

# On-Demand Regression to Improve Preciseness of Time to Failure Predictions

Sylvain Létourneau and Chunsheng Yang and Zhenkai Liu

Institute for Information Technology  
National Research Council Canada  
Ottawa, Ontario, Canada, K1A 0R6  
{*firstname.lastname*}@nrc-cnrc.gc.ca

## Abstract

Despite the availability of huge amounts of data and a variety of powerful data analysis methods, prognostic models are still often failing to provide accurate and precise time to failure estimations. This paper addresses this problem through an innovative approach integrating several machine learning algorithms. The approach proposed relies on a classification system to determine the likelihood of component failures and to provide rough indications of remaining life. It then introduces clustering and SVM-based local regression to refine the time to failure estimations provided by the classification system. The paper illustrates the applicability of the proposed approach through a real world aerospace application and discusses data pre-processing requirements. The preliminary results show that the proposed method can reduce uncertainty in time to failure estimates, which in turn helps augment the usefulness of prognostics.

## Introduction

The need for higher equipment availability and lower maintenance cost is driving the development and integration of prognostic and health management (PHM) systems. Taking advantage of advances in sensor technologies, PHM systems favor a pro-active maintenance strategy by continuously monitoring data from the equipment and informing the maintenance staff whenever there is a risk for a component failure. A PHM system may also supplement the component failure predictions with an estimation of the time to failure (TTF), which is defined as the expected remaining time before the given component stops fulfilling its function. In order to avoid disruption and minimize maintenance costs, these time to failure estimates need to be as reliable and as precise as possible.

Traditional methods to estimate TTF include reliability analysis [8] and knowledge-based approaches from physics and material sciences [1, 2, 7, 9]. These approaches help to understand the underlying physical mechanisms but they require enormous amounts of background information. They may also be difficult to apply as they tend to rely on difficult to obtain data on component damage or material properties. With the development and integration of data acquisition devices into complex equipment, data mining-based

approaches are now starting to complement the traditional methods for building prognostic models [5, 6]. Recent results show the potential of classification systems to identify the likelihood of component failures in a timely manner but none of the existing techniques can provide precise time to failure estimates required for the optimization of maintenance. For instance, the KDD methodology proposed in [5] can build classification models for prognostics [10]. These models continuously assess the probabilities of a component failure within a pre-specified alert target window (e.g., between 1 and 20 days in advance of a functional failure), but often fail to provide precise TTF estimates. When a classifier detects patterns in the data that are characteristic of an incipient failure, it generates an alert indicating that the suspected component is likely to fail within the alert target window without being able to specify the exact number of days or hours of operation left. With this approach, the larger the alert target window, the larger the imprecision on the TTF estimates. In some specific applications, it is reasonable to try to increase precision by reducing the width of the target window. However, this is generally not suitable as it could prevent the end users from getting alerts as early as possible which, in turn, would reduce the opportunity for optimization and the benefits of prognostics. A too narrow target window may also have detrimental effects on the performance of the predictive models. For instance, when a component has various failure modes, each following their own time frame, there is a risk that a model specific to a narrow target window would only be able to detect a fraction of these failure modes.

Since predicting TTF can be seen as a regression problem, regression analysis and time-series forecasting methods could be used to build models that try to directly estimate TTF from the sensor data. To be successful, such models need to accurately map all the subtle changes in the data to specific life reduction estimates. These models also need to account for the fact that with complex components, we often observe significant variations in actual time to failure. Obviously, building such models is a very challenging task that requires ample amounts of high quality and relevant data. Since data from real world equipment is typically characterized by issues such as irregular sampling intervals, small signal/noise ratio, and sensor measurement errors, it is generally hopeless to try to develop a global regression model

for TTF from sensor data. On the other hand, it is plausible that regression could be successfully applied locally on well chosen portions of the real world sensor data. This paper investigates this hypothesis by trying to demonstrate that regression analysis can help improve the preciseness of TTF estimates.

This paper proposes to improve the preciseness of TTF estimates by combining classification and regression-based approaches. It relies on a comprehensive data mining methodology to develop the required classification system. The classifier developed is capable of identifying incipient component failures and providing rough TTF estimates. Clustering is used to partition the sensor data and a regression model is developed to estimate TTF within each cluster. When the classifier uncovers a potential component failure, a mapping function decides which regression model should be used to provide a TTF estimate. A final step produces the final TTF estimation based on the output from the classification and regression models. We name the proposed method “on-demand regression” since regression is only used once the classification system has identified the potential for a component failure.

Before detailing the approach, the paper discusses the difficulty of the task with support from real world data from an aerospace application. The same application is also used to illustrate the applicability and the usefulness of the proposed approach.

## Challenges

Accurate and detailed health information on key systems and components is of utmost importance to help optimize the maintenance and management of complex equipment. Ideally, powerful prognostic models, well integrated into the organization’s information system, would automatically combine sensor data, historical maintenance information, system’s configuration, and other sources of information to continuously provide accurate and precise TTF information. Regression methods, which are specifically designed to predict numerical values such as TTF, appear well suited to develop these models. Unfortunately, many typical issues of real world data from complex equipment severely constrain the applicability and power of regression modeling. To illustrate, let us consider an aerospace application in which the objective is to build a prognostic model for the starter motor of the Auxiliary Power Unit engine (APU).

The data for this application has been produced by a fleet of 35 commercial aircraft over a period of 10 years. The specific dataset used consists of 18 attributes (5 symbolic, 11 numeric, and 2 for date and time of the event). More than 161000 observations are available for this task. Only a subset of these observations are relevant for learning the predictive models. These are the ones collected around each occurrence of component failures. In this particular task, we use engine operating hours as the time unit. We based our analysis on data generated between 250 operating hours prior to the failure and 30 hours after. A comprehensive search in the maintenance database revealed information on 85 occurrences of APU starter motor replacements. Since we do not have access to information from further testing

of the components replaced, we assume that a replacement is equivalent to an actual failure. When an engine suffered consecutive failures in a short period of time, we constrained the above interval to make sure that each observation is included only once. We used data from 60 failures for learning and kept data from the remaining 25 failures for testing.

To evaluate the feasibility of regression as a direct way to predict TTF, we augmented the initial representation with a TTF attribute. This attribute is simply defined as the difference between the engine operating hours in the current observation and the operating hours of this engine at the next starter failure. To make the regression task easier, we removed the instances observed after the failures. We built an SVM-based regression model using the training data set and then applied it on the testing set. Figure 1 shows results from one of the best SVM (Support Vector Machine) models developed. The scatter clearly illustrates the lack of fit of the model. The expected error from this model is 58.7 with a standard deviation of 43.8. Other regression methods such as NN (Neural Network) and linear regression lead to similar performance. The following paragraphs discuss some of the reasons that explain the lack of success of global regression models.

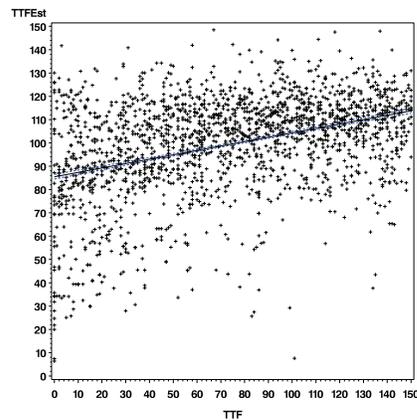


Figure 1: Predicted TTF versus actual TTF from a global SVM-based regression model.

**Irregular sampling rate** Most traditional forecasting techniques require a fixed interval between measurements (e.g., every hour, every day, or every month). In this particular application, the data is systematically collected at every start of the APU. On the other hand, the data is not sent to the central system unless the on-board system decides to do so. Due to different configurations across the fleet and over time, we ended up having APU data at every start for some of the aircraft, at every other start for other aircraft and so on up to every 8 starts. As a result, some forecasting methods can simply not be applied. Unequal sampling rate also means that the time-series corresponding to the different failures have different lengths. This may lead to an unbalanced representation of the failure cases. A possible solution is to re-sample through interpolation or smoothing but given the high variability in the original sampling rate

and the small signal/noise ratio, such processing is likely to hurt the modeling for TTF estimation.

**Lack of relevant information** In order to accurately predict remaining life, the model needs to be able to estimate the current life consumption. The information required to evaluate life consumption could come from highly informative core measurements that adequately account for the internal state of the component. Such high value information is typical from laboratory testing equipment but rarely available from sensors deployed on today’s complex equipment. Information about life consumption can also be captured directly by means of a counter. For instance, in the APU starter motor application, we use the engine operating hours to approximate life consumption but it is far from perfect since the starter motor and the engine may consume their life differently. Moreover, engine overhauls which happen at regular intervals result in a reset of the engine operating hours counter. Since we do not have access to detailed information on work performed during the overhaul, it is impossible to determine if the reset of the engine operating hours counter also corresponds to the repair of the starter motor. Consequently, these resets can possibly introduce cuts in the evaluation of life consumption and cause great difficulty for the regression models.

**Large variance due to contextual effects** Equipment such as aircraft operates in a very dynamic environment. Changes in this environment affect the behavior of the system. In some cases, these changes also affect the measurements taken. For instance, all measurements related to temperature, flow, and pressure are likely to be affected by the altitude of the aircraft. The mode of operation and the status of internal sub-systems and components are also likely to affect the behavior of the performance parameters. All of these contextual effects need to be accounted for in order to understand the behavior of the key parameters and properly use them to infer reliable TTF estimates.

Notwithstanding all of the difficulties mentioned above, this paper argues that regression can still play a role in helping to improve the TTF estimates in prognostic applications. As explained in the following section, the main idea is to partition the data space into relatively homogeneous data subsets and then use different regression models for these subsets.

### On-demand regression

Figure 2 illustrates the proposed hybrid approach to improve the preciseness of TTF estimates. The tree steps are described below.

#### Classification-based prognostic

The first step involves a binary classifier that can identify incipient component failures and provide a rough estimate of the remaining useful time. We build this classifier using the KDD methodology documented in [5, 10]. This methodology consists of several steps which we now succinctly describe.

In order to use classification learning, we need to add a class attribute to the sensor data. We proceed with an automated approach. This approach labels as positive (“1”) all

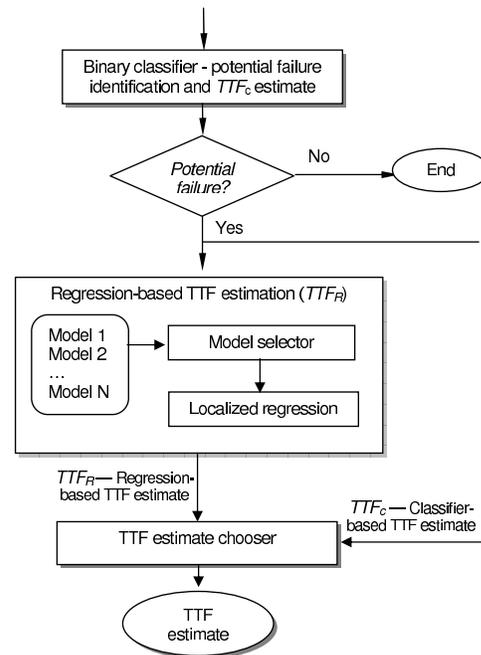


Figure 2: Overview of the hybrid approach proposed to improve TTF estimates.

instances that fall in a pre-determined target window before the occurrence of a starter motor failure and as negative (“0”) all other instances. This labeling scheme allows us to build a classifier that generates an alert whenever the patterns in the data are similar to the ones observed near a failure. In practice, we define the length of the target window by taking into account the optimal period for the end users to receive the alerts and the balance between positive and negative instances. As a rule of thumb, we try to keep a minimum of 15% as positive instances to simplify the learning.

Since data representation is often a key factor, we systematically try to improve the initial representation by augmenting it with new informative features. We construct these features using methods from signal processing, time-series analysis, and constructive induction. Feature selection is also applied on the augmented data representation to automatically remove correlated or irrelevant features [3, 4].

After updating the initial dataset with the class attribute and incorporating data representation enhancements, we build the required classifier. We use data from a subset of all failures for learning the models and keep the remaining data for testing. Any classifier learning algorithm can be used. In early experiments, we tended to prefer simple algorithms such as decision trees and naive-Bayes over more complex ones because of their efficiency and because they produce models that we can easily explain to the end users. We apply the same algorithm several times with varying attribute subsets and cost information. To compare the classifiers obtained, we apply a score-based approach that we have developed to evaluate classifiers for prognostic systems [5, 11]. The one with the maximal score on testing data is selected

as the best classifier.

As illustrated in Figure 2, the classifier needs to provide a rough TTF estimate ( $TTF_C$ ) whenever it predicts a potential component failure. We define this TTF estimate based on the expected number of operating hours left between a positive prediction and the actual failure. We use only the training data to compute this expected value. Precisely,  $TTF_C = 1/N \sum_{i=1}^N RemainingOPH_i$  where  $RemainingOPH_i$  is for the difference between the engine operating hours in the  $i^{th}$  positive prediction from the training set and the operating hours of the corresponding engine at the next starter failure, and  $N$  is the number of positive predictions made by the classifier on the training data set. This value is constant for all positive predictions made by the classifier.

### Regression-based TTF estimation

The objective of the second step is to try to improve the preciseness of the TTF estimates provided by the classifier described above. This is done through localized regression models. Each model accounts for a specific area of the data space. Every time the classifier makes a positive prediction, one of the local regression models is selected and applied to compute a new TTF estimate. The construction of the models required goes as follows.

First, clustering is used to partition the time-series associated to the various failures. The intent is to obtain clusters of time-series as homogeneous as possible with respect to the performance of the core measurements. This is done by clustering based on the attributes that represent meaningful contexts for the component of interest. In other words, we use clustering to obtain subsets into which the potentially negative effect of contextual contexts is minimized. In the case of the APU starter motor application, the predominant contextual attribute is the age of the starter motor at the time of the failure. As explained above, we approximate this age using the engine operating hours at the time of the failure. To help optimize within cluster homogeneity, we create as many clusters as possible while ensuring that each cluster has sufficient data to train and evaluate the regression model. Using the same train and test data as in the step above, we assess the viability of a given clustering by counting how many time-series from training and testing data fall in each cluster. In our experiments, we decided that each cluster should contain at least 2 time-series for training and 1 for testing. This rule forced us to use a maximum of 10 clusters.

Second, a model selector is developed in order to assign each positive prediction to a given data subset (Fig 2). The clustering model built for partitioning the data cannot be deployed for this task as it relies on the operating hours at the failure time, which is unknown for yet to fail components. We resolve this issue with an  $N$ -class classifier, where  $N$  is the number of clusters. Once the clustering scheme has been established, we tag each instance with its cluster ID and learn a classifier that can tell apart the instances as accurately as possible using the measurements available. Based on our experiments, simple decision trees and naive-Bayes classifiers perform very well for this task with a typical accuracy of 90% on test data.

	Total	Train	Test
		70%	30%
No. of failures	85	60	25
No. of instances	11248	8219	3029

Table 1: Summary of the data used in the APU starter motor prognostic application.

Finally, a regression model is built for each cluster. SVM, NN, and many statistical regression techniques can be used to build these models. A key aspect of this step is that it uses only a subset of the training data available in each cluster to learn the models. Precisely, it uses only the instances where TTF (i.e., remaining engine operating hours) is less than the expected TTF from the classifier. This allows us to further limit the scope of the regression models to the areas with the greatest potential for enhancing the precision of TTF estimates. The evaluation procedure starts by running the classifier on the test data to identify positive predictions. For each of these positive predictions, the model selector chooses an adequate regression model. The chosen model computes the regression-based TTF estimate noted  $TTF_R$ .

### Selecting which TTF estimate to use

Two TTF estimates are produced for each positive prediction: one from the classifier ( $TTF_C$ ) and one from a local regression model ( $TTF_R$ ). We now need to decide how to combine them into a single TTF estimate. Our approach is very simple; it returns  $TTF_R$  if  $TTF_R < TTF_C$  and it returns  $TTF_C$  otherwise. This is to avoid potentially significant errors that could come from an extrapolation of a regression model.  $TTF_C$  corresponds to the value used to limit the range of the output attribute while learning the regression models. If at deployment (or testing) time, a regression model outputs a prediction that is higher than the maximal value observed during training (i.e.,  $TTF_C$ ) then the model is extrapolating. Since extrapolation from local models is risky, we prefer to disregard such predictions and rely on the default classifier-based estimate.

## Application

This section reports preliminary results on the application of the proposed method to try to improve the preciseness of TTF estimates for prognostic of APU starter failures on commercial aircraft. Detailed information about the data have been discussed in the *Challenges* section. Table 1 summarizes the training and testing data sets. All models have been built using the WEKA package.

The classifier for identifying potential failures was built using the J48.PART algorithm with default options. We configured the automatic labeling step so that it tags all observations with remaining engine operating hours less than 50 hours as positive and all others as negative. This provides sufficient time for the maintenance staff to plan the repair of the starter prior to the actual failure. In this experiment, we used only the raw measurements without any data representation enhancement. We also used default cost information.

The classifier accurately predicts 21 failures out of the 25 test cases by generating at least one positive prediction for each of them in the target period. The expected TTF estimate from this classifier is 47.8 hours. As reported in the first line of Table 2, the average error of the TTF estimates from this model alone on the test data is 28.3 with a standard deviation of 13.6. Figure 3 shows the graph of the TTF estimates versus actual TTF when using only the classifier. We notice a straight line at 47.8 which is the estimate that this model returns for all positive instances from the test data set.

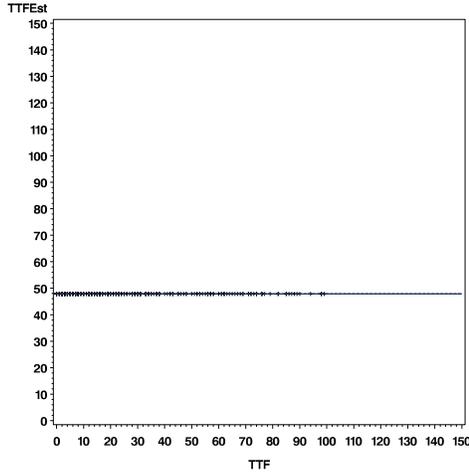


Figure 3: Predicted TTF versus actual TTF using only the binary classification system.

As mentioned earlier, the operating hours at the failure time was used to partition the 85 time-series (one for each failure case) into 10 clusters. Results reported are based on K-means. Experiments with EM produced almost identical results. The model selector was built using J48. Its accuracy on test data is slightly above 85%. We used SMOReg with a linear polynomial kernel to construct the 10 local regression models. If we assume that these models would be used to generate all TTF estimates, then the results would be as illustrated in Figure 4. The scatter in the graph shows the lack of fit between many of the estimates and the actual TTF values. This is also confirmed by the second line in Table 2 which reports an average error of  $50.9 \pm 45.3$  hours.

The results from the on-demand regression approach are presented in Figure 5 and in the last line of Table 2. With this approach, we observed a much better fit between the estimates and the actual TTF values. With an average error of  $9.9 \pm 10.5$  hours and a MSE of 206, the proposed approach clearly outperforms the initial classification-based approach. The strong reductions in the average error and in the standard deviation suggest an improvement in preciseness of TTF estimates by a factor of 3.

## Conclusion

This paper presented a method that we are developing to improve the preciseness of time to failure estimates for prognostic. Relying on real world data from an aerospace appli-

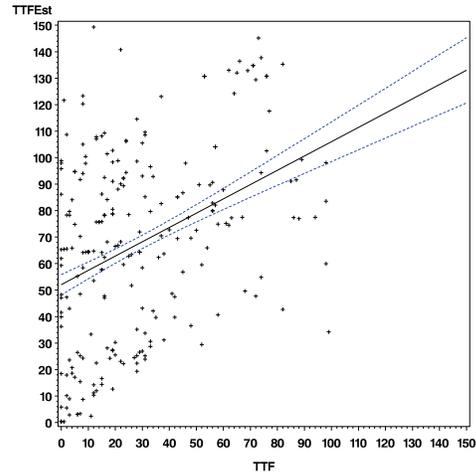


Figure 4: Predicted TTF versus actual TTF using estimates from the regression-based models only.

TTF prediction method	Average error	Std dev.	MSE
Classifier-based	28.3	13.6	988
Regression-based	50.9	45.3	4522
On-demand reg.	9.9	10.5	206

Table 2: TTF estimates on test data using the classification system, the regression-based, and the on-demand regression approach.

cation, the paper describes the difficulties limiting the usefulness of regression for TTF estimation. In spite of these difficulties, the paper argues that regression can help improve TTF estimates. The method introduced to demonstrate this is named “on-demand regression”. It carefully integrates classification-based prognostic, clustering, and local regression. The paper fully describes the process followed to build the various models involved and report preliminary results on a real-world application. These results show the great potential of the approach for the preciseness of TTF estimates in prognostic applications.

## References

- [1] Gebraeel, N.; Lawley, M.; Liu, R.; and Parmeshwaran, V. 2004. Residual life prediction from vibration-based degradation signals: A neural network approach. *IEEE Transactions on Industrial Electronics* 51(3):692–700.
- [2] Grottko, M., and Trivedi, K. 2005. On a method for mending time to failure distributions. In *Proceedings of International Conference on Dependable Systems and Networks*, 560–569.
- [3] Hall, M. 2000. Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the 17th International Conference on Machine Learning*, 359–366.
- [4] Kira, K., and Rendell, L. 1992. A practical approach to

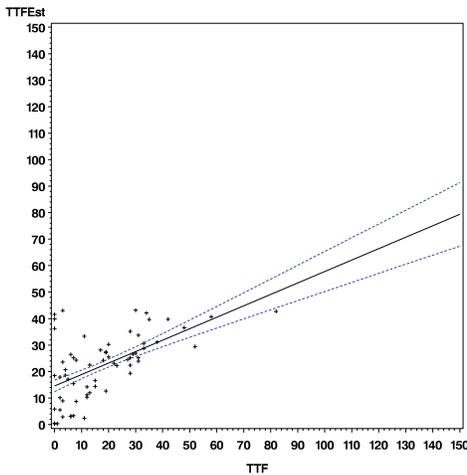


Figure 5: Predicted TTF versus actual TTF from the proposed on-demand regression approach.

feature selection. In *Proceedings of the 9th International Conference on Machine Learning*, 249–256.

- [5] Létourneau, S.; Yang, C.; Drummond, C.; Scarlett, E.; Valdés, J.; and Zaluski, M. 2005. A domain independent data mining methodology for prognostics. In *Proceedings of the 59th Meeting of the Society for Machine Failure Prevention Technology, MFPT '05*.
- [6] Létourneau, S.; Famili, A. F.; and Matwin, S. 1999. Data mining for prediction of aircraft component replacement. *IEEE Intelligent Systems and their Applications* 14(6):59–66.
- [7] Promellec, B.; Riant, I.; and Tessier-Lescourret, C. 2006. Precise life-time prediction using demarcation energy approximation for distributed activation energy reaction. *Journal of Physics: Condensed Matter* 18(2006):2199–2216.
- [8] Schouten, F. A. V. D. D., and Wartenhorst, P. 1980. Time to failure, time to repair and availability of a two unit standby system with markovian degrading units. *Technical report BS-R9007, Center for Mathematics and Computer Science, Amsterdam, The Netherlands*.
- [9] Sornette, D., and Andersen, J. V. 2006. Optimal prediction of time-to-failure from information revealed by damage. *Europhysics Letters* 74(5):778–784.
- [10] Yang, C., and Létourneau, S. 2005. Learning to predict train wheel failures. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2005)*, 516–525.
- [11] Yang, C., and Létourneau, S. 2007. Model evaluation for prognostics: Estimating cost saving for the end users. In *Proceedings of the 6th International Conference on Machine Learning and Applications (ICMLA 2007)*.