Coupling a Dynamic Linear Model with Random Forest Regression to Estimate Engine Wear

James Schimert[1] and Art Wineland[2]

[1] The Boeing Company, Bellevue, WA, 98008, United States
[1] The Boeing Company, Huntington Beach, CA, 92647, United States
james.schimert@boeing.com
art.wineland@boeing.com

ABSTRACT

Mechanical systems wear or change over time. Data collected over a system's life can be input to statistical learning models to predict this wear/change. A research effort at The Boeing Company has modeled gas turbine engine exhaust gas temperature (EGT) as a function of other recorded parameters. Our investigation chose gas turbines, but these techniques could also be used for other systems that slowly change (degrade) over time. Previous work trained a flexible empirical regression model at a fixed point of wear, and then applied it independently at time points over the life of an engine to predict wear. However, wear typically occurs slowly and smoothly. This paper describes the benefit of relating wear predictions over time using a dynamic linear model, which is an example of a state space method. The combined model predicts wear with dramatically reduced variability over both our previous effort and a baseline method. The benefit of reduced variability is that engine wear is more evident, and it is possible to detect operational anomalies more quickly. In addition to tracking wear, we also use the model as the basis for a Bayesian approach to monitor for sudden changes and reject outliers, and adapt the model after these events. Experiments compare methods and give some guidance in applying the methodology.*

1 INTRODUCTION

One goal of any integrated vehicle health management program for aircraft is to monitor engine health, in particular, engine wear. Wear affects engine performance. For example, to get the same thrust output as the engine wears, the engine requires more fuel, and so the engine's exhaust gas temperature (EGT) increases. However, environmental, flight, and other engine parameters also affect EGT. These other factors can affect EGT much more than wear for a given data point. As seen in Figure 1, a time series plot of EGT over the life of an engine does not show an increasing trend, as would be expected due to wear.

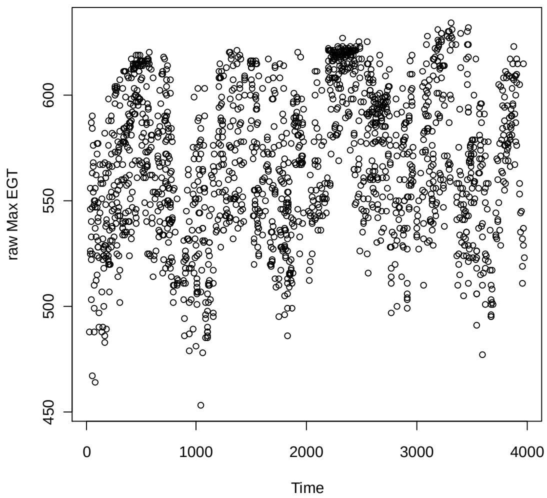


Figure 1: Raw EGT measurements plotted over the lifetime of an engine. The cyclical trend (possibly due to air temperature) obscures the increase in EGT trend due to wear.

Industry practice is to *normalize*, i.e. adjust EGT to a baseline set of environmental and flight conditions. In plots of normalized EGT, the effect of wear on EGT is much more evident. See Figure 2 for an example. When normalized EGT exceeds a threshold temperature, it could trigger a decision to schedule maintenance or take other corrective actions.

Indeed, Figure 2 is a plot of *margin*: the threshold temperature minus normalized EGT. Here, a trend is apparent that is lacking in Figure 1. It is possible to identify several phases: break-in, prime performance life, and end. However, there is more variability than is desirable to accurately determine whether the normalized EGT is close to a threshold temperature, or equivalently, the margin is close to 0, at which time the engine needs to be serviced. The scatter is so large that there is little confidence in any one point. Reducing the variability would save costs by enabling more accurate maintenance decisions. Our goal is therefore to reduce variability in predicted wear.

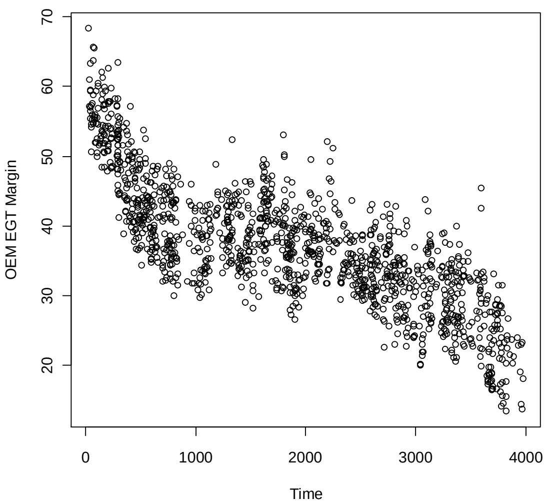


Figure 2: OEM EGT *margin* over the same time period of the same engine. Here, a trend is apparent that is lacking in Figure 1.

This paper discusses a research effort at The Boeing Company, which is investigating a data driven approach to reduce the variability of normalized EGT by accounting for both (1) the effect of other variables and (2) time dependence. The approach builds a data driven model using volumes of flight data that are increasingly collected routinely on modern aircraft. Such a data driven approach contrasts with a physics model approach developed using physics/engineering insight and test data.

As a baseline for comparison, a Boeing customer provided normalized EGT. For brevity, we denote this data by "OEM". A previous investigation (Basu et al., 2008a, b) showed that a data driven approach outperformed the OEM results in the sense that its predictions (using a random forest (Breiman, 2001a)) had a similar range for engine wear, but about 25% smaller variation.

This paper discusses how coupling a state space approach with the random forest further reduces variability by accounting for time dependence. This paper also presents preliminary work on a monitor based on the model. The monitor is an important step toward fault detection and prognosis.

Similar normalization problems occur in other contexts. One example is predicting tire pressure loss. The ultimate goal may be to monitor wear or degradation as equipment is used, in order to repair or replace the equipment in a timely manner. The common elements of our approach to such problems are to (1) adjust a quantity of interest for other influences, and (2) relate these adjustments over time.

We start by briefly giving some technical background on adaptive nonparametric regression and state space methodology. Next, we give the rationale for fitting a regression model to data collected at baseline conditions, producing residuals, and relating these over time using a dynamic linear model. Several experiments show that using dynamic linear models (DLMs) to relate predictions over time dramatically improves a previously developed data driven approach (Basu *et* al., 2008 a, b), which itself improves upon the OEM approach. Promising preliminary work employs a Bayes factor (similar to likelihood ratio) to detect outliers, monitor for sudden shifts, and automatically reject outliers and adapt the DLM to changes. Finally, we discuss the generality of the solution for problems involving sequences of predictions, and suggest future work.

2 TECHNICAL BACKGROUND

2.1 Trees and Ensembles for Regression

Focusing the discussion on regression, a variety of strategies relate a response variable y to a set of covariates x. The classical approach is to postulate a parametric function to predict y at the point x. Harnessing increasingly powerful yet cheap computer resources is one alternative to assuming a parametric model.

Tree-based methods are an example of adaptive nonparametric statistical procedures (Breiman et al., 1984). Trees can capture non-linear relationships and interactions among predictors. The idea of tree-based regression is to partition covariate space into regions with homogeneous response variables. A recursive partitioning algorithm starts by using a splitting rule to divide the training data into two groups. This procedure is recursively applied to each group until the final groups contain only a few observations. These terminal nodes form a partition of the covariate space which is conveniently represented as a binary tree.

Ensembles

Trees earn A+ on interpretability, but only B on prediction (Breiman, 2001b). To raise the grade on prediction, a variety of techniques (e.g. bagging, boosting, random forests) grow an *ensemble* of trees, each fit to a perturbed version of the training set (Breiman, 1996; Breiman, L., 2001a; Bühlmann and Hothorn, 2007; Shapire, 2003).

These procedures are motivated by the observation that slight changes in the data can lead to different tree structures, but comparable error rates. Fitting trees to deliberately perturbed training data produces a set of plausible models, each achieved by the greedy algorithm converging to different local maxima. Rather than choosing one best model and discarding the rest, the resulting set of plausible models are combined to achieve superior accuracy.

The common method of perturbing data is to bootstrap, i.e. sample with replacement from the original data set. Bagging stands for Bootstrap Aggregation. A tree model is fit to each of several bootstrapped samples. In regression, predictions are obtained by averaging the predictions over the trees. Just as an average has lower variance than a single measurement, bagging reduces variance. This is especially effective when bagging unstable predictors like trees.

Random Forests

As with bagging, random forests use the bootstrap to perturb the data. In addition, they introduce another random element into the tree construction. At each node, a random set of predictor variables is chosen. The best split for the node is found by searching only over this set, and not over all predictor variables. This additional randomness allows variables to occur in the tree model that would not otherwise appear in greedy search approaches, and often helps achieve greater accuracy (Murua, 2002).

2.2 State Space Models

State space models provide a flexible yet relatively simple tool for analyzing dynamic phenomena and evolving systems, and extend classical statistical analysis to non-stationary processes. Informally, a state space model consists of:

- unobserved state variables whose dynamics are described by a Markov dependency
- observations, which are independent conditional on the state variables.

They allow interpreting a time series as the combination of several components, such as trend, seasonality, or regression. State estimation and forecasting are solved by recursively computing the conditional distribution of the quantities of interest, given the available information, hence can naturally be treated within a Bayesian framework.

Recursive Bayesian estimation

The goal of recursive Bayesian estimation is to estimate an unknown probability density function over time using observations and a mathematical process model. A Bayes filter uses information about noise and system dynamics to reduce uncertainty from noisy observations. The recursive algorithm consists of two steps at each time: predict and update, which involve state transition and observation equations. The predict step uses the state estimate from the previous time to produce an a priori state estimate at the current time, which is then updated by combining with current observation information to produce an a posteriori state estimate.

Dynamic Linear Model

The specific model used in this work is an example of a Dynamic Linear Model (West and Harrison, 1997; Pole et al., 1994), which is a state space model that is linear and Gaussian. Dynamic Linear Models (DLMs) offer a flexible framework to model both smooth and abrupt changes in time series, and accommodate subjective information. In its simplest form, a DLM is estimated using a Kalman filter, although Markov Chain Monte Carlo (MCMC) and particle filtering are used for more complex versions (Petris et al., 2009).

The observation equation of a DLM is a multivariate regression that relates the observation to trend, regressors, seasonality and other components. A system equation describes the evolution of the regression coefficients (state parameters) through time.

3 DATA DRIVEN APPROACH

Our approach includes preprocessing large amounts of recorded data, selecting suitable training data, applying a flexible regression method (random forest) to model EGT as a function of other variables, producing residuals that can be interpreted as engine wear plus error (Basu et al. 2008 a, b), and recursively tracking this wear using a state space approach. The following sections discuss each of these in turn.

3.1 Preprocessing

Data may be collected many times per second over flight. For the purposes of trending EGT over the life of an engine, we found it sufficient to choose a representative data point for each flight. Our approach is to obtain the most stable engine operational point possible, ideally at or near the maximum engine operating point.

During flight, stable operation occurs in two regimes: cruise and initial climb. Very stable engine data and aircraft parametric data may be recorded during aircraft cruise. However, the aircraft may not reach cruise during every flight (e.g. during training). Also, the engines may not reach maximum operation during cruise. During initial climb, the engine operates near or at its maximum for a fixed amount of time, but aircraft parametric and other engine data may change. A Boeing proprietary algorithm, not presented here, finds a window of data in which maximum EGT occurs. For this investigation, to compare directly with OEM data, only initial climb data will be discussed.

3.2 Select training set

In order to normalize to a common baseline condition, it is important to ensure that the trained model represents a consistent reference point in the life of the system. We select training data at a common temperature from the middle of engine life. Ideally, EGT is constant for long enough so that the training data will contain flights diverse enough to represent a set of environmental, flight, and engine parameter conditions encountered in real operation. Abnormal data are excluded for training the model.

3.3 Model relationship between EGT and nuisance variables

Assume that an observed EGT measurement at a time t is a function of other "nuisance" variables X_t (e.g., environmental, flight, and engine parameters):

$$EGT_t = f(X_t) + W_t + \varepsilon_t \qquad (1)$$

where ε_t is an error term consisting of measurement error, other variables that have not been measured, and possible model misspecification. W_t represents wear. This approach assumes that the EGT sensor is working properly, and cannot distinguish engine wear from EGT sensor (bias) error.

Previous work (Basu et al. 2008 a, b) estimated the function $\hat{f}(X_t)$ using statistical regression. To estimate the function, we experimented with linear, and various nonlinear parametric estimators. What worked the best is a random forest, which is a tree ensemble method (Breiman, 2001a), as discussed in Section 2.1.

The residuals r_t equal wear plus error:

$$r_t = EGT_t - \hat{f}(X_t) = \mathrm{W_t} + \varepsilon_t \qquad (2)$$

In previous work, we used the residuals as our estimate of wear. However, the real goal is to estimate the wear W_t. W_t is unobserved, thus suggesting a state space approach.

3.4 Applying Dynamic Linear Model

As seen in Figure 2, margin over the life of a typical engine follows a roughly piecewise linear pattern: a break-in period, followed by a relatively flat section, followed by decreasing slope, possibly accelerating at the end. This suggests using a locally linear model, and so we used a linear growth (or local linear trend, time varying slope) model, also known as a second order polynomial model (Petris et al., 2009).

The DLM is defined by:

Observation Equation: given by equation (2).

State Evolution
Local level:

$$W_t = W_{t-1} + d_{t-1} + \eta_t \qquad (3)$$

Local growth rate:

$$d_t = d_{t-1} + \nu_t \qquad (4)$$

The error terms η_t and ν_t are assumed to be Normally distributed with mean 0 and covariance Σ. The wear at time *t* equals the wear at the previous time W_{t-1} plus a local growth rate d_{t-1}. The local growth rate is a random walk. The goal is to estimate the state sequence W_t .

3.5 Software

This work used several packages from the open source statistical software R (R Development Core Team, 2009): *randomForest* (Liaw, A. and Wiener, M. (2002)) and *dlm* packages (Giovanni Petris, 2010). Petris et al. (2009) describes the dlm package.

3.6 Implementation

The methodology in a previous effort (Basu *et al*., 2008 a, b) only requires the current data when predicting and generating residuals using a fitted random forest model. The random forest predicts each observation independently of the past.

Here the dynamic linear model (DLM) relates predictions over time. At each iteration, information about the past is encoded and saved as the parameters of the DLM from the previous iteration. These parameters need to be stored for each aircraft/engine. If DLM estimates from the previous iteration do not already exist, then the program gets default starting values, which represent an aggregate model estimated from historical engine data.

To estimate default starting values for the observation variance and state estimates, we fit a DLM model to the random forest residuals for each engine in the training data. The default starting value for the observation variance is the median. For the state estimates, we use the first 15 random forest residuals of each engine data series to estimate a linear regression. The intercept and slope represent initial estimates for wear and growth for that engine. The default starting value for the local level and growth rate are the medians.

In this way, one set of starting values is used for all engines. As shown below, this works well in practice. However, further research could investigate somehow clustering engines, and using separate sets of starting values. Another possibility is to use a small set of data at the beginning of engine life to individually estimate a DLM for each engine. This latter approach would mean that at the beginning of engine life, the program returns just the prediction from the random forest, without yet relating them in time.

We set the evolution variances (i.e., the diagonal elements of Σ) to large values. This allows the data to “speak for themselves”, as is suitable when starting out. The evolution variance is estimated in subsequent iterations using a discount factor = .9 (Pole et al., 1994). See Section 4.8.

4 EXPERIMENTS

This section discusses an experiment on data collected during initial climb. The random forest (RF) model relates Maximum EGT (achieved during climb) to engine, flight, and environmental parameters. The residuals estimate wear (but also include noise as discussed above). The Dynamic Linear Model (DLM) estimates wear, and further reduces variability by relating predictions (residuals) over time.

The goal is to compare the performance of these data driven techniques – RF alone vs. RF + DLM -- and compare each to the OEM method. Section 4.3 gives specific evaluation criteria.

4.1 Data

This investigation used customer Quick Access Recorder (QAR) data, provided from about 30 aircraft from the time the engine was installed to the time of this investigation. There are several years of data for each aircraft. Although there are four engines on each aircraft, data does not always exist for all four engines. In total, there is data for about 100 engines. A challenge is to develop a procedure that works for all engines in a fleet, for aircraft that fly in a variety of geographic locations and conditions.

For each engine, the data consist of about thirty parameters collected during initial climb over many flights. The types of variables include aircraft environmental/operational data such as Mach number and engine parametric data such as engine spool speeds.

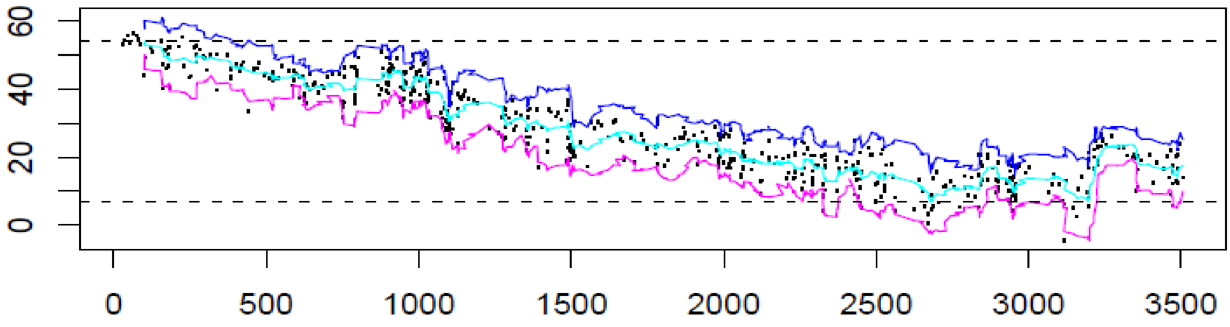
OEM EGT normalization is used as a baseline for comparison. This is expressed as margin, a threshold temperature minus the normalized EGT, and therefore decreases over engine life.

4.2 Selecting Training and Test Sets

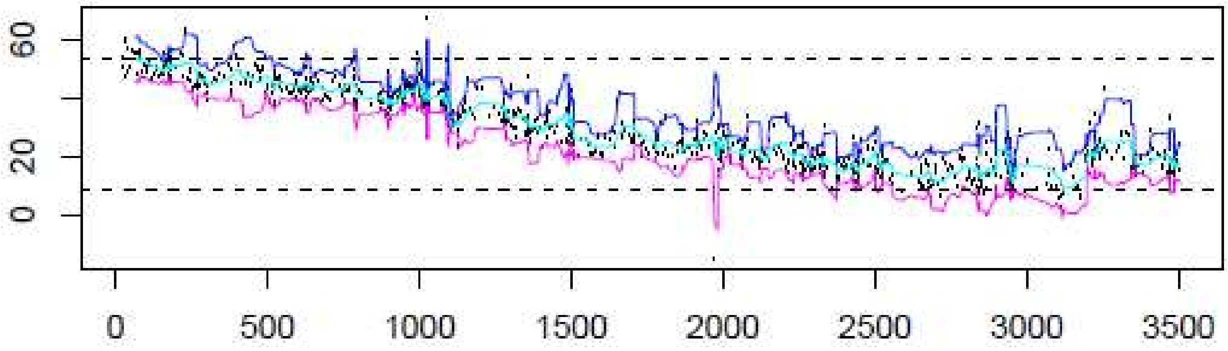
Since the goal is to normalize to a fixed wear point, we collected QAR data from about 30 engines during time periods when the OEM margin values are a common frequently occurring temperature. This selection process yielded a set of 3305 observations to train the random forest model. We then compared OEM and data driven procedures on 73 engines (the test set).

4.3 Evaluation

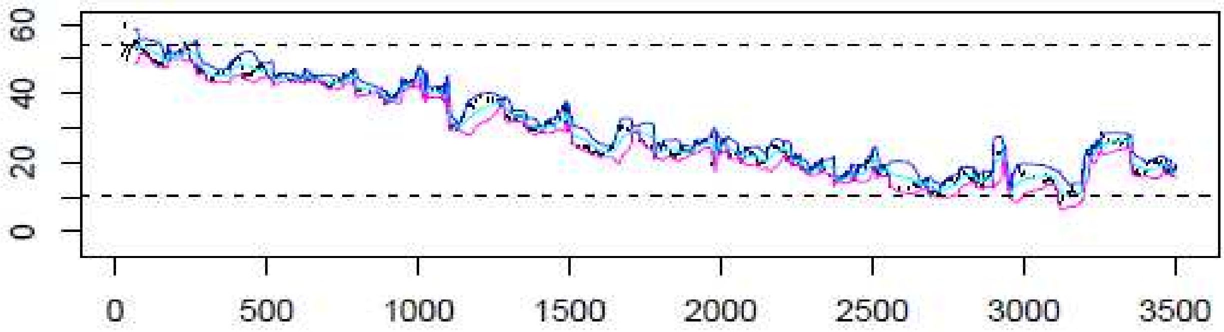
The goal is to reproduce the range of the OEM results, but with less scatter. Range and scatter are defined



OEM: max − min of moving average = 46.91
median spread = 15.93



RF Only: max − min of moving average = 45.46
Median spread = 14.07



RF + DLM: max − min of moving average = 43.5
median spread = 4.03

Figure 3: Bollinger plots comparing the OEM, RF, and RF+ DLM solutions for a randomly chosen engine. The ranges are comparable. The RF solution reduces scatter over the OEM solution, but the RF + DLM dramatically reduces scatter even further.

below. We measure these objectives using *Bollinger bands* as follows:

- Choose a fixed window width
- Calculate average, standard deviation (sd) for each window
- Construct Bollinger bands as average + k * sd where k is a constant. In our results, we choose k = 2, so that the bands roughly correspond to .95 confidence bands under a Gaussian distribution assumption.

Then calculate:

- *range* as the maximum moving average minus the minimum moving average. This is one value.
- *scatter* as the upper minus the lower Bollinger bands. There is one value for each observation.

To more easily compare our results with the OEM margin results, we transform our model outputs to start at the same maximum moving average as the OEM margin.

4.4 Results

The goal is to compare the performance of the data driven techniques – RF alone versus RF + DLM -- to the OEM method. See Figure 3 for Bollinger plots comparing the OEM, RF, and RF+DLM solutions for a randomly chosen engine.

Figures below summarize the results by showing histograms of range and median scatter for the engines in the test set. Figure 4 compares the difference in range from OEM, and shows that most ranges are within +/- five degrees of OEM. Figure 5 shows the difference in median scatter from OEM. These figures show that:

- The RF predictions have comparable range, but reduced median scatter. However, individual time series plots and histograms (not shown here) indicate that there can be extended periods with both higher scatter and lower scatter than the OEM solution. Therefore, this tempers the conclusion of reduced median scatter.

- Relating RF predictions over time using a DLM dramatically reduces median scatter over the OEM solution. Moreover, improved performance is evident in individual plots (not shown here).

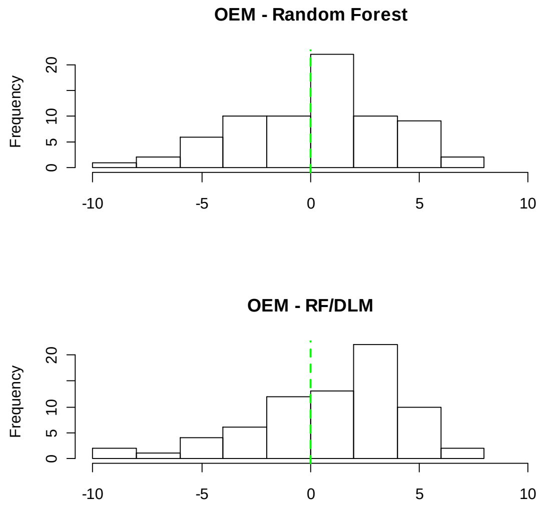


Figure 4: Difference in Range from OEM approach. The ranges are comparable for all approaches. Most are within +/- 5 degrees. A green dotted vertical line is drawn at 0 difference.

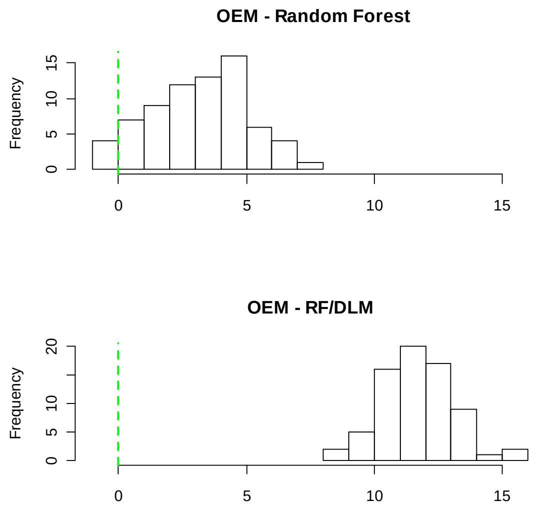


Figure 5: Difference in median scatter from OEM approach. The top plot shows that median scatter for the OEM is greater than for the data driven approach using RF-alone. However, relating RF predictions over time using a DLM dramatically reduces median scatter over the OEM solution. Note that the median scatter for the OEM approach always exceeds that for RF + DLM by at least 8. A green dotted vertical line is drawn at 0 difference.

4.5 MONITOR THAT DETECTS OUTLIERS AND SUDDEN SHIFTS

The RF + DLM model above has shown improvement over both the OEM solution and our previous solution using just RF residuals. In order to deploy this model in real life solutions, a working implementation should also reject outliers and adapt model parameters after structural changes such as a sudden EGT shift.

The approach used here is based on the work by West (1986) and West and Harrison (1986). A more complex approach would use simulation-based Bayesian inference. However, as a first step, it seemed reasonable to apply this approach which is useful and practical in many situations, with the advantage that discount factors (described below) allow closed form calculations.

4.6 Bayesian Model Monitoring

Sequential Bayesian modeling analyzes observations in real time, updating inferences and predictive statements using newly obtained information and observations. It assesses model fit using predictive distributions. The idea behind monitoring is to compare the predictive fit of the standard model with an alternative model that specifies the nature of "unusual". The central problem is to construct suitable alternatives to the "standard" model used for analysis. In this work, the alternative model is similar in form to the standard, but allows for changes in the values of the parameters.

Monitoring is based on the Bayes Factor, the ratio of likelihoods, which compares the predictive ability of the standard model versus the alternative model. It detects discrepancies between the data and standard model predictions. Examples of model failures include outliers and structural changes in the time series. More formally, the Bayes factor at time t is defined as:

$$H_t = p(y_t \mid D_{t-1}) / p_A(y_t \mid D_{t-1}) \quad (5)$$

where D_{t-1} is the data until time $t-1$, and y_t is the current observation. Small values of H_t indicate poor performance of the standard model relative to the alternative. It is possible to put a threshold on (observation – forecast inconsistency): a Bayes factor > 10 gives evidence for the standard model; >100 gives strong evidence. A Bayes factor < 1/10 gives evidence for the alternative model; while < 1/100 gives strong evidence.

The *overall* Bayes factor at time t is the product of the Bayes factors until that time, and gives a measure of the global fit. However, the problem with such a global measure is that the greater weight of historical performance may mask local changes. E.g., good historical performance of the model may swamp a small Bayes factor due to an outlier, which then goes undetected.

In a practical monitoring application, the local changes are of greatest interest. A *cumulative* Bayes factor is a product of the most recent *k* Bayes factors, and is sensitive to local model failure, and can indicate slow changes that may not be evident in a single Bayes factor.

$$W_t(k) = H_t H_{t-1} \cdots H_{t-k+1} = H_t W_{t-1}(k-1) \quad (6)$$

To focus on the most likely point of possible change, calculate the most discrepant group of recent, consecutive observations as:

$$V_t = \min_{1 \le k \le t} W_t(k) \quad (7)$$

It turns out that when the cumulative evidence at time t favors the standard model, so that $V_{t-1} \ge 1$, then

$$V_t = H_t \quad (8)$$

and decisions about possible inadequacies are based on the current observation y_t alone. If the Bayes factor H_t is small enough, then y_t may be an outlier or the beginning of a structural change.

On the other hand, if the evidence before time t is against the standard model, so that $V_{t-1} < 1$, then the cumulative Bayes factor is multiplied by H_t:

$$V_t = H_t V_{t-1} \quad (9)$$

In this way, the monitor detects either gradual or abrupt changes.

4.7 Automatically Adapting to Model Change

When changes in parameter values are the primary cause of standard model failure, an additional goal is to automatically adapt to the onset of change. Incorporating increased uncertainties into the model leads naturally to more rapid adaption by allowing future data to more heavily influence the updating of posterior distributions. In this way, models self correct after structural changes. However, the automatic procedure must also distinguish an outlier from a structural change.

The following scheme (West and Harrison, 1986; West, 1986) isolates and rejects outliers, and in cases of structural change, automatically increases uncertainty about the parameters to rapidly adapt to new data.

A. If $H_t \geq \tau$, then y_t is consistent with standard model. But assess possibility of change before this time by proceeding to (B)

B. If the cumulative Bayes factor $V_t \geq \tau$, proceed to (D) (update as usual). Else change is indicated. Proceed to (C).

C. Reject y_t as providing no useful information at time t about the standard model parameters. Do not update model using y_t (equivalently, treat y_t as missing). Moreover, allow for change by increasing the uncertainty about the parameter vector, leading to more rapid adaptation to new data

D. Standard update: standard model is satisfactory. Update as usual to the posterior and thence to the prior for time t + 1.

4.8 Discount Factors

Specifying the state evolution covariance matrix is a crucial step. The values control the stochastic variation in the evolution of the model, and determine the role of past observations. A key problem is that one covariance is typically not suitable for all times. Moreover, it is difficult to specify the covariance elements.

In the system equation, the covariance leads to an increase in uncertainty, or equivalently a loss of information, about the state vector between successive times. This idea is natural, and leads to specifying the posterior covariance as a fraction $1/\delta$ of the prior covariance, and therefore the state evolution covariance as $(1-\delta)/\delta$ of the prior covariance. The degree of adaptation to new data increases as the discount factor becomes smaller.

The RF + DLM model uses a discount factor of .9. In the monitor, there are two other uses of the discount factor. First, the alternative model uses a discount factor of .05. Second, the monitor enables the model to adapt after a structural change by increasing the discount factor to .1.

4.9 Preliminary Evaluation

A preliminary evaluation first looked at the monitor alerts and the detected outliers for the same data set shown in Figure 3. Figure 6 shows how the algorithm alerts to changes for a threshold of .135 on the Bayes Factor. This corresponds to a weight of evidence against the standard model of roughly 2. These plots give some insight into how the monitor works, and shows that it generally alerts when the slope changes direction or there is a shift.

Next, we ran a simple experiment whose goal is to begin to understand what magnitude shift the monitor can detect, and how long the detection takes. We introduced a shift in the last 300 points by incrementing the data by 10, 15 and 20 degrees.

The threshold on the Bayes factors affects the size of the shift that the algorithm can detect. For a threshold = .2, the monitor detects a shift of 20 degrees, but not a 15 degree shift. It takes 15 observations to detect the 20 degree shift. But for a threshold of .3, the monitor detects a shift of 15 degrees after 25 observations. This would appear to be good performance because as Figure 7 of the raw Max EGT shows, the 15 degree shift is not very noticeable. Deciding on the appropriate threshold depends on what magnitude shifts the user wants to detect, and also the sensitivity to other changes. Smaller threshold values decrease sensitivity to slower, less marked changes.

This has been a preliminary investigation of how the monitor works. More experimentation is needed to more fully explore the impact of the discount factors for the state evolution matrix, the starting values, and the thresholds on the Bayes factors. Future experiments should consider more data sets, gradual ramping of increases and other realistic changes, as well as real shift occurrences in the data.

5 CONCLUSION

This paper deals with a general problem of relating a sequence of predictions over time using state space modeling. We focus specifically on predicting engine wear for aircraft.

In the prediction task, the observed exhaust gas temperature (EGT) depends on both wear and other factors. After suitable preprocessing the data, we fit a flexible empirical model (random forest) to relate EGT to other factors at a baseline wear point. When we predict new data using the model, the residuals represent wear plus error. Using a state space modeling

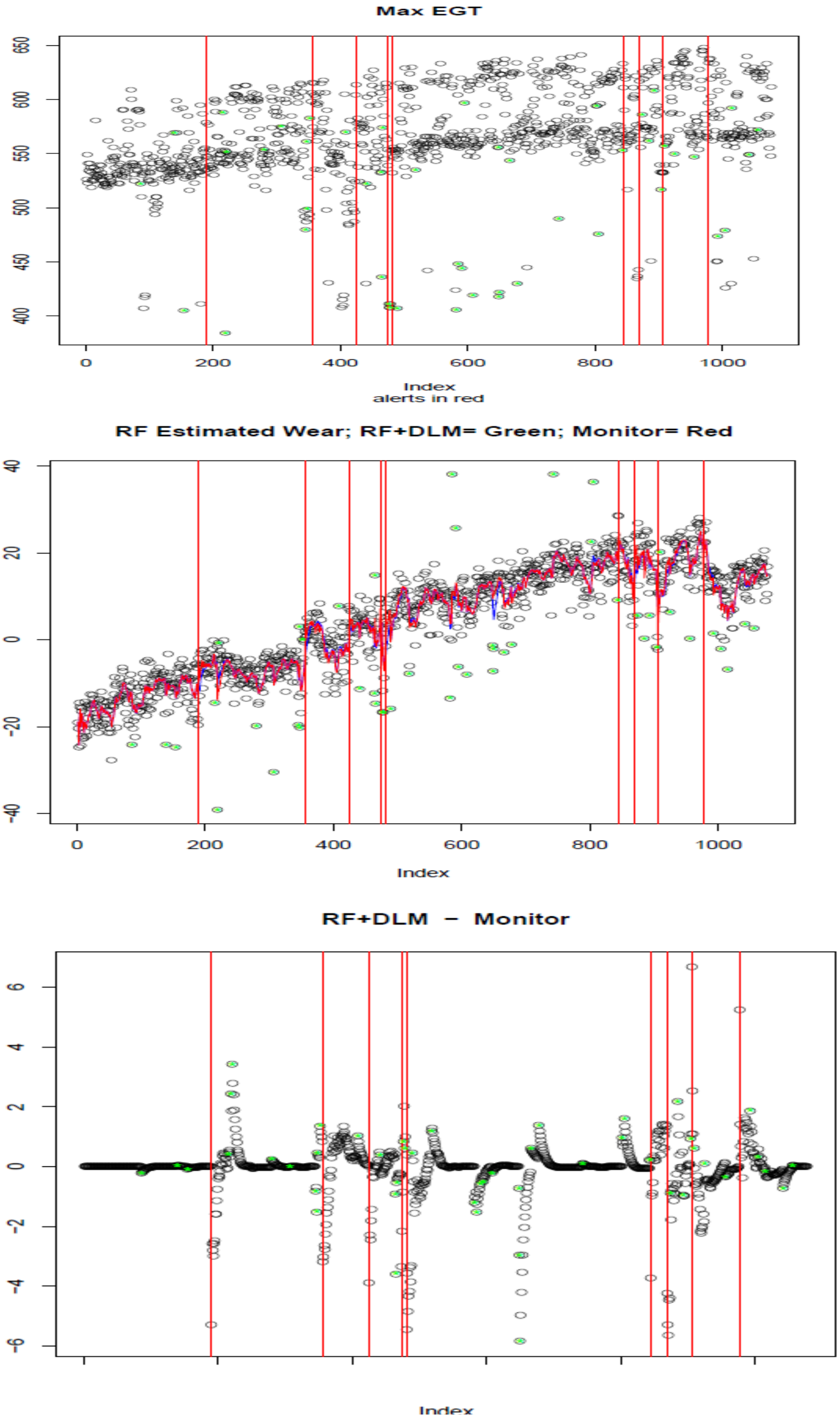


Figure 6: The top plot shows Max EGT. In the middle plot, which shows the random forest residuals, the blue line indicates the estimates of wear using RF+ DLM and red line indicates the monitor. Differences between the wear estimates are subtle in this plot, but are more apparent in the bottom plot. These differences are due to the monitor ignoring potential outliers and allowing the model to self correct after an alert. The detected outliers are indicated by green *s; and the alerts are indicated by vertical lines. The alerts are based on a threshold of .135 on the Bayes Factor thresholds. This corresponds to a weight of evidence against the standard model of roughly 2.

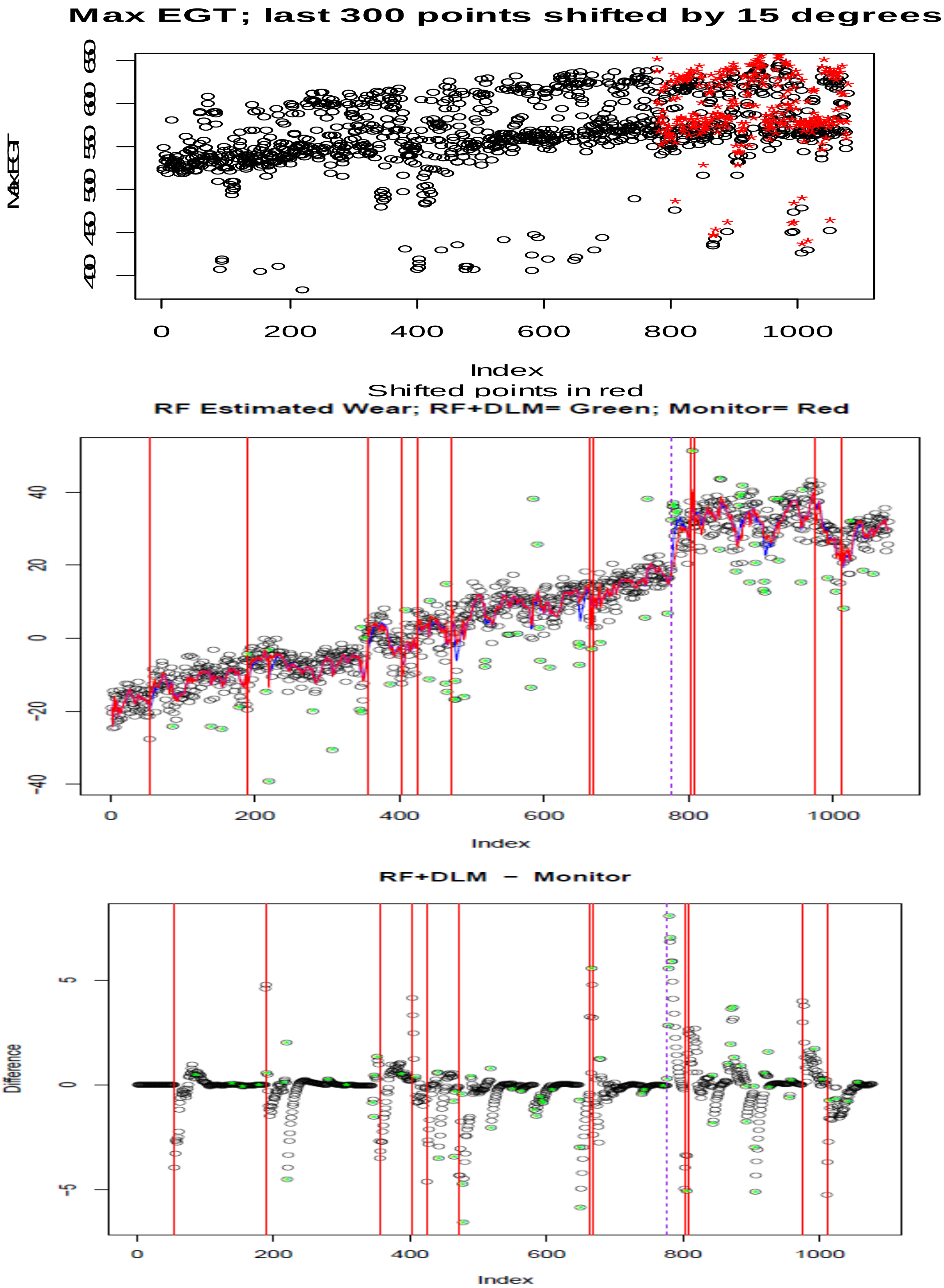


Figure 7: The top plot introduces a shift of 15 degrees at the end of the Max EGT series. The shifts are shown as red *'s. The shift is barely discernible in the Max EGT plot. The middle plot shows the monitor output, and an immediate sharp increase in estimated wear. The dotted purple line indicates the beginning point where the data is shifted. The monitor alerts 25 time points after the shift. It uses a threshold of .3 on the Bayes Factor. Note that the monitor alerts more often than when the threshold = .135, including a change in slope at the end that it missed in the previous plot. The bottom plot shows that the monitor differs the most from the RF+DLM just after the shift.

framework, we estimate the unobserved state (wear) using a dynamic linear model fit using a Kalman filter.

Experiments show that predicting engine wear using a dynamic linear model to relate regressions over time dramatically reduces the variability of predictions from both the OEM approach and our previous approach, which independently generated predictions at each time using random forest regression (Basu *et* al., 2008 a, b).

This RF + DLM model forms the basis of a Bayesian approach to monitor for sudden changes and reject outliers, and adapt the model after these events. More investigation on the monitor is needed, but preliminary work shows that the monitor can detect shifts in temperature that are barely discernible in the originally observed EGT.

Future work on this application includes:

- Investigate the Bayesian monitor in greater depth, including detection and false alarm rates, and the effect of starting values, discount factors, and thresholds on the Bayes factors.
- Incorporate a seasonal component.
- Go beyond linear Gaussian models fit with a Kalman filter, to more general nonlinear models fit using particle filters (Doucet *et* al., 2001).
- Exploit the fact that on each aircraft, there are four engines, which operate in the same operational and environmental conditions. This fact might be used, for example, to distinguish between shifts in rates of wear -- due to common changing conditions -- from sudden EGT shifts on any one of the engines due to a fault. Accomplishing this may involve using a hierarchical model: engines within aircraft.

In this particular application, because the goal is to normalize EGT data to one wear point, the problem is simplified because we only fit the regression relationship once. However, more general applications require updating the regression or classification models over time. Challenges include developing real time algorithms. In order to continue using tree ensemble regression, a possible solution is to use ensembles as incremental learners (Parikh D. and Polikar R., 2007).

ACKNOWLEDGMENT

This work has been funded primarily under an internal research and development (IR&D) project funded by the NET domain, and then applied to data under another IR&D project funded by the Integrated Vehicle Health Management (IVHM) group within the Support and Services domain. Thanks to Boeing colleagues Dragos Margineantu and Tomas Singliar for helpful discussions, and Nancy Gove for selecting training sets and running experiments comparing the original procedure to OEM.

REFERENCES

Basu, S. and Farner, S. and Schimert, J. and Wineland, A. (2008a)."Statistical Methods for Modeling and Predicting Maximum Engine Exhaust Gas Temperature (EGT): First Analysis Using Climb Data from a Single Aircraft". Networked Systems Technology Technical Report (NST-08-001)

Basu, S. and Farner, S. and Schimert, J. and Wineland, A. (2008b). "Regression Based Method for Predicting Engine Wear from Exhaust Gas Temperature", Prognostics and Health Management Conference, Denver, CO.

Breiman, L. and Friedman, J. and Olshen, R. and Stone, C. (1984), Classification and Regression Trees, Wadsworth, Monterey, CA.

Breiman, L. (1996), "Bagging predictors", Machine Learning, volume 26, Pages 123—140.

Breiman, L. (2001a), "Random Forests", Machine Learning, volume 45, number 1, pages 5-32.

Breiman, L. (2001b), "Statistical modeling: The two cultures (with discussion)", Statistical Science, volume 16, number 3, pages 199—231.

Bühlmann, P. and Hothorn, T. (2007). Boosting algorithms: regularization, prediction and model fitting. Statistical Science, 22(4):477-505.

Doucet, A. (Editor) and de Freitas, N. (Editor) and Gordon, N. (Editor), (2001). "Sequential Monte Carlo Methods in Practice, Springer.

Hall, D.L. and McMullen, S.A.H. (2004). “Mathematical Techniques in Multisensor Data Fusion”, Artech House Information Warfare Library.

Friedman, J.H (2001), "Greedy function approximation: A gradient boosting machine", Annals of Statistics, volume 29, pages 1189—1202.

Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. R News 2(3), 18--22.

Liggins, M. E., Hall, D. L., Llinas, J. (Editors) (2009) "Handbook of Multisensor Data Fusion: Theory and Practice," 2nd Edition, CRC Press, New York.

Murua, A (2002), "Upper bounds for error rates associated to linear combination of classifiers", IEEE Transactions on Pattern Analysis and Machine Intelligence.

Parikh D. and Polikar R. (2007), “An Ensemble based incremental learning approach to data fusion”, IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics, vol. 37, No. 2.

Giovanni Petris (2010). dlm: Bayesian and Likelihood Analysis of Dynamic Linear Models. R package version 1.1-1.

Petris, G. and Petrone, S. and Campagnoli, P. (2009) "Dynamic Linear Models with R", Springer.

Pole, A. and West, M. and Harrison, J. (1994). Applied Bayesian Forecasting and Time Series Analysis, Chapman & Hall.

R Development Core Team (2009). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

Schapire, R.E. (2003). The boosting approach to machine learning: An overview. In D. D. Denison, M. H. Hansen, C. Holmes, B. Mallick, B. Yu, editors, Nonlinear Estimation and Classification. Springer.

Schwabacher, M. and Kai Goebel, K. A Survey of Artificial Intelligence for Prognostics. NASA Ames Research Center. MS 269-3. http://ic.arc.nasa.gov/people/schwabac/FS02SchwabacherM.pdf.

Sinhaa, A. and Chenb, H. and Danua, D. G. and Kirubarajana, T. and Farooqc, M. (2008), "Estimation and Decision Fusion: A Survey," Neurocomputing Volume 71, Issues 13-15, pages 2650-2656,

West, M. (1986). Bayesian Model Monitoring", Journal of the Royal Statistical Society, Series B, volume = 48, pages= 70—78.

West, M. and Harrison, J. (1986). "Monitoring and Adaptation in Bayesian Forecasting Models", Journal of the American Statistical Association, Vol. 81, No. 395, pages 741-750.

West, M. and Harrison, J. (1999). "Bayesian Forecasting and Dynamic Models", Springer.

James Schimert received his Ph.D. in statistics from University of Washington, Seattle, Washington in 1992. He is an Advanced Computing Technologist in the Network Systems Technology group, Boeing Research and Technology, The Boeing Company, Bellevue, Washington. His research interests include topics in data mining and machine learning including tree ensembles and anomaly detection, in applications including vehicle Health Management. Previously, he worked in the Research group at Insightful Corp., where he served as consultant and as principal investigator on NIH and NSF SBIR funded research efforts to develop commercial software for advanced statistical computing.

Art Wineland received his Bachelor of Science Degree in Mechanical engineering from The University of Toledo, Toledo Ohio in 1985. He is a Principle Engineer in the Propulsion Systems Engineering group in Long Beach, California. His primary responsibilities are in engine health and diagnosis.