Using Machine Learning Algorithms to Improve HUMS Performance

Daniel Wade
Aerospace Engineer
United States Army
Redstone Arsenal, AL

Ramon Lugos
Aerospace Engineer
Avion Solutions, Inc
Redstone Arsenal, AL

Matthew Szelistowski
Aerospace Engineer
RMCI Inc
Redstone Arsenal, AL

Abstract

The United States Army has implemented a rotorcraft Condition Based Maintenance (CBM) program based on monitoring recorded aircraft parameters and vibration signatures. The data from these Health and Usage Monitoring Systems (HUMS) has been stored for over a decade while Army Engineers have used the information to design features that indicate a need for maintenance, extend component life, or reduce the number of operational precautionary landings. The features developed and fielded today are designed to detect particular failure modes; however, they can be notoriously noisy from reading to reading due to recorded but unaccounted-for variables. This paper outlines the procedures necessary for training a machine learning algorithm to account for these variables, improving HUMS performance and using data sets that have been warehoused for nearly a decade. A number of areas are addressed, including the importance of data selection, feature selection by subject matter experts versus unsupervised machine learning, dimensionality reduction of data sets, error measurement, types of individual and ensemble classification models, and procedures for internal cross validation for model selection. This paper also covers important overarching machine learning practices that seek to maintain the integrity of the process, which include safeguarding against data snooping and establishing credible bounds on generalization error.

INTRODUCTION

The Army Aviation Condition Based Maintenance Plus (CBM+) Program has focused on monitoring the functional failure of dynamic components, recording engine parameters, and developing regime recognition. Vibration and parametric data are recorded by the Modernized Signal Processing Unit (AH-64D, AH-64E, CH-47D, MH-47G, and MH-60M) and Integrated Vehicle Health Monitoring System (UH-60 models A, L, and M and HH-60M), which are collectively called Health and Usage Monitoring Systems (HUMS). The authors are part of the team analyzing vibration data collected by these systems with the ultimate goal of diagnosing and predicting dynamic component failure internal to gearboxes or hanger and swashplate bearings. For the purposes of this paper, it is critical that the reader understand the definitions of the Army CBM+ program associated with maintenance criteria, specifically those relating to baseline risk, fault, and failure.

Baseline Risk: The accepted risk in production, operations, and maintenance procedures reflected in frozen planning, the Operator's Manuals, and the Maintenance manuals for the baseline aircraft. Maintenance procedures include all required condition inspections with intervals, retirement times, and Time Between Overhauls (TBOs) (Ref. 1).

Fault: An undesired anomaly in an item or system (Ref. 1).

Failure: The loss of function of a part, component, or system caused by the presence of a fault (Ref. 1).

An optimized HUMS detects incipient faults, determines the severity of the faults, and provides feedback to maintenance personnel to indicate when a fault has become a failure. For the purposes of health monitoring, it is necessary to define an additional state between fault and failure because many components

[†] *Presented at the American Helicopter Society CBM Specialists' Meeting, Huntsville, AL, 9-11 Feb, 2015.*

on Army aircraft have fault-tolerant designs. In this state, a fault is present and there is an increase in baseline risk, though actual failure has not yet occurred. The authors have chosen to use the term *functional failure* for this purpose, which can be defined in the following way:

- ❖ *Functional failure* is the point at which a diagnostic should alert the user that failure is impending but has not yet actually occurred according to the true definition of failure. The diagnostic should trigger functional failure at least two reporting/data download interval prior to absolute failure. Functional failure is typically shown by the HUMS in *red*, and requires maintenance action(s).

The program for monitoring the health of dynamic components started when the Army began collecting vibration data associated with failures over the last decade. Papers reviewing the success of these programs at diagnosing incipient faults have been published in References 2 through 10. The Army has focused on using and improving the system *as fielded*, rather than making major changes to the software or hardware infrastructure. This means that individual indicators of component health, called Condition Indicators (CIs), were compared across the fleet and thresholds were set based on Receiver Operating Characteristic (ROC) curve optimization. Details regarding the computation and meaning of ROC curves and their importance to the legacy methods for aircraft safety limits was published by Dempsey, et al. in Reference 7.

The Army also instituted two complementary Teardown Analysis (TDA) programs as part of the CBM+ program. They focus on expert reviews of component condition at the end of life (i.e. post-mortem). Gearboxes and bearings are sent to a controlled location for careful disassembly by experts. The output of the HUMS diagnostics can be checked against these TDAs, leading to the modification of software configuration thresholds to improve accuracy.

The TDA program has created a large ground truth data set that contains diagnostic readings from aircraft and associated component condition and damage criticality. For example, a removed gearbox that contains two gears and four bearings will be given a fault/no-fault found rating for each of these six components, as well as an estimated life remaining per the ADS-79 recommendations (Ref. 1). These ratings are defined by ADS-79D color codes:

- Green: No fault found.
- Yellow/Amber/Orange: Incipient failure, life is limited. If possible, perform maintenance to return to a green state.
- Red: Late stage failure, maintenance required prior to further operation.

These fleet-wide thresholds are easy to implement in the field and are historically well understood by pilots and maintenance crew. All of the major platforms flown by the Army today have some kind of vibration-based safety thresholds for dynamic components, typically designed around shaft imbalance detection. The Army Vibration Analyzer (AVA) was used for the measurement of these vibration magnitudes up through the beginning of the Vibration Management Enhancement Program and the institution of the CBM+ Program.

Fleet limits work well for shaft imbalance, especially for situations in which Soldiers can remedy out-of-balance shafts with balancing weights or *shaft indexing* procedures. Diagnosis of faulted bearings and gears, however, has had mixed results. As an example, Keller, et al. (Ref. 3, 10-13) have demonstrated that these diagnostic methods with fleet-wide thresholds work well on the Apache Nose Gearbox and Black Hawk Accessory Modules. However, Dempsey, et al. (Ref. 9) demonstrated that these methods have failed on other components. Krick, et al. (Ref. 12, 13) noted that, in many cases, a better diagnostic is the rate of change of the CI, rather than the actual value of the CI. Furthermore, it was noted by Wade, et al. (Ref. 14-16) that sensor locations for the MSPU and IVHMS were not optimized upon installation. Instead, the sensors were placed in locations where installation was convenient or where they would not interfere with the gearbox design, rather than where vibration transfer paths would be optimal.

The Army therefore took two approaches to improve the gear and bearing diagnostic behavior across the fleet: high frequency gearbox frequency response estimation and *dynamic thresholding*.

Frequency Response Estimation

The vibration characteristics of the gearboxes and bearings on board Army aircraft are responsible for transmitting vibration produced by faulted components to the appropriate sensors. Through a low-cost measurement program, the Army was able to adjust diagnostic algorithms without the need for observed in-situ failures based on gearbox and drive train resonance characteristics. In Reference 15, Antolick predicted that, had the resonance estimation program been implemented prior to the beginning of the CBM+ program, the Army would have seen a 20% improvement in diagnostic accuracy on day one.

The detailed process of making frequency response measurements and their specific application to bearing diagnostics was discussed by Wade and Larsen in Reference 14. Additionally, Szelistowski (Ref. 16) showed the importance of understanding the time domain waveform that is created by a bearing defect and how that is transmitted through the structure via the frequency response function to the installed accelerometers.

Dynamic Thresholding

While the resonance estimation program was being conducted, the Army also demonstrated two methods of alert generation based on CI *history trending*. The results of these studies were published by Krick, in References 12 and 13, who demonstrated that the accuracy of some diagnostic algorithms improved from below 50% to more than 90%, well within acceptable levels per ADS-79 (Ref. 1). The algorithms used for *trending* have all been developed by outside contractors HUMAWARE and Impact/Sikorsky. The algorithms can be described as follows:

- The Constant False Alert Rate algorithm (HUMAWARE) is designed to detect significant step changes in the data stream.
- The Autotrend algorithm (HUMAWARE) is designed to alert the user when a long term trend begins to develop.
- The Statistical Change Detection Gap algorithm (Impact) detects step changes, up or down, in the data stream.
- The Statistical Change Detection Scatter algorithm (Impact) detects scatter changes, increased or decreased, in the data stream.
- The Statistical Change Detection Trend algorithm (Impact) detects linear correlations, up or down, in the data stream.

Each of the algorithms has shown usefulness on the Army dataset for different fault and failure detections. However, the Army cannot implement these time-based algorithms wholesale in the HUMS software because they are not easy to interpret by the end user, as the algorithms are both sensitive to trends in noise and normal fluctuations in aircraft health. Furthermore, implementing all five algorithms would add that many more outputs for Soldiers to review following a flight. While these *dynamic thresholding* methods certainly have engineering value, they require some form of data fusion so that they can be incorporated into the existing HUMS infrastructure without burdening Soldiers.

Summary of the Limitations of Military HUMS as Fielded Today

- Although setting safety-based, fleet-wide thresholds is appropriate for shafts, it is not always possible for gears and bearings. To achieve an acceptable False Negative rate, gear and bearing thresholds are typically set too low, which leads to an unacceptable false alert rate.
- Vibration-based diagnostics require knowledge of the frequency response function of the component and its assembly. Sensors should be placed intentionally, with an understanding of the vibration signals to be detected and measured. In situations where sensors are placed in *convenient locations* without an understanding of assembly dynamics, simply estimating the frequency response function may not improve the diagnostics.
- In some cases, the measured change in the CI is a much better diagnostic than the *level alert* methodology. Adding these algorithms to the HUMS software is not simple, nor is it user-friendly.

MOTIVATION AND GOALS

Much of the data collected by the HUMS is not used for diagnostics. For example, it is well known that drive train vibration is affected by the torque applied (Ref. 17), but it is not always used by engineers to improve diagnostic algorithm performance. Furthermore, airspeed also affects tail rotor vibration, but is only used to broadly categorize vibration data by regimes. In the future, airspeed should be used as an input into diagnostics.

As much as 90% of the data collected by the HUMS consists of parametric variables not associated directly with vibration (e.g., altitude, engine compressor speed, temperature values, stick position, weight-on-wheels switch, etc.). Army engineers can use this data for many things, including a basic diagnosis of engine health. One of the many benefits of HUMS for a CBM+ program is the ability to make better business decisions. The parametric data collected contains the majority of the dataset required to learn the function associated with engine health that could improve mission planning. If such an engine health algorithm was to be developed, overall performance could be used to select the aircraft to be used for critical missions, with a quick glance at the HUMS ground station, based on operational data from the last several missions. However, the parametric datasets necessary to develop and train such algorithms are vast. Sheer dataset size and the number of dimensions prevent the training of such algorithms based solely on physics of failure or intuition.

The field of machine learning has received significant attention in the last decade. It has a broad focus and is used in many fields ranging from medicine, business, and economics to robotics. Some of its most popular and successful applications are in cases where there are such large amounts of data that subject matter experts alone, using physics-based approaches and heuristics, are unable to extract as much useful information as is possible from the data because its sheer volume and dimensionality obscure the relationships between data and ground truth. The algorithms used in machine learning offer Army engineers the opportunity to fuse the many dimensions of HUMS data that have never been correlated together into an output that is user-friendly and actionable. The size of the RIMFIRE database has greatly increased over the last decade and thus offers the ground truth necessary to validate algorithm accuracy.

Machine learning outputs can benefit many aspects of the Army aviation enterprise; for example, they can be used to improve the logistics management system. Each currently available CI from the HUMS is associated with a particular level of maturity. Some of the CIs are associated with a large number of ground truth examples, but most (as many as 80% on a given aircraft) have only one or two ground truth examples. Due to the complexities of operating a large fleet of aircraft, and the constantly changing environment regarding currently fielded software configurations and available ground truth, it is difficult to have a uniform policy for utilizing HUMS data at the aviation enterprise level to make better logistics decisions. Machine learning algorithms can be deployed at the Army aviation enterprise level by the engineers who develop HUMS to bring trusted information to logisticians rather than requiring logisticians to decide what information to trust, which could potentially result in poor decisions made on inaccurate CIs.

The goal of this paper is to outline a foundational process that can be used to create machine learning models and to improve the functionality of the Army HUMS. There are several long-term goals that will be achieved upon establishing this foundation and they are enumerated in the next section.

Long Term Goals

1. Use machine learning algorithms to improve HUMS diagnostic algorithm performance to ADS-79 standards.
    a. Discover the variables not used in HUMS data analysis today that maximize HUMS diagnostic performance.
    b. Fuse *dynamic thresholding* algorithms that account for data history with traditional level alerts.
    c. Use machine learning algorithms to reduce false alerts caused by HUMS to near zero.
2. Reduce the Soldier burden associated with HUMS data review.
3. Develop a rigorous process that can be expanded and used by industry for training machine

learning algorithms while using HUMS data as applied to rotorcraft performance and airworthiness.

4. Reduce the probability of mission-aborting failures caused by dynamic components to near zero.
5. Use machine learning regression algorithms to predict a Maintenance Free Operating Period (MFOP) for the drive train (Ref. 18).
6. Increase the value of warehoused HUMS data combined with the documented RIMFIRE teardowns for engines to diagnose the likelihood of engine turn-ins due to low power.

MACHINE LEARNING

The authors have surveyed and attended several courses offered on the topic of machine or statistical learning from a diverse selection of universities. Courses that are offered cover a wide selection of topics from unsupervised and supervised training procedures, best practices, and rules. Important subtopics covered include bias and variance reduction, overfitting, model selection, and training methods.

The process developed by the authors is from a combination of the following sources:

1. ADS-79D (Ref. 1)
   ADS-79D offers the reader a detailed CBM+ foundation that can be used to create successful diagnostics and prognostics. Data fusion algorithms are considered health indicators by the ADS and they follow the same rules of accuracy as condition indicators.
2. Learning from Data; California Institute of Technology (Ref. 19)
   This course is offered simultaneously on campus and as a Massive Open Online Course (MOOC) through the online edX service (www.edx.org) and is a broad introduction to the topic of machine learning. It offers three main focus areas: development of the theory of learning (Vapnik-Chervonenkis Proof), a survey of available models, and ways to protect the training procedures against over-fitting and data snooping. The authors have chosen to use the nomenclature offered by this course as well as the basic models for validation and testing.
3. The Elements of Statistical Learning; Stanford University (Ref. 20)
   This course offers a very detailed explanation of the available learning algorithms currently used in modern machine learning. There is a tremendous focus on the measurement of error, offering more than the typical least squares methodology for model selection. These error methods are being investigated due to their applicability to machine learning problems associated with skewed populations.

Process Terminology

The nomenclature used by the authors to describe the process controls for machine learning is subdivided into the following categories: training inputs (N, p, and Q), output functions (g), and reported error (E).

N: The number of unique examples of component conditions is N. These examples are fully observed by a minimum number of data samples prior to teardown analysis or other equivalent ground truth grading procedure.

p: The generic term for the number of selected and extracted features is collectively known as p.

p^+ and p^*: In order to differentiate between the maximum available input features and a down-selected subset of input features, the authors utilize the following two modifications to p. The maximum input vector is known as p^+, which includes all available aircraft data associated with a component. Any process that reduces the p^+ inputs to a smaller set results in an input vector called p^*. There are many processes by which p^+ may become p^* and these are discussed in the next section.

Q: During the processes of input parameter down-selection, the engineers may determine that multiple input vectors should be trained for various reasons; therefore, the total number of input vectors available for training is Q. Lowercase q is used to denote a specific input vector.

$\mathcal{H}_m$: A model or hypothesis is noted as $\mathcal{H}_m$. There can be any number of variants of a particular hypothesis, usually with different internal training settings and error reduction techniques.

g_{vq}: The result of training a machine to classify component health based on a specific input vector, p^*_q, is the specific function, g_{vq}.

g: The final function determined by the machine learning process is g.

g^-: An intermediate function learned from a k-fold cross validation.

E_{in}: The in-sample error measured internally, and actively reduced during training, is E_{in}.

E_{out}: The out-of-sample error measured externally, and used for reporting based on test data never used during training, is E_{out}.

E_{val}: The error reported during cross-validation and associated with a specific g_{vq} is known as E_{val}. Validation error is a pessimistic bound on E_{out} (Ref. 19) and can be used for reporting unless it is further used to make learning decisions.

E_{vect}: The error reported based on a pre-test of multiple learned functions, g_{vq}, based on different p^*_q inputs, is E_{vect}. This error is a simulation of the actual E_{out} experienced by each function. It can be used for reporting unless it is used to make learning decisions.

The data for learning is required to be representative of the population for which the machine will classify status. Prior to any work, a full understanding of the intended classification purpose and population must be reached so that the learning dataset can be assembled.

Process Overview

The authors have created a process by which a team of engineers reviewing aircraft health data can collectively develop functions that classify input aircraft variables as either healthy or faulted. The process developed allows hundreds of dimensions of aircraft data to be assembled and learned from in a rigorous manner that bounds the out-of-sample error (E_{out}) produced during machine learning. The training process is shown in Figure 1. The remainder of this paper will outline the important steps in Figure 1 and provide the reasons for their selection.

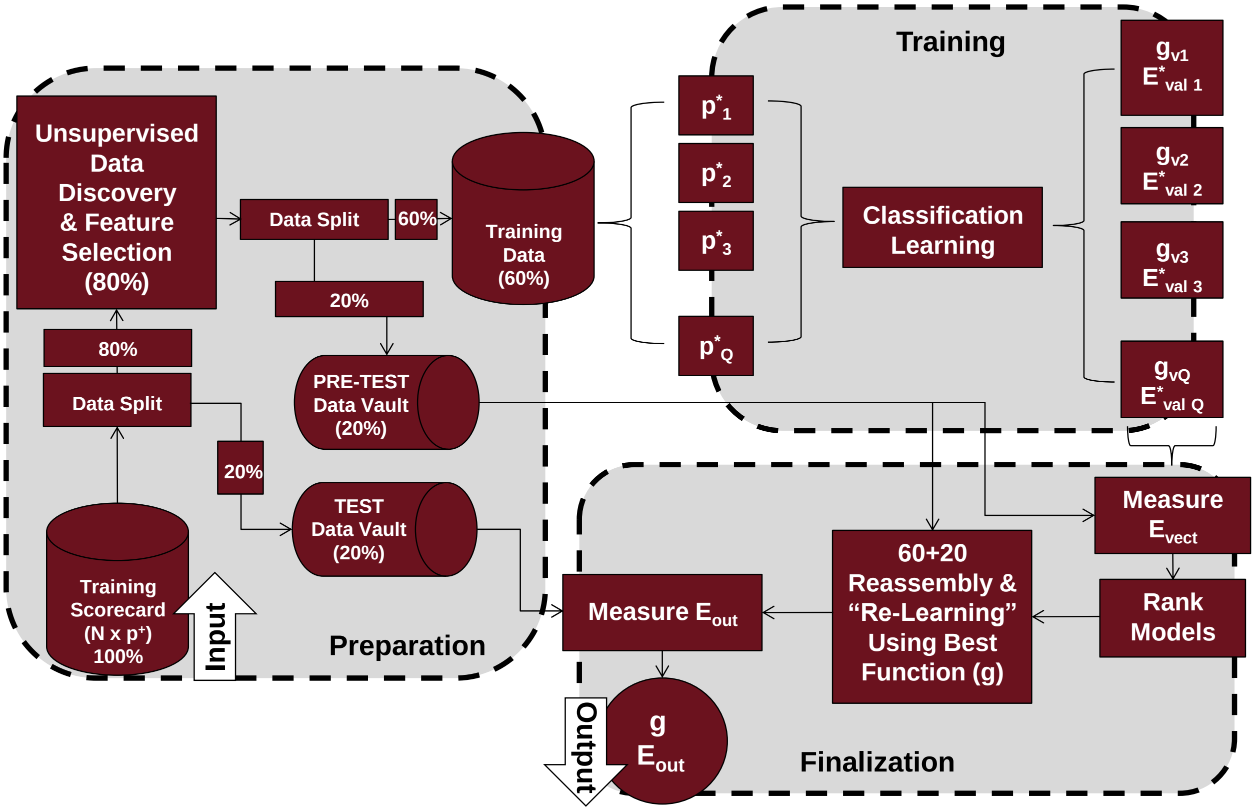


Figure 1. Machine Learning Algorithm Process Flow for a Multi-Disciplinary Engineering Team.

The Preparation block in Figure 1 shows three important steps: N by p assembly, data vault creation, and data discovery. The Training block shows how the different input vectors are input into the classification training schemes and how each of them is associated with an output function and training error. The Finalization block shows how the output functions are graded against each other for selection of a final function for delivery to the customer.

Preparation

Assembly of the N by p matrix is the input into the process which represents the collection of ground truth and associated sensor data. All aircraft and maintenance data that is relevant to aircraft health should be assembled into this matrix. In some cases, missing data (e.g. historical loss) may not adversely affect the output of the developed process. In other cases, however, the missing data may prove to be critical (e.g. malfunctioning sensors) and will prevent the success of machine learning. Until further development of this process can be completed, it is recommended that missing data causes removal of those data sources either as input vector components (p dimension reduction) or as examples of ground truth (N dimension reduction). In future use cases where relatively sparse fully observed ground truth data sets warrant the inclusion of samples with data gaps in the p dimension, data imputation techniques capable of filling these gaps may be explored.

Two data splits are shown in the Preparation block. The first split randomly assigns 20% of the N examples to a test vault for use during the Finalization processes. The data in this vault will never be used for training. The second data split is done after completion of any unsupervised data discovery and feature selection. This split randomly reassigns another 20% of the N examples for transition from large scale training to Finalization.

Unsupervised learning can be used to discover new features in the data or to reduce the size of the p-dimension through transformations of the input space. The two data splits are placed on either side of this discovery process to prevent testing bias. Data used for the final test, which will be reported to the customer and is representative of *real-world* error, should never be used for any form of training (Ref. 19). The second data split contains data that will be used for training during the finalization steps, which also helps prevent bias.

One of the main goals of the unsupervised data discovery process is reduction of the input space to a size that can be used for supervised training. Aircraft HUMS generate thousands of dimensions of data. An example would be the Fourier series coefficients produced by the typical signal processing of vibration data. These spectrums contain thousands of spectral lines from 0 to 50 kHz, which could each be used by a machine for classification. Therefore, the size of p^+ could be nearly nine or ten thousand. For the purposes of training, this is too large relative to the currently available ground truth (tens or hundreds of examples). Engineers must use the CIs and spectrums, along with the parametric data, to gain benefits from machine learning. This will lead to the use of physical parameters directly from the aircraft and unsupervised features learned by a machine.

For the purposes of aircraft HUMS, machine learning algorithms should be used when the length of the p^+ vector is no more than one-tenth the number of the available ground truth examples, N (Ref. 19). There are many unsupervised learning models that can be used to discover important relationships and reduce the dimensionality of the data. Of particular interest to the Army is the fusion of these techniques with the traditional physics-based methods of today.

Aircraft data sources and CIs are determined by the subject matter expert (SME) who designed the data collection system, but CIs are unique in that they are what the SME determines to be important features of the collected data, while the raw data is (nearly) pure sensor data collected by the system. ADS-79 (Ref. 1) states that CIs measure a particular physical degradation of a system, and that they are based in the laws of motion and thermodynamics. From a physics perspective, CIs capture the important features necessary to fully describe the condition of the subject component. These CIs can be either processed on the aircraft from live data, or post-processed on the ground station or other similar software.

Principal Component Analysis (PCA) can be used to create learned CIs from the raw data without review of ground truth. While this process guarantees that the learned CIs will be uncorrelated, it does not guarantee that they will be physically meaningful. This represents a fundamental shift away from the current ADS-79 definition of a CI and towards a more empirical approach. The principal components that result from this process may actually learn mathematical procedures similar to those used by engineers for the creation of a CI, such as a spectral energy-based calculation like root sum square (RSS).

Expectation Maximization (EM) does not assume any relationships between the features of the p^+ vector. EM clustering can reduce multiple CIs to a single classification output and could be used to simplify correlations across unique dimensions of the *CI-space*, or *CI-dimensions* of the N by p^+ matrix. Other approaches, such as clustering and nearest neighbors, exist in the unsupervised learning literature and are being applied to the HUMS dataset. These algorithms typically feature an engineer-determined number of clusters or neighborhoods. Care should be taken when setting the number of clusters so that it is not biased by the engineer.

Unsupervised data reduction ultimately creates p^* vectors that can be passed to the training block in Figure 1 in the following output format. The p^* matrix rows must have a common formatting such that engineers can pass work between one another. The standard formatting is necessary so any function written to transform the p^+ dataset will be standardized for supervised learning. The representation of any p^* row is generalized by the following expression:

$$[Tail][Reps]\left\{\begin{matrix}[RelativeDate],\\ [MetaData],\\ [Features]\end{matrix}\right\}[Category]$$

[Tail] – Tail number of the aircraft associated with the teardown

[Reps] – Historical repetitions of the dataset inside the curly braces

[RelativeDate] – The date of the instances inside the curly braces from time of TDA. The time of the TDA will be represented by zero and all previous measurements inside the braces will be negative time prior to that point. Time should be measured in hours of aircraft usage; however, this is not always available. Calendar date may be used in conjunction with maintenance records to recreate an artificial time history that is meaningful to the HUMS data.

[MetaData] – At minimum, this is the calendar date of the data in the curly braces. It may contain any other selected data that the engineer thinks is important for classification. E.g. Indicated Airspeed, Outside Air Temperature, Engine Torque, Regime Label, etc.

[Features] – Selected or extracted feature(s). E.g. CIs, EM Categories, Principal Components, etc.

[Category] – Healthy or faulted status of the aircraft determined by TDA (0 or 1).

The output of this unsupervised learning results in several models for the p vector. This matrix is referred to as the p^* matrix.

The results of unsupervised learning should be minimally handled by the SMEs to prevent excessive *data snooping* (Ref. 19) (also known as *data dredging*). For example, issues may arise if the SME determines that certain principal components are actually meaningless. Instead, it is more appropriate for the SME to incorporate his or her own model of features that are deemed important to component condition as an appendix to the unsupervised outputs. Obviously, this will result in increased model complexity, but the rules against data snooping must be observed.

Example p^* matrix outputs of the Preparation block:

p^*_0 = Physics-based CIs

p^*_1 = CIs plus the principal components of the spectrum

p^*_2 = Principal components of the spectrums plus the expectation maximization group of all the CIs

Training

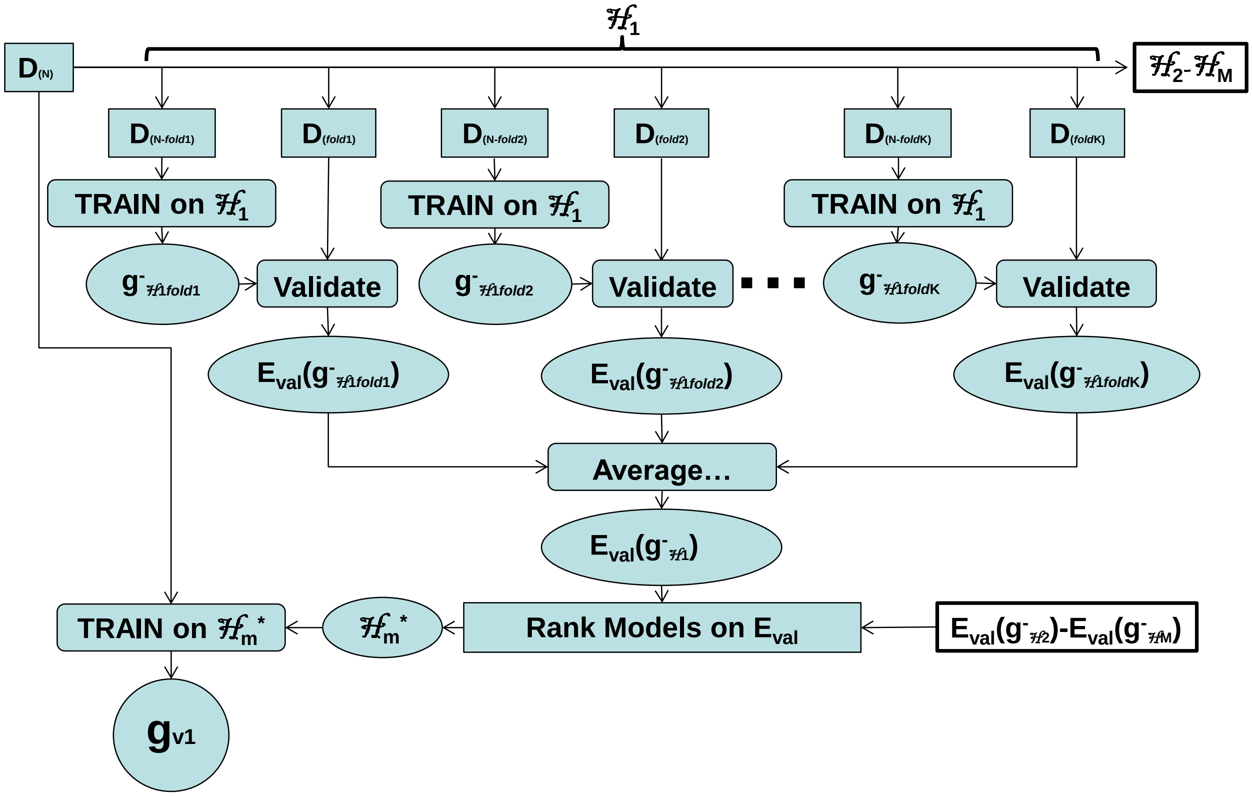
The engineers will assign each of the team members to a particular portion of the p^* matrix. Each of these (p^*) inputs will go through two major processes: Individual Classification Modeling and Ensemble Modeling. The final output of this block will be the learned functions and the error (E_{val}) associated with each of the functions.

K-fold cross validation is used for training unique hypothesis variants ($\mathcal{H}_1$ through $\mathcal{H}_M$) on a particular p^* vector. There can be any number of hypothesis variants that can be a combination of machine learning model types and regularization (λ) techniques. Thus $\mathcal{H}_1$ could be a support vector machine with $\lambda = 0.001$, and H_2 could be the same support vector machine with $\lambda = 0.01$. Details regarding the application of k-fold cross validation can be found in References 19 and 20. A brief explanation of the process as applied by the authors is illustrated in Figure 2.

$\mathcal{H}_1$ through $\mathcal{H}_M$ each have an associated validation error, E_{val}, and function, g^-. Validation error, E_{val}, is a pessimistic bound on the generalization error, E_{out}, for the function $g^-_{\mathcal{H}m}$. With this output, each $\mathcal{H}_m$ creates a single $g^-_{\mathcal{H}m}$ that can be ranked based on the associated E_{val} such that the best hypothesis can be chosen. The model, $\mathcal{H}_m$, with the smallest validation error, E_{val}, is chosen.

Ranking based on validation error, E_{val}, creates the opportunity to reassemble the k-folds such that learning can be accomplished on the best hypothesis using the entire dataset. This creates a new function, $g_{vq,}$ which can be output to the Finalization block. The reassembly of the k-folds and then final training of the winning hypothesis nullifies the pessimistic bound on generalization. Therefore E_{val} can no longer be reported to the customer as a bound on E_{out}, but the benefits of reassembly are more valuable than the loss of the error bound because this process maximizes the data for creating a good output function. This is why the Finalization block contains two data vaults, so that the opportunity to provide E_{out} to the customer still exists.

The output of the k-fold process is a unique g_{vq} and its associated E_{val} which can then be compared against the other functions ({g_{v1} g_{v2} g_{v3} g_{v4} … g_{vQ}}) produced by each of the unique input vectors from p^*.



$\mathcal{H}_1$
$D_{(N)}$
$\mathcal{H}_2$-$\mathcal{H}_M$
$D_{(N-fold1)}$
$D_{(fold1)}$
$D_{(N-fold2)}$
$D_{(fold2)}$
$D_{(N-foldK)}$
$D_{(foldK)}$
TRAIN on $\mathcal{H}_1$
TRAIN on $\mathcal{H}_1$
TRAIN on $\mathcal{H}_1$
$g^-_{\mathcal{H}1fold1}$
Validate
$g^-_{\mathcal{H}1fold2}$
Validate
$g^-_{\mathcal{H}1foldK}$
Validate
$E_{val}(g^-_{\mathcal{H}1fold1})$
$E_{val}(g^-_{\mathcal{H}1fold2})$
$E_{val}(g^-_{\mathcal{H}1foldK})$
Average…
$E_{val}(g^-_{\mathcal{H}1})$
TRAIN on $\mathcal{H}_m^*$
$\mathcal{H}_m^*$
Rank Models on E_{val}
$E_{val}(g^-_{\mathcal{H}2})$-$E_{val}(g^-_{\mathcal{H}M})$
g_{v1}

Figure 2. Detailed K-Fold Cross Validation Process for Model Creation.

Finalization

After the engineers have produced all expected models from all the expected hypotheses, the 20% Pretest Data Vault is opened so that the functions can be compared against each other and ranked by the new data that is available. Ranking is done by comparing each model's E_{vect}. The highest ranked g_v (lowest E_{vect}) is chosen for re-learning, which is accomplished by assembling the 60% training data and the 20% pretest data together. As was done in the k-fold cross validation, the data is assembled so that a final output function, g, can be learned using 80% of the training data.

When the final function is output, the test vault is opened and used to demonstrate the out-of-sample error associated with the machine. All training and learning must be stopped after this vault is opened to prevent bias based on the results. E_{out} is reported to the customer, along with the function, g.

FUTURE WORK

The engineering team that is applying these techniques is currently working towards the following objectives:

1. Generate a fused diagnostic for the Apache Nose Gearbox that (a) incorporates *dynamic thresholds* and all vibration based CIs, and (b) meets or exceeds ADS-79 performance standards.
2. Generate a fused diagnostic for low-power removals of the T-700 series turbine using parametric data and usage history.
3. Generate a fused diagnostic for the Black Hawk Main Transmission that (a) incorporates *dynamic thresholds* and all vibration based CIs, and (b) meets or exceeds ADS-79 performance standards.

CONCLUSIONS

Upon review of the history of vibration-based diagnostics for Army rotorcraft HUMS, it has been determined that machine learning approaches may be used to improve performance and reduce soldier burden. Several sources have been surveyed by Army aerospace engineers and a process flow for using machine learning has been created. The proposed process allows a multi-disciplinary engineering team to pass HUMS data between unsupervised and supervised learning models. The process is designed to allow multitudes of learning models to be validated and tested rigorously. Controls on bias have been implemented such that out-of-sample error can be measured prior to field implementation based on existing aircraft data. The output function of the proposed process can be used for maintenance decisions and logistics planning.

REFERENCES

[1] AMCOM Standardization Technical Data Management Division ADS-79D-HDBK, "Condition Based Maintenance for US Army Aircraft Systems", http://www.amrdec.army.mil/amrdec/rdmr-se/tdmd/StandardAero.htm, 2013.

[2] Keller, J.A., and Grabill, P., "Vibration Monitoring of UH-60A Main Transmission Planetary Carrier Fault," Proceedings of the 59th Annual Forum of the American Helicopter Society, Phoenix, AZ, May 2003.

[3] Keller, J.A., Branhof, R., Dunaway, D., and Grabill, P., "Examples of Condition Based Maintenance with the Vibration Management Enhancement Program," Proceedings of the 61st Annual Forum of the American Helicopter Society, Grapevine, TX, June 2005.

[4] Lewis, W.D., Perry, C.D., and Keller, J.A., "Airworthiness Releases as a Result of Condition Based Maintenance," European Rotorcraft Forum, Maastricht, The Netherlands, 2006.

[5] Suggs, D., and Wade, D., "Utilizing On-Aircraft Distributed Fault Data to Improve the Removal Decision Process," Proceedings of the 63rd Annual Forum of the American Helicopter Society, Virginia Beach, VA, May 2007.

[6] Suggs, D., and Wade, D., "Vibration-Based Maintenance Credits for the UH-60 Oil Cooler Fan Assembly", Proceedings of the 1st American Helicopter Society CBM Specialists' Meeting, Huntsville, AL, February 2008.

[7] Dempsey, P., Keller, J., and Wade, D., "Signal Detection Theory Applied to Helicopter Transmission Diagnostic Thresholds," Proceedings of the 65th Annual Forum of the American Helicopter Society, Grapevine, TX, May 2009.

[8] Dempsey, P., Branning, J., Wade, D. and Bolander, N., “Comparison of Test Stand and Helicopter Oil Cooler Bearing Condition Indicators,” Proceedings of the 66th Annual Forum of the American Helicopter Society, Phoenix, AZ, May 2010.

[9] Antolick, L., Branning, J., Wade, D., and Dempsey, P., “Evaluation of Gear Condition Indicator Performance on Rotorcraft Fleet,” Proceedings of the 66th Annual Forum of the American Helicopter Society, Phoenix, AZ, May 2010.

[10] Wade, D., and Branning, J., “Application of Aeronautical Design Standard Specification 79 to the UH-60L Accessory Gearbox Generator Drive Bearings,” Proceedings of the 3rd American Helicopter Society CBM Specialists’ Meeting, Huntsville, AL, February 2011.

[11] Keller, J.A., Carr, D.J., and Grabill, P., “Analysis of AH-64 Tail Rotor Swashplate Bearing Failures,” Proceedings of the 1st Annual American Helicopter Society CBM Specialists’ Meeting, Huntsville, AL, February 2008.

[12] Krick, S., Wade, D., Pipe, K., “Evaluation of a Novel Adaptive Thresholding and Trend Alert Generation Technology on a HUMS Equipped Fleet,” Proceedings of the 68th Annual Forum of the American Helicopter Society, Fort Worth, TX, May 2012.

[13] Krick, S., and Wade, D., “Development of Engineering Standards for Dynamic Thresholding and Trend Alert Technology for Application to a HUMS-Equipped Fleet,” Proceedings of the Fourth American Helicopter Society CBM Specialists Meeting, Huntsville, AL, February 2013.

[14] Wade, D., and Larsen, C., “Measurement of Gearbox Surface Frequency Response Functions for HUMS Algorithm Improvement,” Proceedings of the 68th Annual Forum of the American Helicopter Society, Fort Worth, TX, May 2012.

[15] Antolick, L., Wade, D., and Brower, N., “Application of Advanced Vibration Techniques for Enhancing Bearing Diagnostics on a HUMS-Equipped Fleet,” Proceedings of the Fourth American Helicopter Society CBM Specialists Meeting, Huntsville, AL, February 2013.

[16] Szelistowski, M., Shepard, S., and Wade, D., “An Impulse Response Driven Approach for Optimizing Bearing Enveloping Diagnostics,” Proceedings of the 69th Annual Forum of the American Helicopter Society, May 2013.

[17] Sheldon, J., Kasper, D., Davis, M., “A Multilayered Approach for Enhancing Rotorcraft Drive System Diagnostics,” Proceedings of the 70th Annual Forum of the American Helicopter Society, Montreal, Quebec, Canada, May 2014.

[18] Lesobre, R., Bouvard, K., Berenguer, C., Barros, A., Cocquempot, V., “A Maintenance Free Operating Period for a Multi-Component System with Different Information Levels on the Components State,” *Chemical Engineering Transactions,* Vol. 33, 2013. ISBN 978-88-95608-24-2.

[19] Abu-Mostafa, Y. S., Magdon-Ismail M., Lin H. T., *Learning From Data*, AMLbook.com, Pasadena, CA, 2012, Chapter 2, Chapter 4, Chapter 5.

[20] Hastie, T., Tibshirani, R., Friedman, J., *The Elements of Statistical Learning*, Springer, New York, NY, 2009, Chapter 2, Chapter 5, Chapter 7.