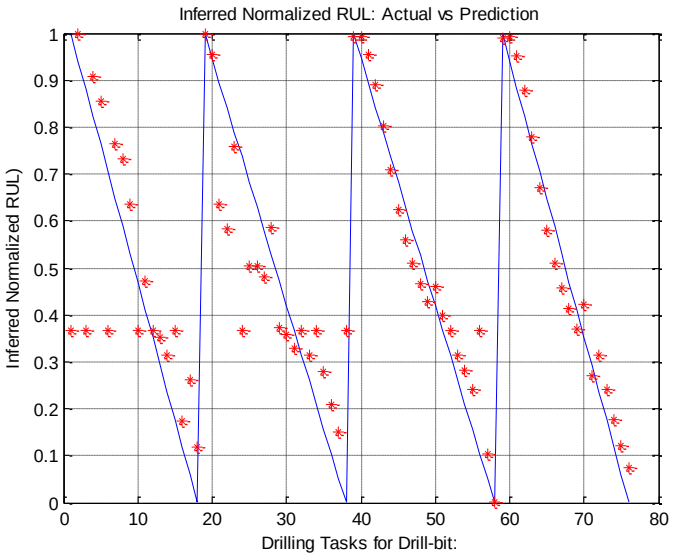


Figure 12: Normalized Inferred RUL vs Predicted RUL for Drill-bit #1~#4

In Table 4.1 and 4.2, we also show the history of cluster assignment history where new cluster creation is marked with * marking

Table 4.1: Cluster assignment history of drill-bit #1 ~ #4

Drilling Task #	Drill-bit #1 Cluster Assgnment	Drill-bit #2 Cluster Assgnment	Drill-bit #3 Cluster Assgnment	Drill-bit #4 Cluster Assgnment
1	* 1	1	1	1
2	* 2	1	1	1
3	2	1	1	1
4	2	1	1	1
5	2	2	1	1
6	* 3	1	1	1
7	* 4	1	1	1
8	3	2	1	11
9	* 5	2	1	11
10	* 6	2	1	11
11	* 7	1	2	1
12	* 8	* 11	2	2
13	* 9	1	4	1
14	9	* 12	* 17	2
15	* 10	12	17	4
16	10	* 13	17	* 20
17	8	* 14	17	20
18	8	12	* 18	18
19		* 15	* 19	
20		* 16	16	

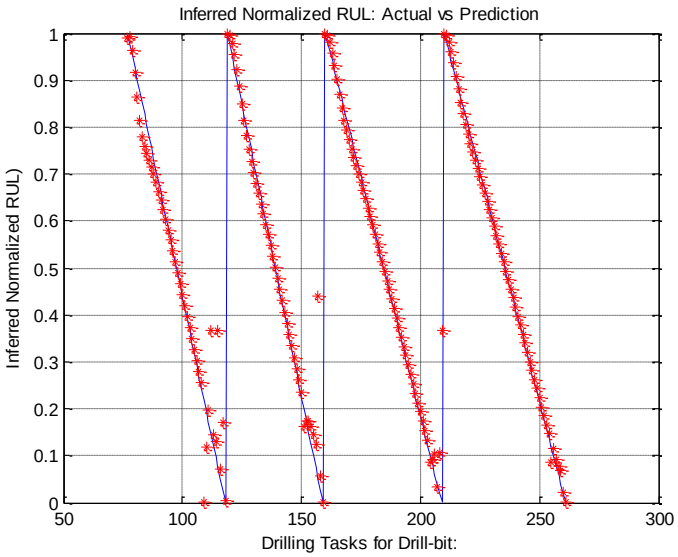


Figure 13: Normalized Inferred RUL vs Predicted RUL for Drill-bit #5~#8

Table 4.2: Cluster assignment history of drill-bit #5 ~ #8

Drilling Task #	Drill-bit #5 Cluster Assgnment	Drill-bit #6 Cluster Assgnment	Drill-bit #7 Cluster Assgnment	Drill-bit #8 Cluster Assgnment
1	1	1	1	1
2	1	1	1	1
3	1	1	1	1
4	11	1	1	1
5	11	1	1	1
6	1	1	1	1
7	11	1	1	11
8	11	11	11	1
9	11	11	11	11
10	11	11	11	11
11	11	11	11	11
12	11	11	11	11
13	11	11	11	11
14	11	11	11	11
15	11	11	11	11
16	11	11	11	11
17	11	11	11	11
18	11	11	11	11
19	11	11	11	11
20	11	11	11	11
21	11	11	11	11
22	11	11	11	11
23	11	11	11	11
24	11	11	11	24
25	11	11	11	24
26	11	11	11	24
27	11	11	11	24
28	11	11	11	11
29	1	11	1	24
30	1	11	11	11
31	1	11	* 24	24
32	1	1	1	11
33	20	2	1	11
34	2	2	1	11
35	1	2	11	11
36	* 21	2	1	24
37	20	2	1	11
38	21	2	11	1
39	20	* 23	1	11
40	20	* 23	1	1
41	* 22	19	1	1
42	14		1	1
43			1	1
44			11	1
45			2	1
46			20	2
47			20	24
48			20	1
49			17	26
50			* 25	20
51				22
52				14

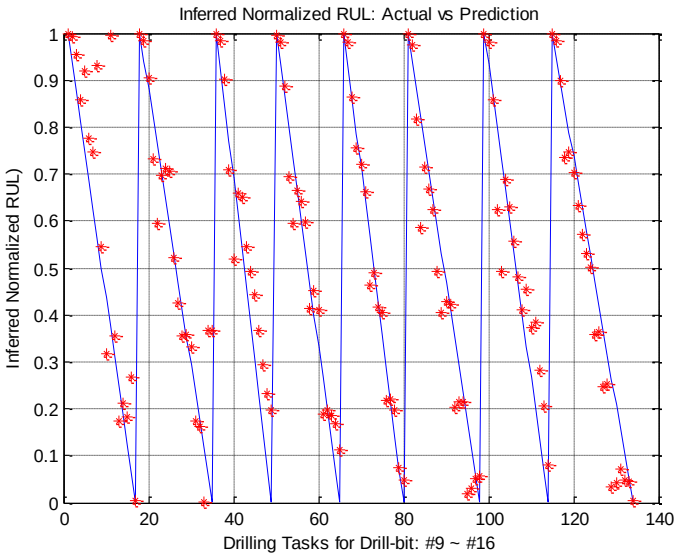


Figure 14: Normalized Inferred RUL vs Predicted RUL for Drill-bit #9~#16

Without additional processing of the predicted RUL values, fully online implementation of the MIRGKL with local models provides estimations closely follow the true values (inferred normalized RUL). These results show that through the use of multiple learned degradations distributions, effective estimation of the RUL can be achieved even if the true underlying distributions and how many of them may be unknown.

4.2.3.3 Setup #2 – Semi-Online Mode (Delayed Information)

In 4.2.3.2, we described a setting where each drilling patterns comes with an inferred normalized RUL reading to demonstrate MIRGKL with embedded local functions to make 1-step-ahead prediction based on the models that are incrementally updated online. However, for many real-world problems such a small delay in the availability of direct or indirect knowledge of RUL related information is unrealistic. The drill-bit experiment is a good example of such a case because it is not possible to know the actual RUL information until actual EOL (end-of-life) has been reached. Only then the RUL information of past drilling tasks' RUL can be back filled where the initial nominal normalized RUL assigned to have a value close to 1 (1-ε) and the last drill to have a RUL close to zero ε.

In this section, to reflect the actual timing when inferred RUL become available we engage in the local functions' update only when a drill-bit's EOL has been reached. Before such scenario occurs, the MIRGKL operates its clustering portion (update, create and pruning) without any update made to the local degradation model from which RUL prediction are made. In other words, we introduced a variable delay in the learning of the local degradation models that is caused by the fact each drill-bit will last somewhat different numbers of drilling tasks before they fail. This is a much more challenging setup in comparison to 4.2.3.2 but matches closely to the scenario with respect to the delayed fashion RUL information becomes available for model refinement (training) and predictions have to be made accordingly.

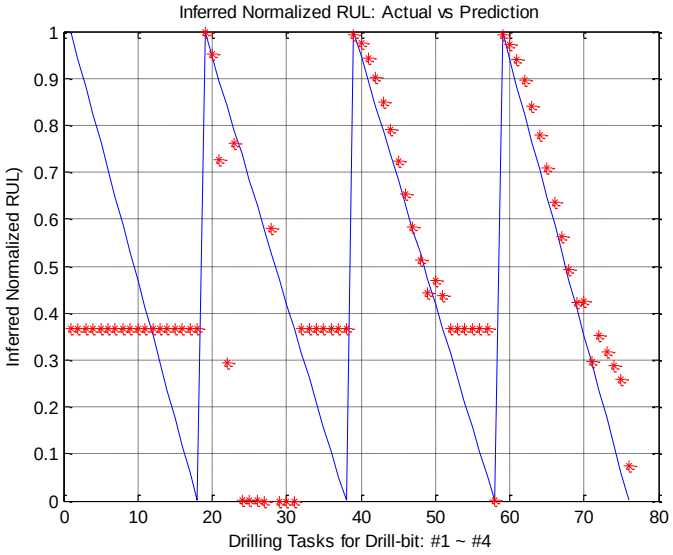


Figure 15: Normalized Inferred RUL vs Predicted RUL for Drill-bit #1~#4 in Delayed Online Setup

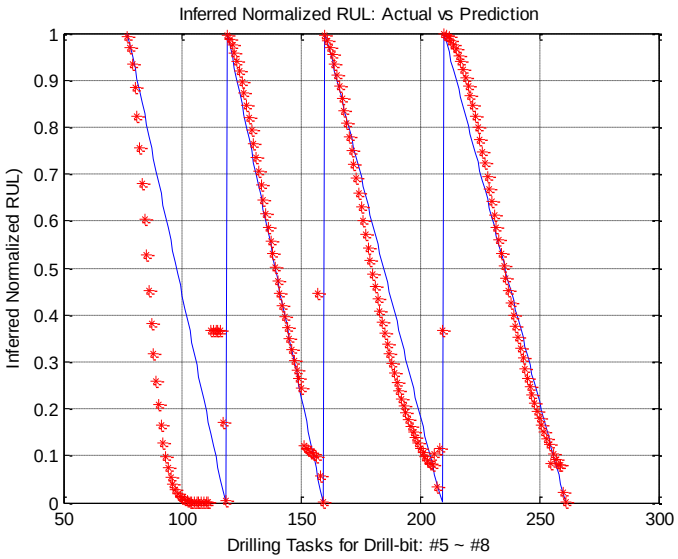


Figure 16: Normalized Inferred RUL vs Predicted RUL for Drill-bit #5~#8 in Delayed Online Setup

In Figure 15 and Figure 16, in comparison to Figure 12 and 13, one can see the effect when variable delays have been introduced. For the first drill-bit (drilling task 1 – 18 in Figure 15), predicted results are constant values because all clusters' local models' parameters are still default values and no update can be performed until the first drill-bit reached end of life (EOL). From drill-bit 2 and 3, prediction results started to improve slowly as each cluster's local model starts to receive additional updates. For the 5th drill-bit, existing clusters' local models were not able to anticipate the much extended usage time (42 drilling tasks) as existing clusters' local models were trained with data from drill-bit 1 – 4 that at most lasted 20 tasks. Nevertheless, prediction results for drill-bit 6 – 8 improved significantly as local models gradually adopted with data from drill-bit 5 – 7. The results for drill-bit 9 – 16 as shown in Figure 17 are largely similar to what is shown in Figure 14 since predictions are made from models that have already seen similar degradation profiles exhibited in drill-bit 1 – 4.

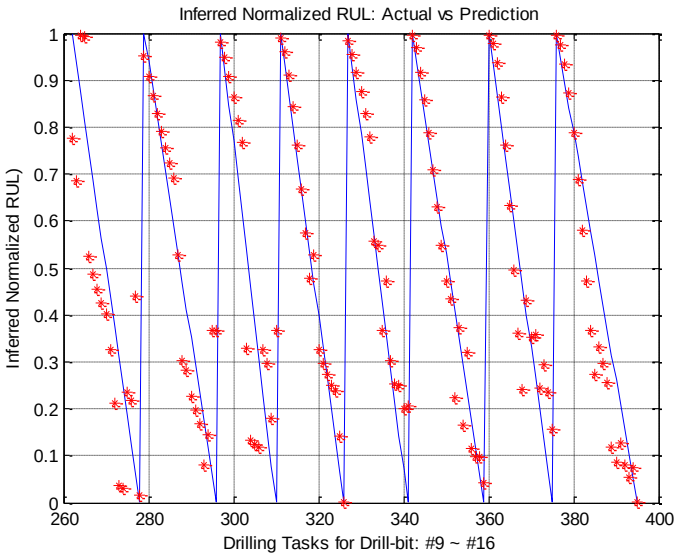


Figure 17: Normalized Inferred RUL vs Predicted RUL for Drill-bit #9~#16 in Delayed Online Setup

In Figure 18 and 19 (zoom-in), clusters with at least 3% (10 updates) are shown their individual local degradations models. Especially in Figure 4.18, if we were to setup the warning threshold for predicted RUL to be around 0.01, one can clearly see that degradation models can be divided into approximately 3 groups that will exceed that threshold at around 25 samples, 30 samples and 70+ samples. Note that while some of these degradation profiles are very similar, however, at the cluster level, each of them occupy different (may be overlapping) regions in the feature space. This is interpreted as the fact that some combinations of the torque and thrust force to end up having similar normalized RUL values at least at certain values ranges in the cumulative usage axis (cumulative drilling task).

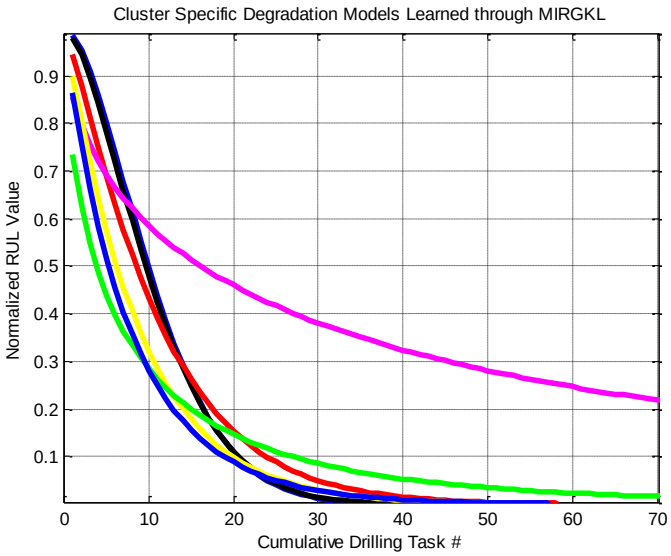


Figure 18: Learned cluster specific degradation models of clusters with 3% (10 points) of all data points

In Figure 18 and Figure 19 (zoom-in view), clusters with at least 3% (10 updates) are shown their individual local degradations models. Note that these clusters are not ordinal meaning that cluster #2 doesn't not necessary have a longer tail than cluster #1 and a shorter tail when compared to cluster #3. At the same token, while some of these degradation profiles are very similar, however, at the cluster level, each of them occupy different (may be overlapping) regions in the feature space. This is interpreted as the fact that some combinations of the torque and thrust force to end up having similar normalized RUL values at least at certain values ranges in the cumulative usage axis (cumulative number of drilling tasks performed).

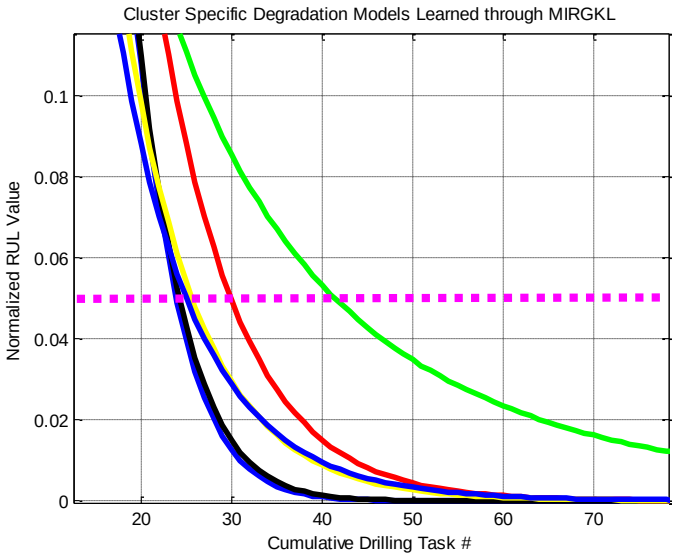


Figure 19: Learned cluster specific degradation models of clusters with 3% (10 points) of all data points (Zoom-in View)

In Figure 19, a zoomed in view of Figure 18, if we were to setup the warning threshold for predicted RUL to be around 0.05, one can clearly see that degradation models can be divided into approximately 3 groups that will exceed that threshold at around 25 samples, 30 samples and 70+ samples. The setup of such threshold requires special care as the value should be sufficient large (pessimistic) such that the warning will be in time before the actual EOL is reached (a couple of drills ahead). In other words, if the threshold is set to low (too optimistic) making warning to come off too late, one may a higher chance of producing parts with bad drills that need to be scrapped.

4.3 BRAKE WEAR MONITORING WITH MIRGKL-PLUS ALGORITHM

Brake and tire systems are critical systems that control the increments, maintenance and decrements of propulsive forces as commanded by the driving inputs. Both of these systems are subjects of wears that may be impacted by the vehicle setup, driving patterns and their operating conditions [WJ14, HM16, BY13 and JS15]. With MIRGKL-plus, we are taking a more data driven approach to the problem of brake pad wear as an example of a system with slow degradation dynamics. The benefits of taking such approach include:

1. Use only existing signals from the vehicle:

   Unlike physics based approach that commonly requires added high resolution sensory information; additional sensor instrumentation may not be needed in our experiment. For this particular experiment, we added only hardware to read and store existing signals off the CAN network.

2. Detailed system characterization is not needed.

   Typical model based approach requires certain details of the mathematical representation of the system which also involves parameters tuning as part of the nominal system characterization process. With a data driven method, less efforts are required on the modeling side of the system but more need to be put on to obtain a set of feature calculations that will have an impact on the degradation process.

4.3.1 Experiment Setup

The experiment involved 5 identical vehicles entered their individual employer sponsored management leases at different times. We chose such a setting due to the fact that these vehicles are maintained at a centralized location where periodic measurements of the brake pads are taken. Front left (FL), front right (FR), rear left (RL) and rear right (RR) brake pads measurements are taken separately with several repetitions and the mean values of FL, FR, RL and RR measurements are recorded during maintenance visits. Measured thickness will be the summation of the pad, backingplate and insulator. Example of such measurements is shown in Table 4.3

The 5 participants used these test vehicles are their personal cars with varying mileages, usages and driving road contents. The reason we are not performing the test in a more controlled driving environment is to be able to accumulate valuable data from real-world usages where multiple pad wear patterns may present themselves. Signals are collected at 50 HZ and include brake torque request (in newton-meter), actual brake torque applied (in newton-meter), brake on

and off signal, brake pedal percentage, vehicle speed (in kph) and odometer reading (in kilometer).

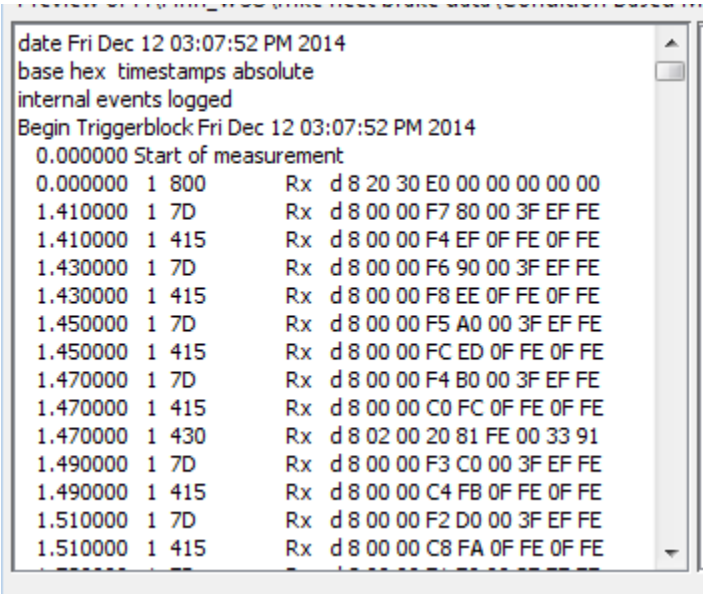


Figure 4.13: Example CAN messages recorded from the vehicle containing time stamp, message ID and hexadecimal encoded data

In Figure 4.13, an example of part of CAN message recording is shown. Message ID 7D represents the message for torque related signals, message ID 415 represents the messages related to vehicle speed and message ID 430 is associated with odometer reading. The brake pedal percentage signal (in message ID 165 and is not shown in Figure 4.13) has known consistency issues and as a result this signal is not included in subsequent analysis. Note that GPS information was not included as part of the signals being collected, as a result using such information to retrieve grade information to take out the effect of grade on the acceleration and deceleration is not possible.

Table 4.3: Example Brake Pad Thickness Measurement Table

	FDI	FDO	FPI	FPO	RDI	RDO	RPI	RPO
Top Right	17.87	17.66	18.00	17.65	15.57	15.58	15.92	15.75
Top Left	18.11	17.81	17.85	17.71	15.86	15.73	15.38	15.35
Bottom Left	18.01	17.92	17.71	17.87	15.69	15.87	15.46	15.51
Bottom Right	17.77	17.75	17.94	17.80	15.50	15.78	15.80	15.77
Average	17.940	17.785	17.875	17.758	15.655	15.740	15.640	15.595

Brake pad measurement for each individual participant takes place on a monthly basis. For each wheel (Front Driver Side or FD, Front Passenger side or FP, Rear Driver Side or RD and Rear Passenger Side or RP), 8 measurements are taken at top right, top left, bottom right and bottom left from both inside and outside facing angles. An example of a typical measurement record table is shown in Table 4.3 above. Overall, data collection process lasted approximately 6 months due to resource limitations of the participating parties.

4.3.2 Feature Extraction

In [FK12 and FK14], the granulation form (space partition) of information representation was treated as states in Markov Chain modeling where current and next states (a transition) is kept track of. In the case of brake data, we engage similar granulation scheme in the space spanned by speed and acceleration but without the tracking of transitions between defined granules. Instead, driving patterns were preserved with cumulative information.

For vehicle speed, 10 partitions are defined as the following:

0-5-10-15-25-35-45-65-85-105-120 (kilometers per hour)

For vehicle acceleration, 10 partitions are defined as the following:

0-0.05-0.1-0.2-0.3-0.5-0.75-1-1.25-1.5-1.75 (G or 9.81 meters / second)

For both speed and acceleration, an additional partition (11^{th} partition) is reserved for values exceeding either the min or max of defined regions. The 11^{th} partition is in place to capture the full picture of the usage profile. Such information is necessary to extrapolate in the usage space in the prediction phase. For example, for the same amount of 1000 km worth of driving, driver A may engage in braking actions for approximately 10 km while driver B may engage in braking actions for approximately 20 km due to variations in their driving content and style of operation. If we were to predict that the brake pad will wear out in the next 100 km, for driver A this translate to approximately 10000 km of driving distance remaining but for driver B this would translate to only 5000 km of driving distance remaining. In Table 4.4 below, we are showing the total driving distances, braking distances and percentage of braking in total driving distances to highlight the

significant variations in the data. Among the 5 participants, we saw the lowest percentage of braking in terms of driving distance to be around 3% for driver #4 and the highest of 7.32% in Driver #2. As a result, even if the brake wear rate was assumed to be the same in braking situations we expect driver #2 to wear out the brake pad much faster (at least 2 times faster) than Driver #4. If we were to present information associated to “Distance to Brake Pad Replacement”, this information need to be taken into account to reflect this particular type of variation at a per driver basis.

Table 4.4: Braking Distance VS Driving Distance Comparison among Participants

	Total Braking Distance (km)	Total Driving Distance (km)	Braking Distance %
Driver #1	876.73	14232.24	6.16%
Driver #2	551.20	7535.11	7.32%
Driver #3	475.96	8614.74	5.52%
Driver #4	450.08	14973.91	3.01%
Driver #5	326.14	7420.22	4.40%

Resulting patterns are granule-based braking pattern maps (GBBP map) that are re-calculated at a predefined interval (25 kilometers) and results from the previous intervals are stored. Mainly, total of 3 types of GBBP-maps are obtained.

- Distance travelled ($GBBP_{Dist}$)
- Deceleration ($GBBP_{Decel}$)
- Actual toque applied ($GBBP_{Torq}$)

Out of the 3 GBBP maps, we calculated two features that intuitively shall capture A. The preferences on the braking levels (exhibited and controlled by the driver) and B. Braking system’s performance degradation progression over time. We defined these two features as the following:

$$X_{decel\_pref}(t) = \int_{i=1}^{k} w(i,t) * \frac{GBBP_{Decel}(i,t)}{GBBP_{Dist}(i,t)} \quad (38)$$

$$X_{decel\_effi}(t) = \int_{i=1}^{k} w(i,t) * \frac{GBBP_{Dist}(i,t)}{GBBP_{Torq}(i,t)} \quad (39)$$

i denotes the unique the index of the granule, *w(i)* stands for the normalized weight obtained from normalizing $GBBP_{Dist}$ and k represents the total number of granules being taken into consideration. In our case, although total of 11*11 = 121 granules are defined only 10*10 = 100 of them will be included in the calculation. This is because the 11th partition in both speed and acceleration space are associated with "non-braking" driving activities and we assume no pad wear should take place in those situations.

Since brake wear is a slowly progressing phenomenon, we imposed a moving window of 6 or the equivalent of 6*25 = 150 kilometers as a way of filter and smooth out the information. X_{decel_pref} represents a quantifiable measure as to the driver's preference in terms of the level of deceleration during such event. For a more spirited or sporty type of braking behavior, the overall value of X_{decel_pref} will be larger relative to a value of X_{decel_pref}corresponds to a more comfortable or relaxed type of behavior. Driving contents and sudden extreme conditions (ie. Crash avoidance) may skew X_{decel_pref} to various extents and the effect of them should be minimized by the use of a moving window. X_{decel_effi} represents the efficiency for the overall braking action. Since the sampling rate is fixed, we use $GBBP_{Torq}$ instead of converting it to work done. When the braking system is operating high efficiency, X_{decel_effi} will have a *"low"* absolute value indicating unit torque achieved a *"short"* breaking distance; when X_{decel_effi} is *"high"* in absolute value, it represents a lower efficiency for the braking system as unit torque results in a *"longer"* braking distance. Note that in Figure 4.15 shown below the values take negative signs because the braking induced deceleration are negative in value to distinguish it between acceleration behaviors.

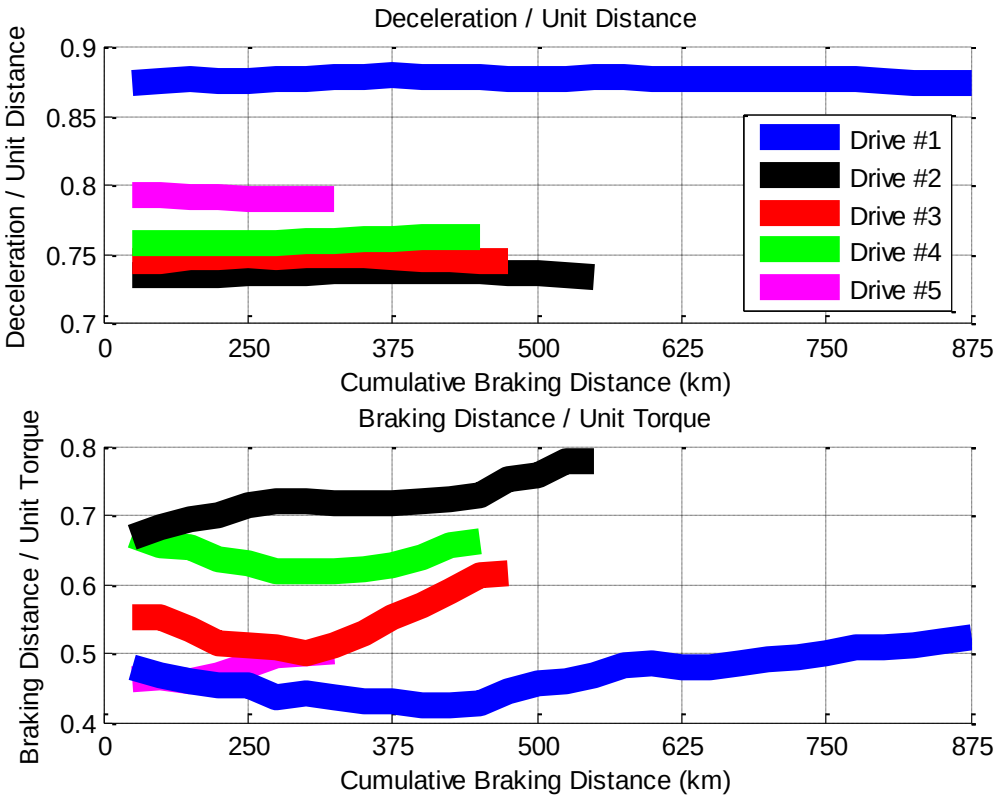


Figure 20: (upper) and Figure 4.15 (lower): X_{decel_pref} (upper) shows the deceleration preferences and X_{decel_effi} (lower) shows a long-term performance of the braking system.

In the upper plot of Figure 20, it is interesting that in the long run, despite differences in the driving content, the deceleration preferences for participating drivers remain relatively constant. This observation corresponds well with the intuition that an individual's preference in terms of the feel of a deceleration level regulated by a driver's control inputs is largely a personal driving attribute that does not change or evolve rapidly. On the other hand, in the lower plot of Figure 20, one should note that for all drivers there are clear trends of degradation in terms of the braking systems' performance. For driver #2 ~ #4, initial signs of improvement (downward trends) led to later degraded trend after the trend deflected at different cumulative brake mileage. For driver #1 and #5, their trends in the lower plot of Figure 20 are rather monotonic except for a somewhat extended flat region for driver #1.

4.3.3 Normalized Remaining Useful Life

Table 4.5: Nominal Front and Rear Geometrical Brake Thickness

	Nominal Value (mm)	Threshold Value (mm)
Geometrical Front Brake Thickness	18.350 (FBT_n)	15.175 (FBT_x)
Geometrical Rear Brake Thickness	16.150 (RBT_n)	12.975 (RBT_x)

From Table 4.5, one can see that the nominal values of thickness (thickness combining pad, backingplate and insulator) of front and rear brakes are different. However, the minimum threshold limits are the nominal values subtract by 1/8 inch (25.4 mm) according to specifications. With the knowledge of the above information, we can obtain normalized remaining useful life of both front and rear brakes as the following:

$$RUL_{Front\ Brake}(t) = \frac{X_{Front\ Brake}(t) - FBT_x}{FBT_n - FBT_x} \tag{40}$$

$$RUL_{Rear\ Brake}(t) = \frac{X_{Rear\ Brake}(t) - RBT_x}{RBT_n - RBT_x} \tag{41}$$

X represents the average of mean values for each brake as shown in Table 4.3. In (40, 41), the term t represents cumulative braking distance due to our assumption of no break wear during vehicle engagement in non-braking (idle, cruise and acceleration) activities. One should note that this inference of normalized RUL if different from (35, 36). This is because all participants started to have their data collected months after their lease started and the data collection stopped before any of the brake pads on all vehicles reached end of life (6-month worth of data). In other words, we do not have access to both near-brand-new or near-of-life types of data from this set of data.

In the experimental setup, Generic in-vehicle signals to obtain (38, 39) arrives much more frequently than the brake pad measurements to calculate (40, 41). This would be fairly typical for systems involving the tire and brake as precise information regarding thread and brake thickness (and wear) is made available through manual measurements. As a result, a procedure to generate artificial normalized RUL information is applied to increase the occurrence of learning

opportunities of cluster specific degradation models. Note that since the feature is extracted then filtered as described in 4.3.2 based on cumulative stats, interpolated normalized RUL makes it possible to enable additional trainings of local degradation models according to cluster assignment (and creation). This is equivalent of aligning occasionally obtain inferred RUL to the more frequently sampled and calculated feature set. One should be cautious of attempting to perform the task differently and attempt to map the clustering result to line up with the inferred RUL data. This is because interpolation performed to the overall cluster evolution process (assignment, creation and pruning are all possible outcomes) is not such a trivial task.

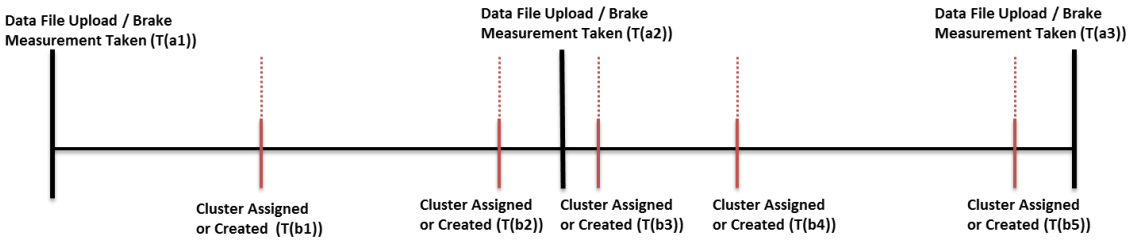


Figure 21 Example of timeline of events of data upload, clustering triggered by feature calculated from cumulative statistics and interpolated inferred RUL.

In Figure 21, T(a1) - T(a3) represents times when actual measurements of the brake pad thickness were taken. T(b1) – T(b5) are times when the clustering procedure took place due to availability of new features based on cumulative statistics. When clustering procedures are applied, possible outcomes include cluster assignment, cluster creation or pruning. The red dotted line at T(b1) - T(b5) represents interpolations is taken with mileage information recorded at those times with measured thickness and observed at T(a1) – T(a3).

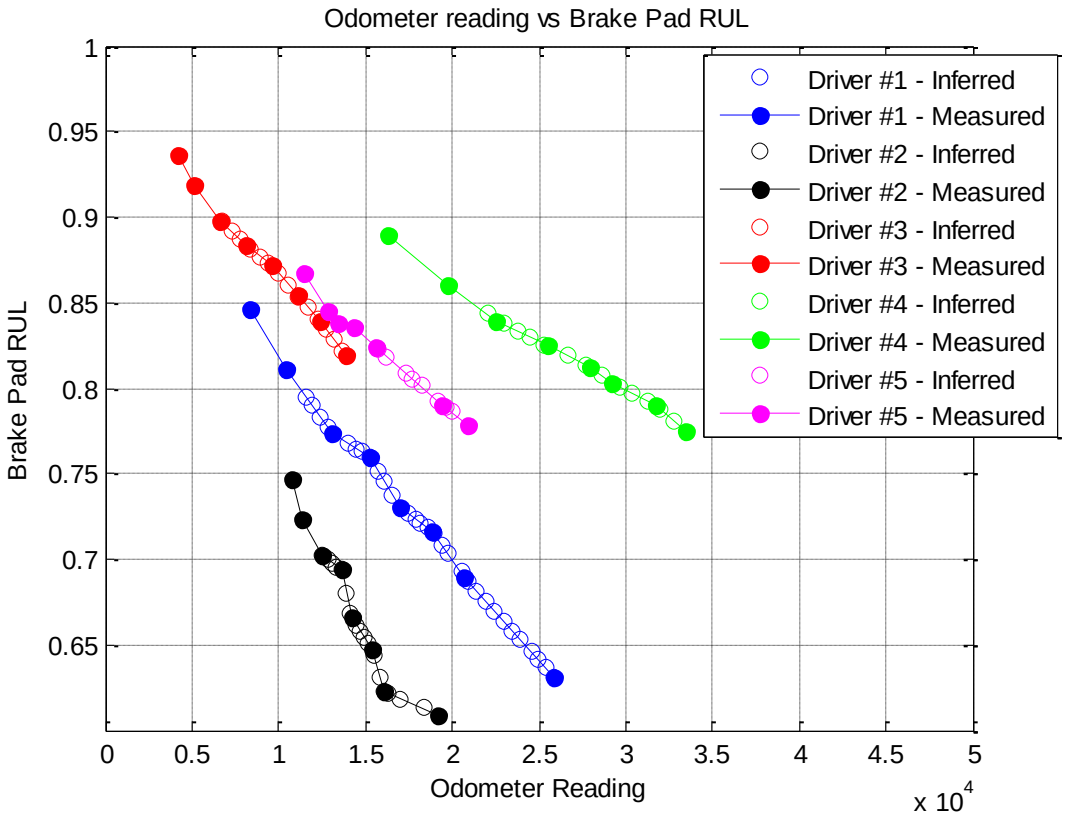


Figure 22: Measured (Solid Dots) and inferred (Unfilled Circles) RUL of Front Driver's Side Brake Pad Thickness

Figure 22 shows the RUL curves of the front driver side's brake pads for all 5 participants. Due to the asynchronous nature between manual measurements and how features were calculated, we show the RUL calculated from actual measurements as solid dots and inferred RUL as unfilled circles and color coded all 5 participants with different colors. We noted that the rate of decrement in RUL for Driver #2 is 1.63% per 1000 miles driven vs 0.67% per 1000 miles driven by Driver #4. This observation can be partially explained by Driver #2's high brake distance to total driving distance ratio and the same ratio for Driver #4 turned out to be the lowest. Such relationship holds well for both front brakes but much less so for the rear brakes. Evaluation of the normalized brake pad wear in terms of cumulative braking distance as shown in Figure 4.18-19 aligns with the notion that wear activities should be minimal in non-braking situations. Note that when compared between Figure 23 and Figure 24, one can clearly see that with cumulative

braking distance as the x-axis aligns data from all participants better. It also better demonstrates the notion the degradation patterns from all 5 participants may be summarized by a few prototypes (Figure 23 and Figure 24) as to all have to be individually treated (as shown in Figure 22). The main reason for cumulative braking distance being a more effective horizon for presenting brake wear data is due to the fact that under normal vehicle usages when the brake system is assembled properly the wear to the brake pads shall be minimal.

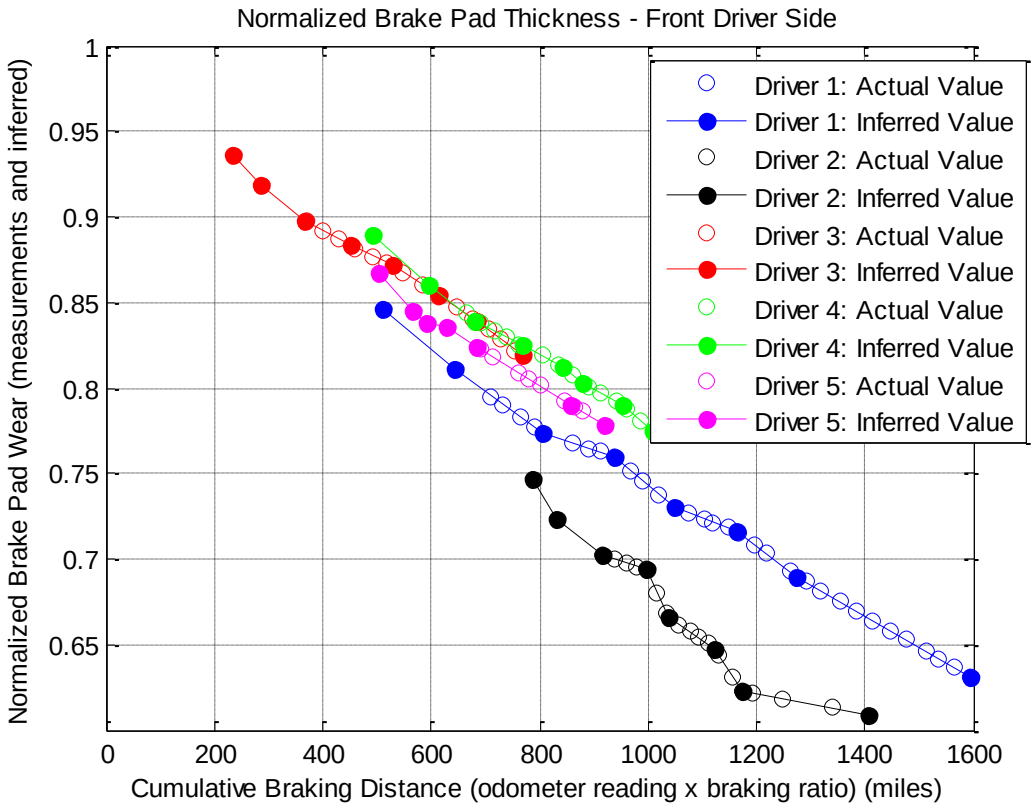


Figure 23: Measured (Solid Dots) and inferred (Unfilled Circles) RUL of Front Driver's Side Brake Pad Thickness against Cumulative Braking Distance as the X-axis

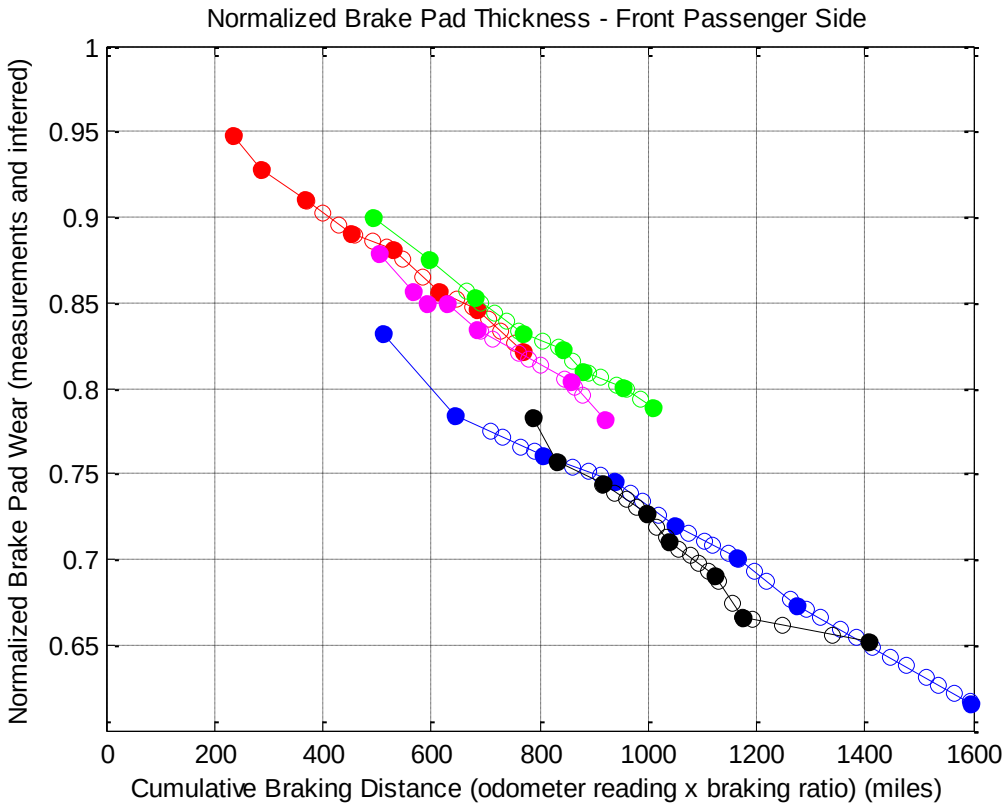


Figure 24: Measured (Solid Dots) and inferred (Unfilled Circles) RUL of Front Driver's Side Brake Pad Thickness against Cumulative Braking Distance as the X-axis

Essentially, we are taking the effect of different ratios of braking to driving exhibited by different drivers as depicted in Table 4.4. If we were to build models with information as shown in Figure 23 and 24, to translated the horizon where predictions are made (cumulative braking distance) into actual driving distance ratios in Table 4.4 need to applied for the conversion.

4.3.4 Results

Initially, we set the probability was set at 0.95 and for 2-dimension data this corresponds to a threshold value for Chi square statistics is 5.9915. With the features as described in 4.3.2 which are also shown in Figure 4.20, it produced an unexpected result in terms of the clusters that were identified. As show in Table 4.6, in this setting the identified the 5 clusters corresponds to the 5 participating driver. One thing to note is that since we arranged the data by the time of measurement taken as they correspond to the time data become available to the system and

subjected to processing. Since driver numbers were assigned by each driver's name in alphabetical order and cluster numbers are assigned at the time when it is created, in Table 4.6 cluster #4 matches driver #2, cluster #2 matches driver #4 and driver #1, #3 and #5 matches cluster #1, #3 and #5 respectively.

Table 4.6: Log of data sequence, cluster assignment and driver identification number with probability threshold set at 0.95

Data #	Cluster #	Driver #	Data #	Cluster #	Driver #
1	1	1	19	1	1
2	1	1	20	3	3
3	1	1	21	3	3
4	1	1	22	3	3
5	2	4	23	1	1
6	3	3	24	1	1
7	3	3	25	1	1
8	3	3	26	1	1
9	1	1	27	2	4
10	1	1	28	4	2
11	1	1	29	4	2
12	2	4	30	4	2
13	2	4	31	4	2
14	2	4	32	4	2
15	2	4	33	5	5
16	1	1	34	5	5
17	1	1	35	5	5
18	1	1			

In the ideal setting where a large number of the same type of vehicles' data are being collected, we expect the model to be able to summarize drivers with similar patterns into the approximately the same clusters. In other words, the amount prototypical patterns as represented by identified clusters should be much smaller than the total number of individuals that formed the crowd or participating drivers. As drivers belong to the same cluster incrementally contribute to the cluster specific model, the model can be trained at the accelerated pace with respect to a portion of the population (belongs to the same cluster) for predictive purposes. To test this notion, even though we only have total of 5 participants, we tightened the probability threshold to extreme high value of 0.999 to limit its criteria to create new clusters. In Table 4.7, one can see that in this setting driver #2, #4 and #5 are grouped as one cluster in cluster # 2 and driver #1 and driver #3 remained to have clusters of their own.

Table 4.7: Log of data sequence, cluster assignment and driver identification number with probability threshold set at 0.999

Data #	Cluster #	Driver #	Data #	Cluster #	Driver #
1	1	1	19	1	1
2	1	1	20	3	3
3	1	1	21	3	3
4	1	1	22	3	3
5	2	4	23	1	1
6	3	3	24	1	1
7	3	3	25	1	1
8	3	3	26	1	1
9	1	1	27	2	4
10	1	1	28	2	2
11	1	1	29	2	2
12	2	4	30	2	2
13	2	4	31	2	2
14	2	4	32	2	2
15	2	4	33	2	5
16	1	1	34	2	5
17	1	1	35	2	5
18	1	1			

In Table 4.8, we summarize the performance in terms of mean square error for each driver, each driver's individual brakes and the overall value of all drivers. When studied together with Table 4.9 where the probability threshold was set at 0.95, one can see that with a tighter probability threshold, while less clusters (total of 3) were created, the algorithm shows good capability to generalize information and produced a better MSE over 26% lower than the setting that created more clusters (total of 5).

Table 4.8: Mean square error for each driver, each driver's individual brakes and overall values with probability threshold at 0.999

Probably Threshold: 0.999				
Driver#1 FD	Driver#1 FP	Driver#1 RD	Driver#1 RP	Driver#1 Overall
0.0062	0.0056	0.0059	0.0052	0.0057
Driver#2 FD	Driver#2 FP	Driver#2 RD	Driver#2 RP	Driver#2 Overall
0.0027	0.0015	0.002	0.0031	0.0023
Driver#3 FD	Driver#3 FP	Driver#3 RD	Driver#3 RP	Driver#3 Overall
0.0202	0.0211	0.0179	0.0171	0.0191
Driver#4 FD	Driver#4 FP	Driver#4 RD	Driver#4 RP	Driver#4 Overall
0.018	0.0188	0.0158	0.0167	0.0173
Driver#5 FD	Driver#5 FP	Driver#5 RD	Driver#5 RP	Driver#5 Overall
0.00048	0.0003	0.00063	0.00057	0.000495
				Overall
				0.0086

Table 4.9: Mean square error for each driver, each driver's individual brakes and overall values with probability threshold at 0.999

Probably Threshold: 0.95				
Driver#1 FD	Driver#1 FP	Driver#1 RD	Driver#1 RP	Driver#1 Overall
0.0062	0.0056	0.0059	0.0052	0.0057
Driver#2 FD	Driver#2 FP	Driver#2 RD	Driver#2 RP	Driver#2 Overall
0.0067	0.0084	0.0059	0.0058	0.0067
Driver#3 FD	Driver#3 FP	Driver#3 RD	Driver#3 RP	Driver#3 Overall
0.0202	0.0211	0.0177	0.0145	0.0184
Driver#4 FD	Driver#4 FP	Driver#4 RD	Driver#4 RP	Driver#4 Overall
0.0175	0.0186	0.0154	0.0164	0.017
Driver#5 FD	Driver#5 FP	Driver#5 RD	Driver#5 RP	Driver#5 Overall
0.026	0.0272	0.0219	0.023	0.0245
				Overall
				0.0117

Finally, Figure 4.21 – Figure 4.25 show from Driver #1 – Driver #5 their 4 brakes' (front driver side, front passenger side, rear driver side and rear passenger side) inferred normalized brake pad thickness versus the predicted value in a 1-step ahead prediction setup. These results are generated with the probability threshold set to be 0.999.

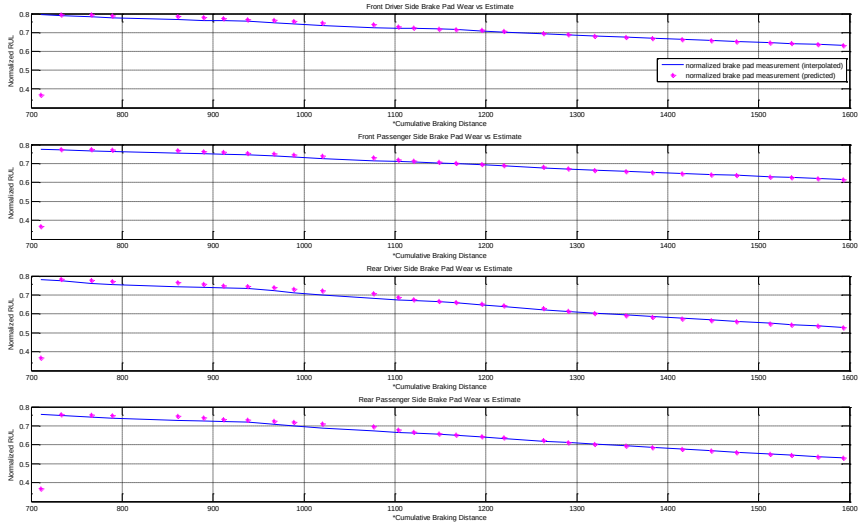


Figure 25: Driver #1, Inferred Normalized Brake Wear (blue) vs Predicted Brake Wear. Top – Bottom: Front Driver Side, Front Passenger Side, Rear Driver Side and Rear Passenger Side

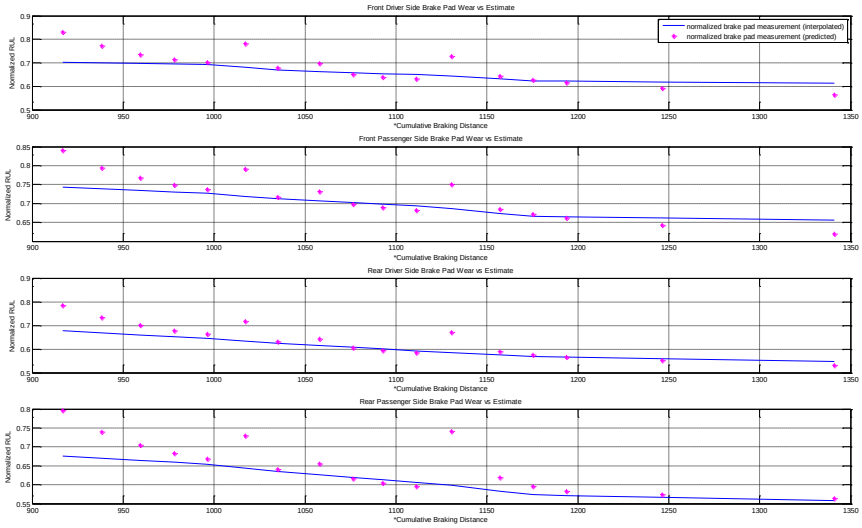


Figure 26: Driver #2, Inferred Normalized Brake Wear (blue) vs Predicted Brake Wear. Top – Bottom: Front Driver Side, Front Passenger Side, Rear Driver Side and Rear Passenger Side

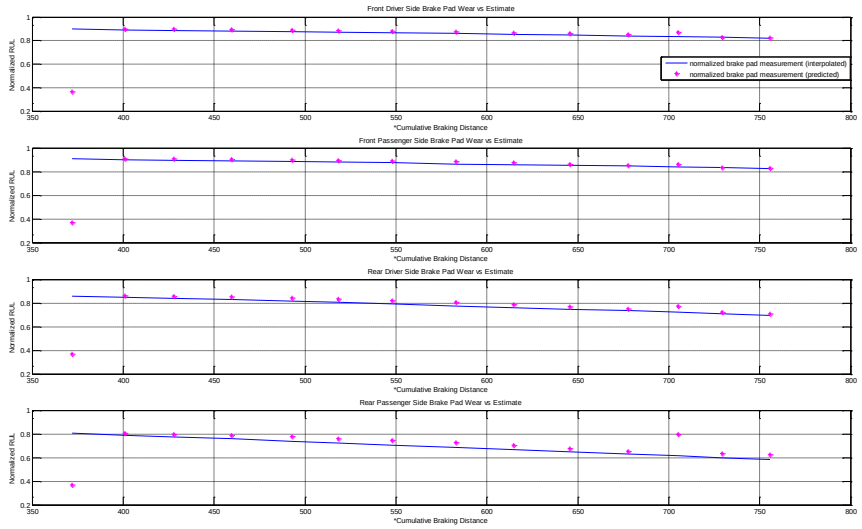


Figure 27: Driver #3, Inferred Normalized Brake Wear (blue) vs Predicted Brake Wear. Top – Bottom: Front Driver Side, Front Passenger Side, Rear Driver Side and Rear Passenger Side

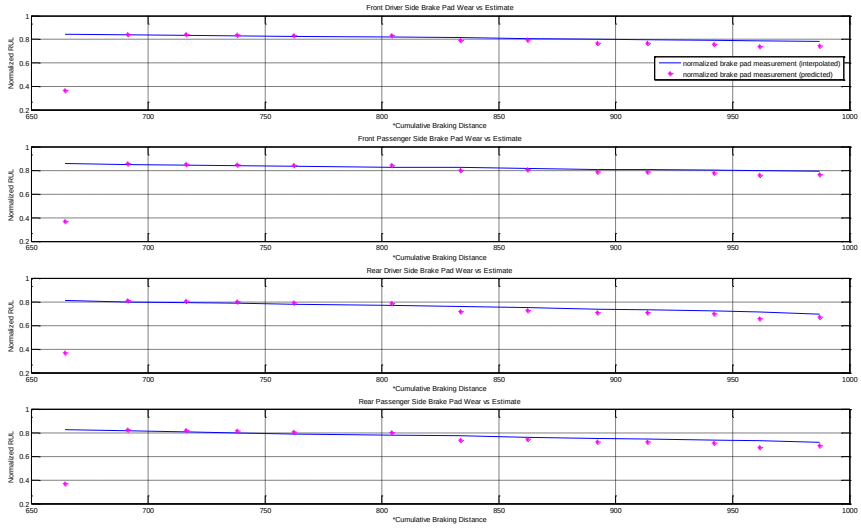


Figure 28: Driver #4, Inferred Normalized Brake Wear (blue) vs Predicted Brake Wear. Top – Bottom: Front Driver Side, Front Passenger Side, Rear Driver Side and Rear Passenger Side

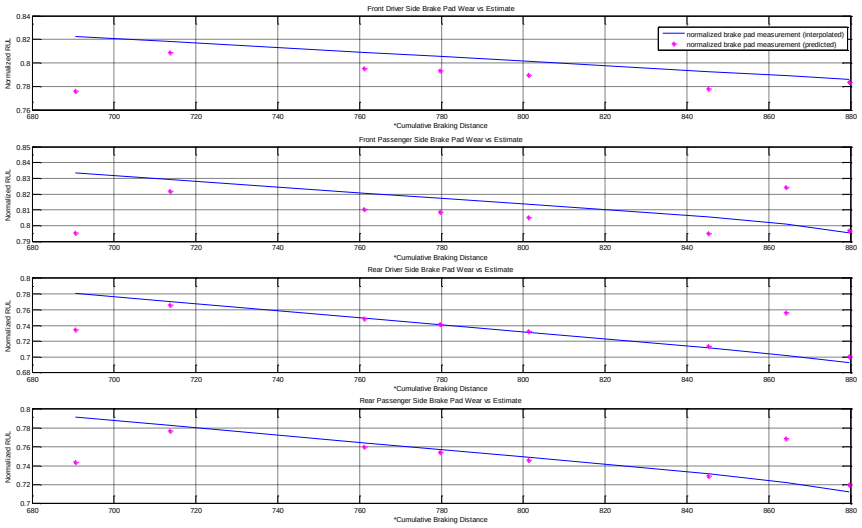


Figure 29: Driver #5, Inferred Normalized Brake Wear (blue) vs Predicted Brake Wear. Top – Bottom: Front Driver Side, Front Passenger Side, Rear Driver Side and Rear Passenger Side

4.5 CONCLUSION AND NEXT STEPS

In this chapter, we showed examples prognostics problems, one experimental and one real-world, that present great challenges in practice. Difficulties to address similar types of problems can be summarized as the following. 1. Physical representation of the system with high enough precision for prognostics purpose is difficult to obtain. 2. Large piece to piece variations cannot

be well explained or tracked once experiments have been completed. 3. There is more than one underlying degradation modes. 4. Limited observability to the degradation signal. As a result, we applied proposed MIRGKL embedded with local functions to tackle abovementioned problem with very promising results.

From both examples, we showed the applicability of evolving clustering algorithm to systems of the same type under somewhat different usage patterns can be effectively separated into groups (clusters). We have also proposed the use of limited degradation information that arrives at a different rate when compared to other more generic signals used for clustering through use of linear interpolation and normalization if the operation limits of the crucial system parameter is known. Aligned normalized degradation through inference allows cluster specific local models to be updated along with the update procedure of the overall clustering process. In the brake wear example, one thing to note is that the degradation (inferred normalized brake thickness) has to be presented in a meaningful horizon (cumulative braking distance) in order to observe and obtain meaningful results. Even though some drivers may exhibit similar rate of wear in the horizon of cumulative braking distance, the final odometer where the limit may differ significantly due to their differences in braking / driving ratios which is an aggregate of individual driving content and behavior.

Future research direction including the further use of clustering outcomes with novelty detection methods which is a combination of probability theory (i.e. Markov Chain), information theory (Kullback-Leibler divergence) and statistical analysis. Also, we would like to further validate the results from the brake data with the actual end of life information which is not available at the moment.

CHAPTER 5: DATA EVOLUTION BASED DIAGNOSTICS AND PROGNOSTICS WITH EVOLVING CLUSTERS

Conventional approach to system diagnostics and prognostics problem is based on the quantifiable comparisons between nominal and current observation of some defined attributes. These attributes, features generally, are defined as a set of values and are results of physical understanding and approximation of underlying mechanism, operational inputs including its operational mode, conditions, noise and inputs that will have an impact on the system behavior and subsequent failure modes. Further complicating the issue is the underlying aging or functional degradation of the system, which are generally hard to quantify in an absolute manner making such information mostly not observable. All these makes providing robust diagnostics and prognostics state of a system a difficult problem. As a result, providing such health information of a system typically requires the OEM to put in additional efforts early on in the design and development phase.

In this paper, we propose the use of real-time evolving algorithm as the engine for continuous monitoring of system health and prediction of potential upcoming faults. At the kernel of the system is a generic evolving data clustering model that is capable of adaptation in terms of both its parameters and structure. Since it is purely data driven and unsupervisory in nature, the application of method of this type requires less problem specific details and domain knowledge of the particular system or machine for which diagnostics and prognostics functions have to be developed. As a result, it is suitable to be realized as a stand-alone event driven software agent that may interface data stream either directly in onboard applications or through severs when data is transmitted to and stored with given infrastructure, sensing network and connectivity setup.

The feature set serves as the primary input to the algorithm may consist of various measured and calculated variables capturing of the state of the system, the environment the system is being operated in and control inputs. For a machine with rotating parts exhibits vibrational signatures,

such variables may include calculations that define vibration characteristics such as time and frequency domain based features, statistically based features and process parameters such as the rotating parts measured speed in terms of RPM and loading the machine is being operated on … etc. Dimension reduction is considered to be option but may be most necessary for the purpose of visualization as well as part of the de-noise procedure. Subsequently the feature set is processed through the evolving clustering algorithm where patterns are preserved and refined parametrically and structurally as needed.

Multiple strategies exploiting identified clusters as states using probability theory (i.e. Markov Chain), information theory (Kullback-Leibler divergence) and statistical analysis (i.e. statistical process control) will be proposed where their association to diagnostics (health state determination) information and prognostics (health state or remaining useful life prediction) information can be established for predictive uses. Majority the abovementioned techniques are inspired by how one differentiate between normal vs cancerous cells in biological and medicinal studies [DR08 and NCI15]. Pattern of divisions, sizes, shapes, organizational arrangement, lack of specialization in its feature and crispness of the boundaries of the cells are important aspects to consider when the existence of abnormal cells is suspected.

The rest of the paper is organized as the following. We will only briefly summarize concepts applying evolving clustering algorithm (ECA) as the mean to monitor evolution in data for diagnostics and prognostics purposes. Some of the concepts will be illustrated with examples that were discussed in prior papers. We will then this chapter with comments and ideas for future research.

5.1 DATA EVOLUTION BASED DIAGNOSTICS AND PROGNOSTICS

5.1.1 Pattern of Cluster Creations and Cluster Health Quantification

In Chapter 3, we have discussed an evolving clustering method with a newly proposed measurement in the degree of similarity. Equipped with similar methodologies designed to deal with data with indefinite length, identified clusters are the representation of groups that together

defines the whole operational space of system with various degree of overlap. In [FT06a, FT06b, FCT10], each cluster incrementally accumulates information related to their health throughout the clustering process, namely the age factor and the relative population size factor. Unlike clusters with significant age and sizable population representing repetitive stable patterns, clusters with low health factor (product of age and relative population size factor) exemplify the case of patterns with lack of specialization during system operation, which is a symptom for abnormal cells. As a result, projection of the active cluster to move inside of a region defined by such type of cluster is considered to be indicative of an impending failure.

Another notion proposed was to monitor the rate of new cluster creations through the use of an Exponential Weighted Moving Average (EWMA) Statistical Process Control (SPC) chart. This proposed criterion as a symptom of a drastic system failure corresponds to the phenomenon of rapid cell divisions during certain developmental stages of cancerous cells. In addition, SPC is also applicable in the original or normalized individual feature space to determine if one or multiple of them may exhibit patterns of instability (out of EWMA control limits). It should be noted that for certain features with specific physical meanings, it is possible that only the upper or lower control limit makes sense but not both of them.

Aforementioned method monitors the evolving clustering process as a whole (macro scale or unconditional) with EWMA SPC and associates out of control instances as precursors to drastic changes in the data stream. One potential extension will be to model the pattern of new cluster creations specifically for each existing cluster as an event arrival process and pay close attention if such rate is to increase significantly over time. Traditionally, the arrival process can be approximated as a Poisson process as the following:

$$P(k\ events\ in\ interval) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (42)$$

For each cluster i, the Poisson distribution parameter λ will be separately identified as λ_i. The mechanism of using such model may involve direct monitor of λ_i or to evaluate the active

cluster to see if in the next time intervals, the number of new cluster creation in some time interval will exceeded some predefined threshold. An alternative to look at only the active cluster will be to look at all existing or a group of clusters with relative high importance together to conduct similar evaluation to predict if system instability (creation of excessive clusters) is anticipated in the near future.

$$P^*(k\ events\ in\ interval) = \sum_{i=1}^{N} \frac{{\lambda_i}^k e^{-\lambda_i}}{k!} \tag{43}$$

In the equation above, the number of event k need to be determined as a threshold value and i = 1 ~ N represent a subset (or full set) of existing clusters selected by some predefined criteria such as the age and relative population size factors as mentioned in 5.1.1.

5.1.2 Cluster Transitions and Change in Patterns of Transitions

With the capability of partitioning data stream with indefinite length into individual ones (clusters), it becomes possible to model the transitions between identified clusters as states synonymic to the definition of a Markov model with a transition probability matrix P where column represents the originating state and the row represents the destination state:

$$P = \begin{bmatrix} p_{1,1} & \cdots & p_{1,j} \\ \vdots & \ddots & \vdots \\ pi,1 & \cdots & p_{i,j} \end{bmatrix} \tag{44}$$

From observations of the clustering process, the learning of the transition matrix P can be F with the same size as P:

$$F = \begin{bmatrix} f_{1,1} & \cdots & f_{1,j} \\ \vdots & \ddots & \vdots \\ fi,1 & \cdots & f_{i,j} \end{bmatrix} \tag{45}$$

A. At time t-1, If the data was assigned to cluster i or if cluster i was created and at current time of t the data is assigned to cluster j or if cluster j is created. $f_{i,j}(t) = f_{i,j}(t-1)+1$.

B. At time t-1, If the data was assigned to cluster i or if cluster i was created and at current time of t the data is assigned to cluster j or if cluster j is created. Following updates should take place:

$$f_{i,j}(t) = \alpha * f_{i,j}(t-1) + (1-\alpha) * 1 \quad (46)$$

$$f_{i,k}(t) = \alpha * f_{i,k}(t-1),\ k \neq j \quad (47)$$

Transition probability matrix P is obtained by dividing each row in F by the sum of a that row. Both inference method A and B originated from counting the occurrence of a specific transition through the clustering process. They differ in the sense that inference method A is batch in nature and after significant amount of data has been processed, it will be hardly changing with subsequent data. On the other hand, inference method B is equivalent of imposing a predefined moving window associated with the learning rate α.

The use of updating procedures in inference method B need to be carefully taken. Given a previous state (previous cluster assignment) and current state (current cluster assignment), update should only take place for the row that matches the previous (originating) state. For that particular row, only one element that matches the current state (destination) will experience an increment in value in (46) while the rest of the element shall all experience a decrement in value (47).

For a diagnostics task of determining whether current clustering assignment is indicative of a faulty state, a few avenues may be taken. First of all, as proposed in [FT06a, FT06b], cluster health with unlabeled data stream may be inferred from the aggregate of factors that are related to an individual cluster's relative utilization rate and age. However, there are other cluster health related factors which will be discussed in 5.1.3. Secondly, as mentioned in [AT08] latent state in HMM associated with inferred fault label may also assist in the identification of cluster assignment patterns with an impending faulty state. Thirdly, in case when sporadic health labels (including fault) are available, they may either be treated as part of the feature set such that the knowledge is incorporated in the clustering process or use it to create an additional health related factor.

Such information may be learned as a distribution to indicate the propensity of a fault when associated cluster is active. With cluster specific health information, the cluster transition matrix may then be applied to determine with some predefined horizon if the system has a high predicted probability of transition to a less desired cluster. Evaluations may also be performed with the steady state cluster transition matrix to access if the system's tendency or average probability to be at less desired state(s) has exceeded a predefined threshold or exhibited an uptick trend indicating the system as a whole may be showing signs of deteriorating health.

It should be noted that some of the techniques mentioned in 5.1.1 regarding application of SPC at the micro scale (overall clustering process) is also applicable at the cluster level. One example would be to construct EWMA SPC for in the feature space for each individual cluster or to monitor the increment of 2^{nd} level clusters if hierarchical clustering is taking place. This type of monitoring procedure is conducted at the micro scale (cluster specific) and may provide additional insights associated with the health status and stability that is more conditional as they are only relevant to one particular cluster.

5.1.3 Cluster Utilization, Homogeneity and Crispness of Boundaries

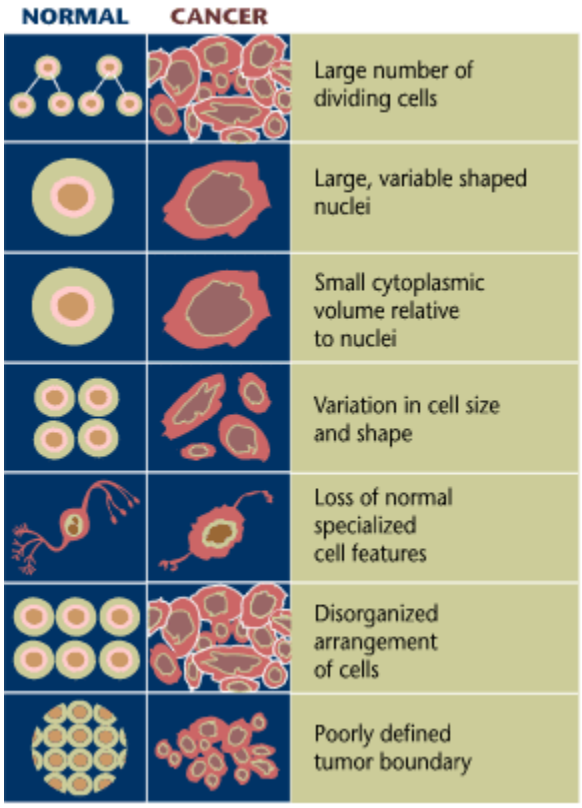


Figure 30: Differences between normal vs. cancer cells [VW15]

Due to the fact that evolving clustering may be applied to data completely unsupervised, the relevancy of the notion of a cell's loss of specialization to the characterization of a cluster is not straightforward. Instead, the activation frequency or utilization of a cluster can be readily calculated. In its simplest form, how many times a specific cluster has been activated (creation and assignment of a new data) are directly observable therefore can be counted to establish the utilization vector with the same amount of elements as the total number of clusters, N:

$$U = [u_1, \dots, u_n] \tag{48}$$

While level of utilization can be quantified in an absolute sense, it can also be performed in a relative fashion with a moving horizon of a predefined length. EWMA associated with a moving window may perform such task with the same recursive formula as shown in (46, 47). When this type of implementation is in place, vector *U* becomes a vector of relative activation frequencies of

existing clusters in a moving window with the window size associated with the learning rate α. Vector *U* may go through a normalization to produce a vector, $\widetilde{U}$,such that the summation of all its element become 1. The normalization step's only purpose is to make such information compatible and more convenient during further processing (information aggregation). Utilization information can be useful as a cluster specific attribute related to its health or it can also be applied at the macro level to see if the ratio between data being assigned to highly utilized cluster(s) versus rarely utilized cluster(s) is showing a pattern of system deterioration.

Homogeneity and crispness of separation between clusters are two aspects that are rarely discussed. Assuming identified clusters are approximately normal, the matrix form of the Kullback–Leibler divergence is expressed as the following:

$$D_{KL}(\aleph_0||\aleph_1) = \frac{1}{2}(tr(\sum_1^{-1}\sum_0) + (\mu_1 - \mu_0)^T\sum_1^{-1}(\mu_1 - \mu_0) - k + \ln(\frac{det\sum_1}{det\sum_0})) \quad (49)$$

Note that the 2nd term is the Mahalanobis distance which is a distance measure originated from the difference in the centroid between populations scaled by the base population's norm inducing function. As a result, when such term is omitted from (49), it becomes a measure focused on shape and orientation information between populations.

$$D_H(\aleph_0||\aleph_1) = a * tr(tr(\sum_1^{-1}\sum_0) + b * \ln(\frac{det\sum_1}{det\sum_0})) + c \quad (50)$$

(50) includes part of (49) where cluster pairs relative shape similarities are calculated. The addition of constant a-c are subjected to tuning to prevent the outcome becomes negative.

Assuming there are total of N clusters resulting N*(N-1)/2 pairs, we may define a homogeneity (variation in cluster size and shape) measure among this group of clusters as the following:

$$H = \sum_{i=1}^{N*(N-1)} D_H(i)\,/(N*(N-1)) \quad (51)$$

i is an index of a specific pair of existing clusters where pair (A, B) is different from pair (B, A). If we apply mutual information based calculation to (49):

$$D_{q,p}(\aleph_q||\aleph_p) = \frac{1}{2}D_H(\aleph_q||\aleph_M) + \frac{1}{2}D_H(\aleph_p||\aleph_M), q = [1, C], q \neq p \quad (52)$$

Where $\aleph_M = \frac{1}{2}\aleph_q + \frac{1}{2}\aleph_p$

(52) makes originally asymmetric formulation (50) symmetric therefore the total number of cluster pairs, N*(N-1), will be effectively be reduced by half. This is because when (53) is applied, the cluster pairs (A, B) and (B, A) will produce the same value. As a result, (50) will effectively become:

$$H = \sum_{i=1}^{N*(N-1)/2} D_H(i) \,/(N * (N-1)/2) \quad (53)$$

Note that when (53) is applied, an additional condition of q > p should be applied in using (52) where q and p as index of existing clusters. This is condition is to ensure only one of the cluster pairs (A, B) and (B, A) will be included in the calculation of (53).

Referencing to (49), the 2nd term, the Mahalanobis distance is the only term associated with pair-wise distance between clusters. As a result, we propose the use of the Mahalanobis distance:

$$D_{MD} = (\mu_1 - \mu_0)^T \sum_1^{-1} (\mu_1 - \mu_0) \quad (54)$$

We define clustering crispness and boundary factor as the following:

$$CB = \sum_{i=1}^{N*(N-1)/2} D_{Mutual\ MD}(i) \,/(N * (N-1)/2) \quad (55)$$

One can see that (55) is simply the average degree of pairwise separations among exiting clusters. The rationale is that the larger value of CB is, the crispness and more clearly defined boundary will be in the average sense since all existing clusters will be well separated. Since we mainly rely on the centroid and covariance pair to define any existing clusters, the notion of dissimilarity between nucleus and cell border (in volume or shape) is not really applicable because the centroid itself does not really possess the shape attribute. Other factors to consider including and the degree of disorganization or arrangement and more exact quantification the level of crispness of individual clusters will not be discussed here but be considered as part of the future research.

5.1.4 Novelty Detection at the Macro and Micro Level

In 5.1.1 – 5.1.3, we have summarized some prior work and also discussed additional new ideas as to how evolving modeling methods as a framework of to perform diagnostics and prognostics tasks. Such an integrated procedure is realized through novelty detection techniques involving supervised and unsupervised online learning with strong emphasizes on machine learning, classification, inference of degradation and decision making. Aforementioned learning of state transition and characterized are applicable both at the macro (overall clustering process) and micro (cluster specific or conditional information) levels to expand the width and depth of pattern spaces novel (emerging or out of norm) piece of information can be derived such their direct or indirect implications to impending or immediate system faults can be further studied.

As a result, evolving clustering as an integrated novelty detection framework for diagnostics and prognostics purposes addresses some of the most challenging issues in the domain such as A. The ability to learn with every new piece of information without excessive buffering reprocess old data. B. Incrementally adapt the model parametrically and structurally. C. As a generic information processing and modeling engine applicable to wide variety of systems with low requirement on problem specific knowledge.

5.2 CONCLUSION AND NEXT STEPS

In this chapter, we discussed different ideas related to the monitoring of an ongoing clustering process. At the macro level, clustering evolution can be tracked through cluster complexity (total number of clusters), the arrival rate of new clusters, the ratio of activation frequencies between health versus non-health clusters … etc. We have also proposed additional biologically inspired factors associated with utilization, homogeneity and crispness of boundaries. With these newly proposed quantifiable attributes, they may provide additional insight to the relationship between an evolving clustering process and the status of the system from which the data stream is originated.

When cluster specific health indicators are defined, it becomes possible to model the transitions between clusters as states transition with a Markov model. This type of model preserves both conditional and unconditional information as to the propensity of the system's likelihood moving into undesired conditions in the future hence lend themselves conveniently as tools to provide prognostics functionalities. As an alternative, monitoring of the learned transition probability matrix is also possible to gauge if system deterioration has occurred. As part of the cluster assignment process, additional cluster specific health indicators also provides additional avenue and opportunity to determine the current status of the system (diagnostics information).

Our future research will involve the application and validation of proposed biologically inspired overall (clustering) health condition indicators and cluster specific health indicators with real-world data.

CHAPTER 6: PUBLICATIONS, CONCLUSIONS AND NEXT STEPS

6.1 PUBLICATIONS

Scholarly journal and conference publications that resulted from the dissertation research are listed below:

Journal Papers

- D.P. Filev, R.B. Chinnam., F. Tseng and P. Baruah, 2010. “An industrial strength novelty detection framework for autonomous equipment monitoring and diagnostics.” IEEE *Transactions on Industrial Informatics*, 6(4), pp.767-779.
- F. Tseng, D.P. Filev, R. B. Chinnam, “A Mutual Information Based Online Evolving Clustering Approach and Its Applications”, *Evolving Systems, Springer* (in review, submitted)
- F. Tseng, D.P. Filev, I. Makki, J. Lu, R. B. Chinnam, “An Evolving Clustering Based Approach for Brake System Prognostics”, (to be submitted)
- F. Tseng, D.P. Filev, I. Makki, R. B. Chinnam, “Novelty Detection with a Data Evolution Model and Its Applications in Condition Based Monitoring”, (to be submitted)
- F. Tseng, D.P. Filev, I. Makki, R. B. Chinnam, “An Online Evolving Relevance Based Model for Trip Prognosis”, (to be submitted)

Conference Papers

- D.P. Filev and F. Tseng, "Novelty detection based machine health prognostics." 2006 *International Symposium on Evolving Fuzzy Systems*, pp. 193-199. IEEE, 2006.
- D.P. Filev and F. Tseng, “Real time novelty detection modeling for machine health prognostics.” 2006, In *NAFIPS 2006 Annual Meeting of the North American Fuzzy Information Processing Society*.
- A. Kumar, F. Tseng, Y. Guo and R.B. Chinnam, “Hidden-Markov model based sequential clustering for autonomous diagnostics.” 2008, *IEEE International Joint*

*Conference on Neural Networ*ks (IEEE World Congress on Computational Intelligence) (pp. 3345-3351). IEEE.

- D.P. Filev, F. Tseng, J. Kristinsson and R. 2011, “Contextual on-board learning and prediction of vehicle destinations.” In *Computational Intelligence in Vehicles and Transportation Systems (CIVTS), 2011 IEEE Symposium on* (pp. 87-91). IEEE.
- A. Kumar, F. Tseng and R. B. Chinnam, 2011, “Role of Hidden-Markov models for autonomous diagnostics of cutting tools”. In *Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on* (pp. 509-513). IEEE.
- F. Tseng, I. Makki, D.P. Filev and R.B. Chinnam, 2014, “A Novel Feature Extraction Method for Monitoring (Vehicular) Fuel Storage System Leaks”. *Proceedings of the Annual Conference of the Prognostics and Health Management Society 2014. Prognostics and Health Management Society*, p. 604-611 8 p.

6.2 CONCLUSION AND NEXT STEPS

With a newly proposed evolving clustering algorithm (ECA), this dissertation presents several approaches to address the multiple general facets of the diagnostics and prognostics problem. As part of the development and validation process of the newly proposed Mutual Information based Recursive Gustafson-Kessel-like (MIRGKL) clustering algorithm, we also validated its enhanced performance in terms of several cluster quality indices. In these examples, with real-world GPS trace/streaming data, we demonstrated that MIRGKL can form a powerful means for knowledge extraction of frequent locations as well as its use as a tool to compress full GPS route into data to only a small fraction of its original size.

Diagnostics (current system state determination) and prognostics (future system state determination and remaining useful life (RUL) prediction) are the two most critical elements of any Condition-Based Maintenance (CBM) practice. As a generic data processing procedure, ECA, including proposed MIRGKL, relaxes the requirement of system specific domain knowledge and

is computationally efficient since it is realized as a recursive learning and update procedure. Clusters, exemplified new and recurrent patterns of the system, their numerical characterizations, and the fact that the transitioning process is fully observable, makes Markov model a suitable method to capture such information. Complement to cluster specific health factors including age and normalized population size, we proposed some novel biologically inspired factors that are applicable to both the clustering process and individual clusters.

In addition, ECA lends itself applicable for function approximation tasks including the learning of RUL. In Chapter 4, we examined two cases with limited RUL information observability and obtained promising results. This is accomplished by following the eTS (Evolving Takagi-Sugeno) method to allow cluster specific local (degraded) function to be incremented with inferred RUL information. It should be noted that while we applied the Weibull function as the cluster specific model, it is possible and may be even more effective when cluster specific local functions are replaced with a simplified 1^{st} principle or other approximated equations that have a closer relationship to the problem at hand. Since which cluster specific local model to be updated is dictated by the cluster assignment, interpolation of inferred RUL is necessary. In the brake wear example, we also discovered the identification of appropriate horizon to present the degradation information in which the model will learn and adapt is of critical importance. Without that, the ability of ECA to generalize multiple drivers into a handful of prototypes to produce good RUL prediction capability will not be demonstrated effectively.

Finally, directions for future research can be broken down into the following: 1) An effective data driven mechanism to adaptively weigh the similarity measures in the KL divergence matrix formulations. While applying the mutual information based formulation, we have so far omitted the terms associated with shape and size while solely focusing on the normalized distance which is effectively the Mahalanobis distance. This should be considered a special case for information that are (approximately) normally distributed with the assumption that clusters far apart in the feature space are of great importance. When such adaptive mechanism is obtained, the whole

method will become even more generic and applications in biological imaging will become applicable. 2) While the learning of the transition probability matrix is straightforward, proposed monitoring procedures (macro and micro levels) in Chapter 5 and associated attributes need to be further validated with appropriate test data to see if they can be useful in the development of diagnostics and prognostics functions. 3) In Chapter 4, good results for the degradation of the drill-bit and brake wear were presented. However, the break wear experiment did not capture data associated with near end-of-life situations. Ideally, we would like to also see how far a prediction horizon we can extend in the prediction to maintain RUL prediction precision.

REFERENCES

[JL06] Jardine, A.K., Lin, D. and Banjevic, D., 2006. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20(7), pp.1483-1510.

[BMT99] Begg, C., Merdes, T., Byington, C.S. and Maynard, K.P., 1999, May. Mechanical system modeling for failure diagnostics and prognosis. *In Proc. Maintainability and Reliability Conf*.(MARCON 99) (pp. 10-12).

[KR02a] Kacprzynski, G.J., Roemer, M.J. and Hess, A.J., 2002. Health management system design: development, simulation and cost/benefit optimization. *In Aerospace Conference Proceedings*, 2002. IEEE (Vol. 6, pp. 6-3065). IEEE.

[HZ09] Heng, A., Zhang, S., Tan, A.C. and Mathew, J., 2009. Rotating machinery prognostics: State of the art, challenges and opportunities. *Mechanical Systems and Signal Processing*, 23(3), pp.724-739.

[LOS09] Luchinsky, D.G., Osipov, V.V., Smelyanskiy, V.N., Timucin, D.A., Uckun, S., Hayashida, B., Watson, M., McMillin, J., Shook, D., Johnson, M. and Hyde, S., 2009, March. Fault diagnostics and prognostics for large segmented SRMs. *In Aerospace conference*, 2009 IEEE (pp. 1-8). IEEE.

[EJJ13] Esperon-Miguez, M., John, P. and Jennions, I.K., 2013. A review of integrated vehicle health management tools for legacy platforms: challenges and opportunities. *Progress in Aerospace Sciences*, 56, pp.19-34.

[Pat02] Jr. JD. Patton, Preventive Maintenance, New York: Instrument of Society of America, 1983

[Yan02] S. K. Yang, An Experiment of State Estimation for Predictive Maintenance Using Kalman Filter on a DC Motor, Reliability Engineering and System Safety, 75(1), (2002) 103-111

[LT97] R. Logenrand, D. Talkington, Analysis of cellular and functional manufacturing systems in the presence of machine breakdown, International Journal of Production Economics, 53(3), (1997) 239-256

[BG01] C.S. Byington and A.K. Garga, Data Fusion for Developing Predictive Diagnostics for Electromechanical Systems, Handbook of Multisensor Data Fusion, D.L. Hall and J. Llinas eds., CRC Press, FL: Boca Raton, 2001.

[OSL02] R. Orsagh, C. Savage, M. Lebold, Development of Performance and Effectiveness Metrics for Gas Turbine Diagnostics Technologies IEEE Aerospace Conference Proceedings, 6 (2002) 2825-2835

[SK97]] Saranga, H. and Knezevic, J., 2001. Reliability prediction for condition-based maintained systems. Reliability Engineering & System Safety, 71(2), pp.219-224H. Saranga and J. Knezevic, “Reliability prediction for condition-based maintained systems,” Reliability Engineering and System Safety, Vol. 71, pp. 219-224, 2001.

[F05] Camci, F., 2005. Process monitoring, diagnostics and prognostics using support vector machines and hidden Markov models, Dissertation

[KT08] Kumar, A., Tseng, F., Guo, Y. and Chinnam, R.B., 2008, June. Hidden-Markov model based sequential clustering for autonomous diagnostics. *In Neural Networks*, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on (pp. 3345-3351). IEEE.

[LM03] Lin, D. and Makis, V., 2003. Recursive filters for a partially observable system subject to random failure. *Advances in Applied Probability*, 35(1), pp.207-227.

[SF03] Sohn, H., Farrar, C.R., Hemez, F.M., Park, G., Robertson, A.N. and Williams, T.O., 2003, July. A coupled approach to developing damage prognosis solutions. *In Key Engineering Materials* (Vol. 245, pp. 289-306). Trans Tech Publications.

[ZL11] Zhang, J. and Lee, J., 2011. A review on prognostics and health monitoring of Li-ion battery. *Journal of Power Sources*, 196(15), pp.6007-6014.

[EGBH00] S. J. Engel, B. J. Gilmartin, K. Bongort, A. Hess, Prognostics, The Real Issues Involved With Predicting Life Remaining, *IEEE Aerospace Conference Proceedings,* 6, (2000) 457-469.

[KS02] Kasabov, N.K. and Song, Q., 2002. DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems*, IEEE Transactions on, 10(2), pp.144-154.

[AK98] Amari, S.I. and Kasabov, N., 1998. *Brain-like computing and intelligent information systems*. Springer-Verlag Singapore Pte. Limited.

[YM98] Yamakawa, T. and Matsumoto, G., 1998. Evolving fuzzy neural networks-algorithms, applications and biological motivation. Methodologies for the conception, design and application of soft computing, World Scientific, 1, pp.271-274.

[FG10] Filev, D. and Georgieva, O., 2010. An extended version of the Gustafson–Kessel algorithm for evolving data stream clustering. *Evolving intelligent systems: methodology and applications*. IEEE Press Series on Computational Intelligence, Wiley, pp.273-300.

[SR12] Serir, L., Ramasso, E. and Zerhouni, N., 2012. Evidential evolving Gustafson–Kessel algorithm for online data streams partitioning using belief function theory. *International journal of approximate reasoning*, 53(5), pp.747-768.

[TK12] Tafaj, E., Kasneci, G., Rosenstiel, W. and Bogdan, M., 2012, March. Bayesian online clustering of eye movement data. In Proceedings of the Symposium on Eye Tracking Research and Applications (pp. 285-288). ACM.

[DS11] Dovžan, D. and Škrjanc, I., 2011. Recursive clustering based on a Gustafson–Kessel algorithm. *Evolving systems*, 2(1), pp.15-24.

[AM11] Ahmed, H.A., Mahanta, P., Bhattacharyya, D.K. and Kalita, J.K., 2011, October. Gerc: tree based clustering for gene expression data. In Bioinformatics and

Bioengineering (BIBE), 2011 IEEE 11th International Conference on (pp. 299-302). IEEE.

[YF94] Yager, R.R. and Filev, D.P., 1994. Approximate clustering via the mountain method. Systems, Man and Cybernetics, IEEE Transactions on, 24(8), pp.1279-1284..

[A92] Altman, N.S., 1992. An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3), pp.175-185.

[BE84] Bezdek, J.C., Ehrlich, R. and Full, W., 1984. FCM: The fuzzy c-means clustering algorithm. Computers & Geosciences, 10(2), pp.191-203.

[GK78] Gustafson, D.E. and Kessel, W.C., 1978. Fuzzy clustering with a fuzzy covariance matrix." *Scientific Systems. Inc*., Cambridge, MA.

[M36] Mahalanobis, P.C., 1936. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2, pp.49-55.

[GF09] Georgieva, O. and Filev, D., 2009, June. Gustafson-kessel algorithm for evolving data stream clustering. *In Proceedings of the International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing* (p. 62). ACM.

[MD08] Masson, M.H. and Denoeux, T., 2008. ECM: An evidential version of the fuzzy c-means algorithm. Pattern Recognition, 41(4), pp.1384-1397.

[FCT10] Filev, D.P., Chinnam, R.B., Tseng, F. and Baruah, P., 2010. An industrial strength novelty detection framework for autonomous equipment monitoring and diagnostics. *Industrial Informatics*, IEEE Transactions on, 6(4), pp.767-779.

[RS06] Rong, H.J., Sundararajan, N., Huang, G.B. and Saratchandran, P., 2006. Sequential adaptive fuzzy inference system (SAFIS) for nonlinear system identification and prediction. *Fuzzy sets and systems*, 157(9), pp.1260-1275.

[LC13] Lemos, A., Caminhas, W. and Gomide, F., 2013. Adaptive fault detection and diagnosis using an evolving fuzzy classifier. *Information Sciences*, 220, pp.64-85.

[IA12] Iglesias, J.A., Angelov, P., Ledezma, A. and Sanchis, A., 2012. Creating evolving user behavior profiles automatically. *Knowledge and Data Engineering,* IEEE Transactions on, 24(5), pp.854-867.

[MB15] Moshtaghi, M., Bezdek, J.C., Leckie, C., Karunasekera, S. and Palaniswami, M., 2015. Evolving fuzzy rules for anomaly detection in data streams. *Fuzzy Systems*, IEEE Transactions on, 23(3), pp.688-700.

[AF04] Angelov, P.P. and Filev, D.P., 2004. An approach to online identification of Takagi-Sugeno fuzzy models. *Systems, Man, and Cybernetics*, Part B: Cybernetics, IEEE Transactions on, 34(1), pp.484-498.

[YF94] Yager, R.R. and Filev, D.P., 1994. *Essentials of fuzzy modeling and control*. New York.

[F01] Filev, D., 2001, July. Rule-base guided adaptation for mode detection in process control. In IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th (Vol. 2, pp. 1068-1073). IEEE.

[FK15] Filev, D., Kolmanovsky, I. and Yager, R., 2015. Developing Fuzzy State Models as Markov Chain Models with Fuzzy Encoding. In Fifty Years of Fuzzy Logic and its Applications (pp. 79-106). Springer International Publishing.

[FT11] Filev, D., Tseng, F., Kristinsson, J. and McGee, R., 2011, April. Contextual on-board learning and prediction of vehicle destinations. *In Computational Intelligence in Vehicles and Transportation Systems* (CIVTS), 2011 IEEE Symposium on (pp. 87-91). IEEE.

[MG14] Maciel, L., Gomide, F. and Ballini, R., 2014. Enhanced evolving participatory learning fuzzy modeling: an application for asset returns volatility forecasting. Evolving Systems, 5(2), pp.75-88.

[LL01] Lin, F.J., Lin, C.H. and Shen, P.H., 2001. Self-constructing fuzzy neural network speed controller for permanent-magnet synchronous motor drive. Fuzzy Systems, IEEE Transactions on, 9(5), pp.751-759.

[ZD14] Zdešar, A., Dovžan, D. and Škrjanc, I., 2014. Self-tuning of 2 DOF control based on evolving fuzzy model. Applied Soft Computing, 19, pp.403-418.

[JM99] Jain, A.K., Murty, M.N. and Flynn, P.J., 1999. Data clustering: a review. ACM computing surveys (CSUR), 31(3), pp.264-323.

[XW05] Xu, R. and Wunsch, D., 2005. Survey of clustering algorithms. *Neural Networks*, IEEE Transactions on, 16(3), pp.645-678.

[B95] Bishop, C.M., 1995. *Neural networks for pattern recognition*. Oxford university press.

[CV95] Cortes, C. and Vapnik, V., 1995. Support vector machine. Machine learning, 20(3), pp.273-2

[T01] Tipping, M.E., 2001. Sparse Bayesian learning and the relevance vector machine. The journal of machine learning research, 1, pp.211-244.

[DH01] Domingos, P.M. and Hulten, G., 2001, May. Catching up with the Data: Research Issues in Mining Data Streams. In DMKD.

[GZ09] Gaber, M.M., Zaslavsky, A. and Krishnaswamy, S., 2009. Data stream mining. In *Data Mining and Knowledge Discovery Handbook* (pp. 759-787). Springer US.

[LA14] Langone, R., Agudelo, O.M., De Moor, B. and Suykens, J.A., 2014. Incremental kernel spectral clustering for online learning of non-stationary data. *Neurocomputing*, 139, pp.246-260.

[FC04a] F. Camci, R. B. Chinnam, Online Novelty Detections for Non-stationary Classes using a Modified SVM Proceedings of Neural Networks and Computational Intelligence, Switzerland, 2004

[FC04b] F. Camci, R. B. Chinnam, Non-stationary Data Domain Description using Support Vector Novelty Detector, International Joint Conference on Neural Networks, Budapest, 2004

[A13] Angelov, P.P., 2013. *Evolving rule-based models: a tool for design of flexible adaptive systems* (Vol. 92). Physica.

[A12] Angelov, P., 2012. Evolving fuzzy systems. Computational Complexity: Theory, Techniques, and Applications, pp.1053-1065.

[L08] Lughofer, E.D., 2008. FLEXFIS: a robust incremental learning approach for evolving Takagi–Sugeno fuzzy models. *Fuzzy Systems*, IEEE Transactions on, 16(6), pp.1393-1410.

[LC15] Lughofer, E., Cernuda, C., Kindermann, S. and Pratama, M., 2015. Generalized smart evolving fuzzy systems. *Evolving Systems*, 6(4), pp.269-292.

[R90] Yager, R.R., 1990. A model of participatory learning. *Systems, Man and Cybernetics*, IEEE Transactions on, 20(5), pp.1229-1234.

[LG06] Lima, E., Gomide, F. and Ballini, R., 2006, September. Participatory evolving fuzzy modeling. In *Evolving Fuzzy Systems*, 2006 International Symposium on (pp. 36-41). IEEE.

[AF05] Angelov, P. and Filev, D., 2005, May. Simpl_eTS: a simplified method for learning evolving Takagi-Sugeno fuzzy models. In Fuzzy Systems, 2005. FUZZ'05. The 14th IEEE International Conference on (pp. 1068-1073). IEEE.

[OL04] Oliveira, A.C. and Lorena, L.A., 2004. Detecting promising areas by evolutionary clustering search. In *Advances in Artificial Intelligence*–SBIA 2004 (pp. 385-394). Springer Berlin Heidelberg.

[AG01] Anagnostopoulos, G.C. and Georgiopoulos, M., 2001, March. Ellipsoid ART and ARTMAP for incremental unsupervised and supervised learning. In

Aerospace/Defense Sensing, Simulation, and Controls (pp. 293-304). International Society for Optics and Photonics.

[BB91] Bobrowski, L. and Bezdek, J.C., 1991. c-means clustering with the l l and l∞ norms. Systems, Man and Cybernetics, IEEE Transactions on, 21(3), pp.545-554.

[CG91] Carpenter, G.A., Grossberg, S. and Rosen, D.B., 1991. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural networks, 4(6), pp.759-771.

[HB00] Hathaway, R.J., Bezdek, J.C. and Hu, Y., 2000. Generalized fuzzy c-means clustering strategies using L p norm distances. Fuzzy Systems, IEEE Transactions on, 8(5), pp.576-582.

[M67] MacQueen, J., 1967, June. Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability (Vol. 1, No. 14, pp. 281-297).

[MJ94] Mao, J. and Jain, A.K., 1994. A self-organizing network for hyperellipsoidal clustering (HEC). In Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on (Vol. 5, pp. 2967-2972). IEEE.

[M36] Mahalanobis, P.C., 1936. On the generalized distance in statistics. Proceedings of the National Institute of Sciences (Calcutta), 2, pp.49-55.

[B67] Bregman, L.M., 1967. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR computational mathematics and mathematical physics, 7(3), pp.200-217.

[BM05a] Banerjee, A., Guo, X. and Wang, H., 2005. On the optimality of conditional expectation as a Bregman predictor. Information Theory, IEEE Transactions on, 51(7), pp.2664-2669.

[BM05b] Banerjee, A., Merugu, S., Dhillon, I.S. and Ghosh, J., 2005. Clustering with Bregman divergences. The Journal of Machine Learning Research, 6, pp.1705-1749.

[L91] Lin, J., 1991. Divergence measures based on the Shannon entropy. Information Theory, IEEE Transactions on, 37(1), pp.145-151.

[VWK99] G. Vachtsevanos, P. Wang and N. Khiripet, “Prognostication: Algorithms and Performance Assessment Methodologies”, ATP Fall National Meeting Condition-Based Maintenance Workshop, San Jose, California, November 15-17, 1999. 50

[HW79] Hartigan, J.A. and Wong, M.A., 1979. Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1), pp.100-108.

[WC01] Wagstaff, K., Cardie, C., Rogers, S. and Schrödl, S., 2001, June. Constrained k-means clustering with background knowledge. In ICML (Vol. 1, pp. 577-584).

[BM99] Begg, C., Merdes, T., Byington, C.S. and Maynard, K.P., 1999, May. Mechanical system modeling for failure diagnostics and prognosis. In Proc. Maintainability and Reliability Conf.(MARCON 99) (pp. 10-12).

[MC01] Mathur, A., Cavanaugh, K.F., Pattipati, K.R., Willett, P.K. and Galie, T.R., 2001, July. Reasoning and modeling systems in diagnosis and prognosis. In Aerospace/Defense Sensing, Simulation, and Controls (pp. 194-203). International Society for Optics and Photonics.

[CW10] Caesarendra, W., Widodo, A. and Yang, B.S., 2010. Application of relevance vector machine and logistic regression for machine degradation assessment. Mechanical Systems and Signal Processing, 24(4), pp.1161-1171.

[HL07] Huang, R., Xi, L., Li, X., Liu, C.R., Qiu, H. and Lee, J., 2007. Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods. Mechanical Systems and Signal Processing, 21(1), pp.193-207.

[KR02b] Kacprzynski, G.J., Roemer, M.J., Modgil, G., Palladino, A. and Maynard, K., 2002. Enhancement of physics-of-failure prognostic models with system level features. In Aerospace Conference Proceedings, 2002. IEEE (Vol. 6, pp. 6-2919). IEEE.

[RK00] Roemer, M.J. and Kacprzynski, G.J., 2000. Advanced diagnostics and prognostics for gas turbine engine risk assessment. In Aerospace Conference Proceedings, 2000 IEEE (Vol. 6, pp. 345-353). IEEE.

[TV01] Tappert, P., Von Flotow, A. and Mercadal, M., 2001. Autonomous PHM with blade-tip-sensors: algorithms and seeded fault experience. In Aerospace Conference, 2001, IEEE Proceedings. (Vol. 7, pp. 7-3295). IEEE.

[HH01] Hardman, W., Hess, A., Ahne, R. and Blunt, D., 2001. USN Development Strategy, Fault Testing Results, and Future Plans for Diagnostics, Prognostics and Health Management of Helicopter Drive Train Systems. In DSTO International Conference on Health and Usage Monitoring (Vol. 1).

[GH09] Guo, H., Watson, S., Tavner, P. and Xiang, J., 2009. Reliability analysis for wind turbines with incomplete failure data collected from after the date of initial installation. Reliability Engineering & System Safety, 94(6), pp.1057-1063.

[SR03] Schömig, A.K. and Rose, O., 2003, January. On the suitability of the Weibull distribution for the approximation of machine failures. In IIE Annual Conference. Proceedings (p. 1). Institute of Industrial Engineers-Publisher.

[G00] Groer, P.G., 2000. Analysis of time-to-failure with a Weibull model. In Proceedings of the maintenance and reliability conference (pp. 59-01).

[DT12] Di Maio, F., Tsui, K.L. and Zio, E., 2012. Combining relevance vector machines and exponential regression for bearing residual life estimation. Mechanical Systems and Signal Processing, 31, pp.405-427.

[NA14] Newcombe, P.J., Ali, H.R., Blows, F.M., Provenzano, E., Pharoah, P.D., Caldas, C. and Richardson, S., 2014. Weibull regression with Bayesian variable selection to identify prognostic tumour markers of breast cancer survival. Statistical methods in medical research, p.0962280214548748.

[CC10] Camci, F. and Chinnam, R.B., 2010. Health-state estimation and prognostics in machining processes. Automation Science and Engineering, IEEE Transactions on, 7(3), pp.581-597.

[B14] Bryant, M.D., 2014, June. A data driven method for model based diagnostics and prognostics. In Proceedings of the 2014 IEEE Conference on Prognostics and Health Management, Cheney, WA, USA (pp. 22-25).

[J2007] Lee, J., 2007. A systematic approach for developing and deploying advanced prognostics technologies and tools: methodology and applications. In Proceedings of the Second World Congress on Engineering Asset Management, Harrogate, UK (pp. 1195-1206).

[JR06] Joshi, R. and Reeves, C., 2006. Beyond the Cox model: artificial neural networks for survival analysis part II. In Proceedings of the eighteenth international conference on systems engineering (pp. 179-184).

[SG09] Saha, B., Goebel, K., Poll, S. and Christophersen, J., 2009. Prognostics methods for battery health monitoring using a Bayesian framework. Instrumentation and Measurement, IEEE Transactions on, 58(2), pp.291-296.

[FT06a] Filev, D.P. and Tseng, F., 2006, September. Novelty detection based machine health prognostics. In Evolving Fuzzy Systems, 2006 International Symposium on (pp. 193-199). IEEE.

[FT06b] Filev, D.P. and Tseng, F., 2006, June. Real time novelty detection modeling for machine health prognostics. In NAFIPS 2006-2006 Annual Meeting of the North American Fuzzy Information Processing Society.

[HC06] Hess, A., Calvello, G., Frith, P., Engel, S.J. and Hoitsma, D., 2006, March. Challenges, issues, and lessons learned chasing the “Big P”: real predictive prognostics Part 2. In Aerospace Conference, 2006 IEEE (pp. 1-19). IEEE.

[CC05] F. Camci, R. B. Chinnam, Dynamic Bayesian Networks for Machine Diagnostics: Hierarchical Hidden Markov Models vs. Competitive Learning, Proceedings of the International Joint Conference on Neural Networks, 2005

[LT12] Leonardo Silva Teixeira, E., Tjahjono, B., Crisóstomo Absi Alfaro, S. and Manuel Soares Julião, J., 2012. Harnessing prognostics health management and product-service systems interaction to support operational decisions. Journal of Manufacturing Technology Management, 24(1), pp.78-94.

[DA11] Collins, D.H., Anderson-Cook, C.M. and Huzurbazar, A.V., 2011. System health assessment. Quality Engineering, 23(2), pp.142-151.

[KT06] Khalak, A. and Tierno, J., 2006, June. Influence of prognostic health management on logistic supply chain. In American Control Conference, 2006 (pp. 6-pp). IEEE.

[SZ10] Sun, B., Zeng, S., Kang, R. and Pecht, M., 2010, January. Benefits analysis of prognostics in systems. In Prognostics and Health Management Conference, 2010. PHM'10. (pp. 1-8). IEEE.

[E02] Estivill-Castro, V., 2002. Why so many clustering algorithms: a position paper. ACM SIGKDD explorations newsletter, 4(1), pp.65-75.

[DK32] Driver, H.E. and Kroeber, A.L., 1932. Quantitative expression of cultural relationships. University of California Press.

[T39] Tryon, R.C., 1939. Cluster analysis: correlation profile and orthometric (factor) analysis for the isolation of unities in mind and personality. Edwards brother, Incorpora

[AF10] Angelov, P., Filev, D.P. and Kasabov, N. eds., 2010. Evolving intelligent systems: methodology and applications (Vol. 12). John Wiley & Sons.

[QW08] Qiao, J. and Wang, H., 2008. A self-organizing fuzzy neural network and its applications to function approximation and forecast modeling. Neurocomputing, 71(4), pp.564-569.

[TK12] Tafaj, E., Kasneci, G., Rosenstiel, W. and Bogdan, M., 2012, March. Bayesian online clustering of eye movement data. In Proceedings of the Symposium on Eye Tracking Research and Applications (pp. 285-288). ACM.

[BL05] Boubacar, H.A. and Lecoeuche, S., 2005. A new kernel-based algorithm for online clustering. In Artificial Neural Networks: Formal Models and Their Applications–ICANN 2005 (pp. 583-588). Springer Berlin Heidelberg.

[BZ06] Bouguila, N. and Ziou, D., 2006. Online clustering via finite mixtures of Dirichlet and minimum message length. Engineering Applications of Artificial Intelligence, 19(4), pp.371-379.

[DH12] Duda, R.O., Hart, P.E. and Stork, D.G., 2012. Pattern classification. John Wiley & Sons.

[W50] Woodbury, M.A., 1950. Inverting Modified Matrices, Memorandum Rept. 42. Statistical Research Group, Princeton University, Princeton, NJ, 316.

[D07] Duchi, J., 2007. Derivations for linear algebra and optimization. Berkeley, California.

[K12] Kohonen, T., 2012. Self-organization and associative memory (Vol. 8). Springer Science & Business Media.

[TF16] Tseng, F., Filev, D.P., Makki, I.H. and Michelini, J.O., FORD GLOBAL TECHNOLOGIES, LLC, 2016. SUGGESTIVE MAPPING USING RELEVANCE BASED DRIVE INFORMATION. U.S. Patent 20,160,109,243.

[MF12] McGee, R.A., Filev, D.P., Kristinsson, J.G., Tseng, F. and Phillips, A.M., Ford Global Technologies, Llc, 2012. Methods and Apparatus for Context Based Trip Planning. U.S. Patent Application 13/714,919.

[YT15] Yu, H., Tseng, F. and Wang, Q., Ford Global Technologies, Llc, 2015. Trip partitioning based on driving pattern energy consumption. U.S. Patent 9,121,722.

[DB79] Davies, D.L. and Bouldin, D.W., 1979. A cluster separation measure. Pattern Analysis and Machine Intelligence, IEEE Transactions on, (2), pp.224-227.

[SW11] Si, X.S., Wang, W., Hu, C.H. and Zhou, D.H., 2011. Remaining useful life estimation–A review on the statistical data driven approaches. European Journal of Operational Research, 213(1), pp.1-14.

[GMM10] Gorjian, N., Ma, L., Mittinty, M., Yarlagadda, P. and Sun, Y., 2010. A review on degradation models in reliability analysis. In Engineering Asset Lifecycle Management (pp. 369-384). Springer London.

[K94] Kosko, B., 1994. Fuzzy systems as universal approximators. IEEE transactions on computers, 43(11), pp.1329-1333.

[KK03] Klutke, G.A., Kiessler, P.C. and Wortman, M.A., 2003. A critical look at the bathtub curve. IEEE Transactions on Reliability, 52(1), pp.125-129.

[CL13] Cordeiro, G.M. and Lemonte, A.J., 2013. On the Marshall–Olkin extended weibull distribution. Statistical Papers, 54(2), pp.333-353.

[AN14] Almalki, S.J. and Nadarajah, S., 2014. Modifications of the Weibull distribution: a review. Reliability Engineering & System Safety, 124, pp.32-55.

[NA15] Neuner, S., Albrecht, A., Cullmann, D., Engels, F., Griess, V.C., Hahn, W.A., Hanewinkel, M., Härtl, F., Kölling, C., Staupendahl, K. and Knoke, T., 2015. Survival of Norway spruce remains higher in mixed stands under a dryer and warmer climate. Global change biology, 21(2), pp.935-946.

[BL12] Bebbington, M., Lai, C.D., Wellington, M. and Zitikis, R., 2012. The discrete additive Weibull distribution: A bathtub-shaped hazard for discontinuous failure data. Reliability Engineering & System Safety, 106, pp.37-44.

[UG15] Urrutia, J.D., Gayo, W.S., Bautista, L.A. and Baccay, E.B., 2015. Survival Analysis of Patients with End Stage Renal Disease. In Journal of Physics: Conference Series (Vol. 622, No. 1, p. 012014). IOP Publishing.

[W51] Weibull, W., 1951. A Statistical Distribution Function of Wide applicability. Journal of applied mechanics, 103, pp.293-297

[KK03] Keogh, E. and Kasetty, S., 2003. On the need for time series data mining benchmarks: a survey and empirical demonstration. Data Mining and knowledge discovery, 7(4), pp.349-371.

[KR05] Keogh, E. and Ratanamahatana, C.A., 2005. Exact indexing of dynamic time warping. Knowledge and information systems, 7(3), pp.358-386.

[SC78] Sakoe, H. and Chiba, S., 1978. Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing, 26(1), pp.43-49.

[KP00] Keogh, E.J. and Pazzani, M.J., 2000, August. Scaling up dynamic time warping for datamining applications. In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 285-289). ACM.

[DT08] Ding, H., Trajcevski, G., Scheuermann, P., Wang, X. and Keogh, E., 2008. Querying and mining of time series data: experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment, 1(2), pp.1542-1552.

[RC12] Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J. and Keogh, E., 2012, August. Searching and mining trillions of time series subsequences under dynamic time warping. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 262-270). ACM.

[AC01] Aach, J. and Church, G.M., 2001. Aligning gene expression time series with time warping algorithms. Bioinformatics, 17(6), pp.495-508.

[WJ14] Wu, Y., Jin, H., Li, Y., Ji, Z. and Hou, S., 2014. Simulation of Temperature Distribution in Disk Brake Considering a Real Brake Pad Wear. Tribology Letters, 56(2), pp.205-213.

[HM16] Hassan, A.K.F. and Mohammed, S., 2016. Artificial Neural Network Model for Estimation of Wear and Temperature in Pin-disc Contact.

[BY13] Bao, J., Yin, Y., Lu, Y., Hu, D. and Lu, L., 2013. A cusp catastrophe model for the friction catastrophe of mine brake material in continuous repeated brakings. Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology, p.1350650113482825.

[JS15] Jegadeeshwaran, R. and Sugumaran, V., 2015. Fault diagnosis of automobile hydraulic brake system using statistical features and support vector machines. Mechanical Systems and Signal Processing, 52, pp.436-446.

[FK14] Filev, D.P. and Kolmanovsky, I., 2014. Generalized markov models for real-time modeling of continuous systems. IEEE Transactions on Fuzzy Systems, 22(4), pp.983-998.

[FK12] Filev, D.P. and Kolmanovsky, I., 2012. Markov chain modeling and on-board identification for automotive vehicles. In Identification for Automotive Systems (pp. 111-128). Springer London.

[DR08] DeBerardinis, R.J., Lum, J.J., Hatzivassiliou, G. and Thompson, C.B., 2008. The biology of cancer: metabolic reprogramming fuels cell growth and proliferation. Cell metabolism, 7(1), pp.11-20.

[NCI15] National Cancer Institute. What is Cancer? Updated 02/09/15. https://www.cancer.gov/about-cancer/understanding/what-is-cancer

[VW15] Cacer Cells vs. Normal Cells – A Dozen Difference https://www.verywell.com/cancer-cells-vs-normal-cells-2248794

ABSTRACT

EVOLVING CLUSTERING ALGORITHMS AND THEIR APPLICATION FOR CONDITION MONITORING, DIAGNOSTICS, & PROGNOSTICS

by

FLING FINN TSENG

May 2017

Advisor: Dr. Ratna Babu Chinnam

Major: Industrial Engineering

Degree: Doctor of Philosophy

Application of Condition-Based Maintenance (CBM) technology requires effective yet generic data driven methods capable of carrying out diagnostics and prognostics tasks without detailed domain knowledge and human intervention. Improved system availability, operational safety and enhanced logistics and supply chain performance at a reduced cost level could be achieved with the widespread deployment of CBM.

Proposed Mutual Information based Recursive Gustafson-Kessel-like (MIRGKL) clustering algorithm operates recursively to identify underlying model structure and parameters from stream type of data. Inspired by the Evolving Gustafson-Kessel-like Clustering (eGKL) algorithm, we applied the notion of mutual information to the well-known Mahalanobis distance as the governing similarity measure throughout. This is also a special case of the Kullback-Leibler (KL) Divergence where between cluster shape information (governed by the determinant and trace of the covariance matrix) is omitted and only applicable in the case of (approximately) normally distributed data. In the cluster assignment and consolidation process, we proposed the use of the

Chi-square statistic with the provision of having different probability thresholds for the evaluations of cluster assignment and subsequent cluster merging (pruning) process. Due to the symmetry and boundedness property brought in by the mutual information formulation, we have shown with real-world data that the algorithm's performance becomes less sensitive to the same range of probability thresholds which makes tuning a simpler task in practice. As a result, improvement demonstrated by the proposed algorithm has implications in improving generic data-driven methods for diagnostics, prognostics, generic function approximations and knowledge extractions for stream type of data.

When discovered evolving clusters are attached with local models, the algorithm becomes a generic engine for function approximation tasks. For prognostics purposes, the algorithm incrementally refines identified cluster structure and parameters and gradually learns local models to match variants of degradation profiles from directly observed or inferred remaining useful life (RUL) information.

Proposed Mutual Information based Recursive Gustafson-Kessel-like (MIRGKL) clustering algorithm, like other evolving clustering methods, may also serve as a tool to monitor the evolution of data through means of novelty detection. Association between diagnostics and prognostics models to evolving clustering is realized by modeling of identified clusters as states using probability theory (i.e. Markov Chain), information theory (Kullback-Leibler divergence) and statistical analysis (i.e. statistical process control) at both the macro and micro level. MIRGKL's effectiveness in clustering, knowledge representation, and its use for diagnostics and prognostics applications all show very promising results and these findings are included as part of the dissertation.

AUTOBIOGRAPHICAL STATEMENT

Fling Finn Tseng is a Doctoral Candidate in the Department of Industrial & Systems Engineering at Wayne State University (U.S.A.). He received his B.S. degree in Mechanical Engineering from National Chiao-Tung University (1996, Taiwan) and M.S. degrees in Mechanical Engineering and Industrial and Operations Engineering from University of Michigan – Ann Arbor (2001 and 2002, USA). His research interests include Computational Intelligence, Supervised and Unsupervised Clustering and their applications in Knowledge Extraction, Predictive Modeling and Data-driven Methods for Condition-Based Maintenance.