Degradation Modeling and Remaining Useful Life Prediction of Aircraft Engines Using Ensemble Learning

Zhixiong Li
Department of Mechanical and Aerospace Engineering,
University of Central Florida,
Orlando, FL 32816
e-mail: zhixiong.li@Knights.ucf.edu

Kai Goebel
NASA Ames Research Center,
Moffett Field, CA 95134;
Division of Operation and Maintenance Engineering,
Luleå University of Technology,
Luleå 971 87, Sweden
e-mail: kai.goebel@nasa.gov

Dazhong Wu[1]
Department of Mechanical and Aerospace Engineering,
University of Central Florida,
Orlando, FL 32816
e-mail: dazhong.wu@ucf.edu

Degradation modeling and prediction of remaining useful life (RUL) are crucial to prognostics and health management of aircraft engines. While model-based methods have been introduced to predict the RUL of aircraft engines, little research has been reported on estimating the RUL of aircraft engines using novel data-driven predictive modeling methods. The objective of this study is to introduce an ensemble learning-based prognostic approach to modeling an exponential degradation process due to wear as well as predicting the RUL of aircraft engines. The ensemble learning algorithm combines multiple base learners, including random forests (RFs), classification and regression tree (CART), recurrent neural networks (RNN), autoregressive (AR) model, adaptive network-based fuzzy inference system (ANFIS), relevance vector machine (RVM), and elastic net (EN), to achieve better predictive performance. The particle swarm optimization (PSO) and sequential quadratic optimization (SQP) methods are used to determine optimum weights that are assigned to the base learners. The predictive model trained by the ensemble learning algorithm is demonstrated on the data generated by the commercial modular aero-propulsion system simulation (C-MAPSS) tool. Experimental results have shown that the ensemble learning algorithm predicts the RUL of the aircraft engines with considerable robustness as well as outperforms other prognostic methods reported in the literature.
[DOI: 10.1115/1.4041674]

Keywords: remaining useful life prediction, prognostics and health management (PHM), degradation modeling, aircraft engines, ensemble learning

1 Introduction

Aircraft or turbine engine failures may result in significant economic losses and even accidents in extreme cases. While the reliability of turbine engines in use on modern turbine-powered aircrafts has been improved over the past few decades, abnormal engine degradation can occur at any time because of a variety of mechanical problems. According to a report by the International Development Consulting, global maintenance, repair, and overhaul (MRO) spend on commercial aircraft engines in 2016 was valued at $27 billion [1]. The global maintenance, repair, and overhaul market size is expected to grow by 4.1% annually, reaching over $40 billion in 2026. For example, due to three engine failures in 2016, the Japanese airline ANA will refurbish 100 Rolls-Royce Trent 700 engines on its 50 Boeing 787 Dreamliners. These engine failures were caused by corrosion and cracking of turbine blades.

Predictive maintenance enables airlines to avoid costly equipment downtime and reduce maintenance costs by performing just-in-time maintenance actions [2]. Predictive maintenance determines the condition of in-service equipment in order to predict equipment failure or remaining useful life (RUL). The RUL of an aircraft engine is defined as the amount of time in hours or cycles from the current time to the end-of-life in which an aircraft engine is expected to serve its intended function. Predictive maintenance requires health monitoring systems and predictive modeling technologies. The existing literature pertaining to RUL prediction for aircraft engines can be classified into two categories: model-based and data-driven prognostics [3–5]. Model-based prognostic methods describe system behavior and system degradation using physics-based models typically in combination with state estimators such as the Kalman filter, the particle filter, and the hidden Markov model [6]. While model-based prognostic methods provide closed-form solutions, certain assumptions must be made. To address this issue, data-driven prognostic methods represent the system degradation process using machine learning algorithms. Current data-driven methods are developed based on classical machine learning algorithms such as neural networks, support vector machines (SVM), and decision trees. One of the primary limitations associated with classical machine learning algorithms is that they are not able to predict the RUL of aircraft engines with sufficient accuracy. In addition, it is difficult to determine which or what type of learning algorithm should be selected among many competing learning algorithms. Therefore, a novel ensemble learning-based prognostic approach is introduced to model the degradation process of aircraft engines due to wear as well as to predict the RUL. The unique advantage of ensemble learning is that it can select machine learning algorithms with better performance. Using an ensemble of multiple learning algorithms instead of a single algorithm, one can reduce the risk of choosing a learning algorithm with poor performance. The ensemble learning-based prognostic approach is demonstrated on one of the datasets (i.e., FD004) collected from the commercial modular aero-propulsion system simulation (C-MAPSS) tool.

The remainder of this paper is organized as follows: Sec. 2 reviews the related work on degradation modeling and prognostics of aircraft engines. Section 3 presents the ensemble learning-based prognostic approach. Section 4 presents a case study. Section 5 provides conclusions and future work.

2 Related Work

This section reviews the related work on degradation modeling and prognostics of aircraft engines. The datasets generated by

[1]Corresponding author.

Manuscript received May 13, 2018; final manuscript received October 2, 2018; published online November 16, 2018. Assoc. Editor: Liang Tang.

NASA's C-MAPSS tool have been widely used to evaluate the performance of prognostic algorithms [4]. For example, Mosallam et al. [7] introduced a method based on an unsupervised variable selection method and *k*-nearest neighbors (KNN) classifier. Experimental results have shown that a mean absolute percentage error of 12.19% can be achieved. Liu and Huang [8] developed a data-level fusion method to construct a health index (HI) that can characterize the condition of aircraft engines. This data fusion method allows for minimizing model fitting errors and the variance in the failure threshold simultaneously. Experimental results have shown that the HI outperforms that of existing data-level fusion methods. Nieto et al. [9] introduced a hybrid particle swarm optimization (PSO)-SVM-based model to predict the RUL of aircraft engines. This method integrates support vector regression (SVR) with the PSO techniques. The experimental results have shown that a coefficient of determination of 0.9 can be achieved. Khelif et al. [10] introduced an SVR-based predictive modeling method that predicts the RUL of aircraft engines. A variable selection method was developed to perform feature selection. Experimental results have shown that this method outperforms some of the current methods in terms of S-score. Yu [11] developed a method that combines logistic regression and particle filtering techniques for engine health assessment and prediction. A HI-based on logistic probability was introduced to characterize the health conditions of aircraft engines. A data-level fusion method was developed to correlate the HI with sensor signals. Hu et al. [12] proposed a data-driven prognostic approach that combines multiple member algorithms with a weighted-sum formulation. Three weighting methods, including the accuracy-based weighting, diversity-based weighting, and optimization-based weighting, were used to determine the weights of member algorithms. Experimental results have shown that the method can predict the RUL with sufficient accuracy using cross valuation. Ramasso and Gouriveau [13] introduced a data-driven method that predicts the RUL of aircraft engines using a neuro-fuzzy system and the Dempster–Shafer theory. Eight features were extracted using a feature selection method based on the Kullback–Leibler divergence. Experimental results have shown that the prediction accuracy ranges between 74.4% and 92.2%. Li et al. [3] developed an ensemble learning-based method that takes into account the effects of time-dependent degradation. Experimental results have shown that the method can predict the RUL of aircraft engines with an S-score of 5.75. While model-based and data-driven prognostic approaches have been introduced to predict the RUL of aircraft engines, little research has been reported on degradation modeling and RUL prediction of aircraft engines using novel learning techniques. To fill the research gap, this paper presents an ensemble learning-based prognostic approach to degradation modeling and RUL prediction of aircraft engines.

3 Ensemble Learning-Based Predictive Modeling

Ensemble learning methods are meta-algorithms that combine multiple base learners into a single predictive model in order to improve prediction performance [3]. Ensemble learning methods are classified into two categories: parallel and sequential ensemble methods. The parallel ensemble methods such as random forests (RFs) build multiple base learners independently and then average their predictions or take a weighted sum of their predictions. Parallel ensemble learning methods can be implemented, for example, using a Bayesian [14] or a Dempster–Shafer framework [15] where the weights are interpreted as probabilities. In contrast, sequential ensemble methods such as AdaBoost construct base learners sequentially and then reduce the bias of the combined base learners. In this study, the parallel ensemble method will be used to develop the ensemble learning algorithm.

Figure 1 illustrates a computational framework of the ensemble learning-based prognostic approach. This framework consists of variable selection, model training, and model validation and test phases. The training dataset $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^{\mathrm{T}}$ contains the observations of N different run-to-failure units Each observation $\mathbf{x}_i = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_P]$ $(i = 1, \ldots, N)$ consists of P variables acquired from P variables. Because not all of the P variables are significant for RUL prediction, a machine learning algorithm (i.e., RFs) is used to select the most important variables. In the model training phrase, cross-validation (CV) is conducted to train predictive models (i.e., base learners) using the base learning algorithms. PSO and sequential quadratic optimization (SQP) methods are used to optimize the weigh vector $\mathbf{w}$ for the base learners. In the model validation phase, the weighted sum of the predictions of individual base learners is used with a weight vector $\mathbf{w}$ to obtain the predicted RULs of the test units. The pseudocode of the ensemble learning algorithm can be found in Table 1. In this study, the base learning algorithms are combined using the following parallel ensemble method:

$$\hat{L}_i^{\mathrm{T}} = \sum_{j=1}^{J} w_j \hat{L}_i^j \tag{1}$$

where $\hat{L}_i^{\mathrm{T}}$ denotes the ensemble prediction of ith training dataset, w_j $(j = 1, 2, \ldots, J$ with J being the number of base learners) denotes the optimum weight assigned to jth base learner, and $\hat{L}_i^j$ denotes the predicted RUL of ith training dataset using jth base learner. Equation (1) can be written in matrix form as $\hat{\mathbf{L}}^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}}\hat{\mathbf{L}}$, where $\hat{\mathbf{L}}^{\mathrm{T}} = [\hat{L}_1^{\mathrm{T}}, \ldots, \hat{L}_N^{\mathrm{T}}]$ (N is the total number of training datasets), $\mathbf{w} = [w_1, w_2, \ldots, w_J]^{\mathrm{T}}$, $\hat{\mathbf{L}} = [\hat{\mathbf{L}}^1, \hat{\mathbf{L}}^2, \ldots, \hat{\mathbf{L}}^7]^{\mathrm{T}}$ and $\hat{\mathbf{L}}^j = [\hat{L}_1^j, \ldots, \hat{L}_N^j]$ $(j = 1, 2, \ldots, J)$.

The PSO and SQP methods are used to determine optimum weights algorithm is used to obtain the optimum weight vector $\mathbf{w}$ [16]. The objective of the optimization methods is to minimize the prediction error ε_{CV} of the weighted sum of the predicted RULs.

$$\begin{cases} \underset{\mathbf{w}}{\operatorname{argmin}}\, \varepsilon_{CV} = \dfrac{1}{N}\sum \Gamma\left(\mathbf{w}^{\mathrm{T}}\hat{\mathbf{L}}, \hat{\mathbf{L}}^{\mathrm{T}}\right) \\ \text{subject to} \quad \displaystyle\sum_{j=1}^{J} w_j = 1 \end{cases} \tag{2}$$

where $\mathbf{w}$ $(= [w_1, w_2, \ldots, w_J]^{\mathrm{T}})$ is the weight vector and $\Gamma(\cdot)$ is a predefined error criterion that measures the discrepancy between the predicted RUL $(\hat{\mathbf{L}}^T)$ and the true RUL $(\mathbf{L}^{\mathrm{T}})$. A greater weight will be assigned to the base learner with better performance.

3.1 Variable Selection. The RFs algorithm is used to select the most important variables or features from the original twenty-one (21) variables based on a measure called variable importance [17]. Variable importance is measured by averaging the sum of the weighted reduction in residual sum of squares for all of the nodes where a variable is used over all of the decision trees of a random forest. The reason why dimensionality reduction is important is that selecting a subset of relevant variables or features for training predictive models can increase prediction accuracy and computational efficiency as well as avoid overfitting.

3.2 Base Learning Algorithms. To improve the prediction accuracy of the ensemble learning algorithm, the base learning algorithms should be as diverse as possible. As shown in Table 2, the ensemble learning algorithm combines seven machine learning algorithms of different type, including RFs, classification and regression tree (CART), recurrent neural networks (RNN), autoregressive (AR) model, adaptive network-based fuzzy inference system (ANFIS), relevance vector machine (RVM), and elastic net (EN). The theories behind these algorithms can be referred to Refs. [17–25], respectively.

3.3 Model Training and Validation. The predictive model is trained on the training dataset using the ensemble learning

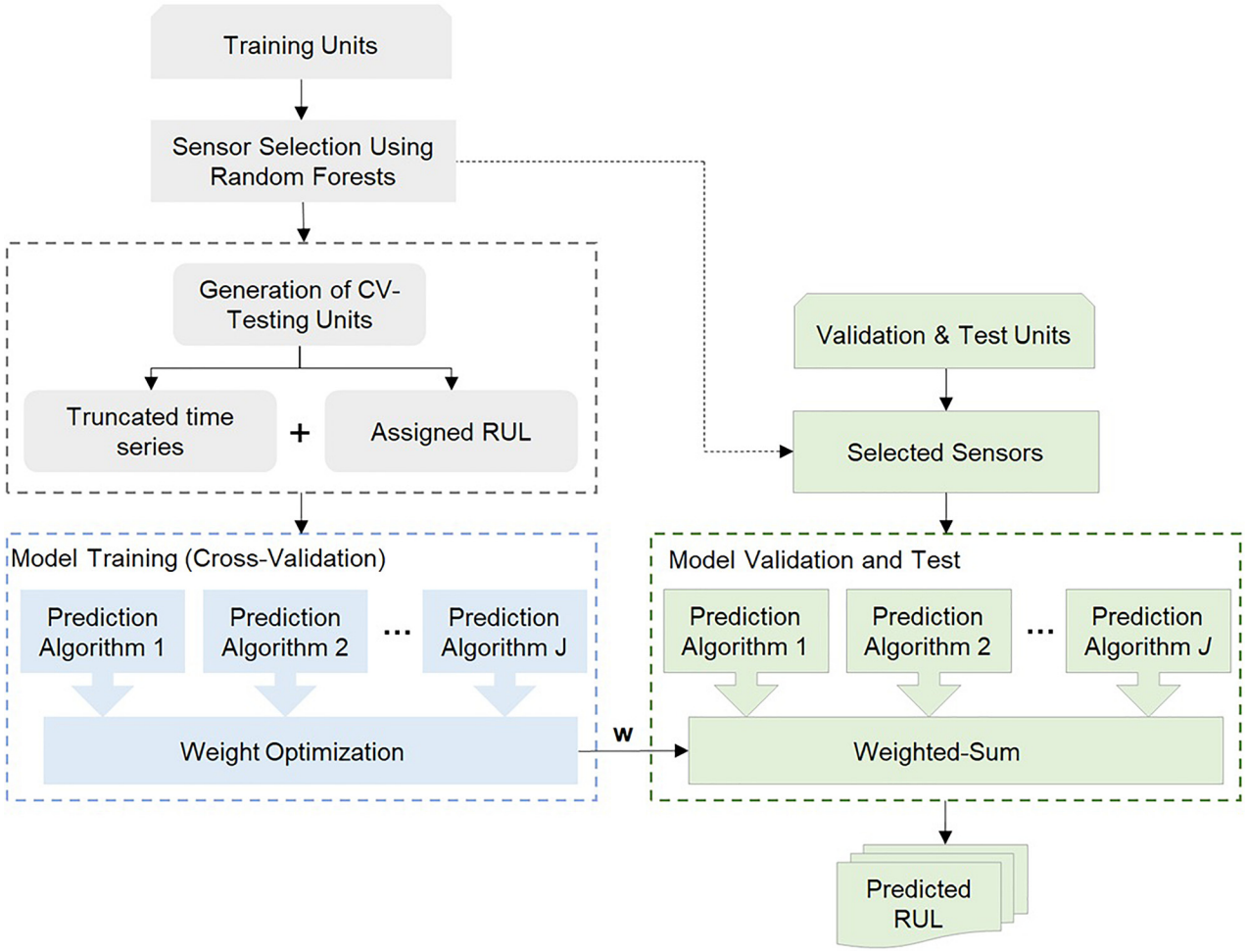

Fig. 1 A computational framework for the ensemble learning-based prognostics

Table 1 Pseudocode of the ensemble learning algorithm

Input:	Training units $X = [x_1, x_2, \ldots, x_N]^{\mathrm{T}}$
1.	Select the most important variables
2.	Generate CV-test units from X with random truncations
3.	Perform k-fold CV to obtain the predicted RUL of the CV-test units
4.	Compute the optimal weight vector
Output:	An optimal weight vector $\mathrm{w} = [w_1, w_2, \ldots, w_J]^{\mathrm{T}}$

method and then validated on the test dataset. The model training process consists of the following two steps:

Step 1: Generate a new training dataset from the original training dataset using random sampling. The T-matrix transformation method [12] is used to transform the Q-dimensional measurement signal into one-dimensional HI. Q denotes the number of variables in the measurement data. The T-matrix transformation can be defined as follows:

$$\mathbf{T} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{S}_{\mathrm{off}} \tag{3}$$

where $\mathbf{F}$, an $N \times Q$ matrix, denotes the Q-dimensional measurement data, N is the number of cycles of the measurement. $\mathbf{F}$ can be further divided into two parts ($\mathbf{F} = [\mathbf{F}_{\mathrm{o}}; \mathbf{F}_1]$) where subscripts 0 and 1 denote failure and healthy states. The dimension of the matrix $\mathbf{F}_{\mathrm{o}}$ is $N_0 \times Q$. The dimension of the matrix $\mathbf{F}_1$ is $N_1 \times Q$. N_0 and N_1 ($N = N_0 + N_1$) denote the number of cycles. The matrix $\mathbf{S}_{\mathrm{off}} = [\mathbf{S}_{\mathrm{o}}; \mathbf{S}_1]$ where $\mathbf{S}_{\mathrm{o}}$ is a $1 \times N_0$ zero-vector and $\mathbf{S}_1$ is a $1 \times N_1$ unit vector. The one-dimensional normalized HI can be calculated by $\mathbf{H} = \mathbf{F} \cdot \mathbf{T}$.

A new training data point is generated for each training unit by randomly selecting the number of cycles and the corresponding HI. The number of cycles ranges between the minimum and

Table 2 Base learning algorithms

Category	Selected algorithm
Ensemble tree-based	RFs
Decision tree-based	CART
ANN-based	RNN
Stochastic model-based	AR
Association rule learning-based	ANFIS
Bayesian-based	RVM
Regularization-based	EN

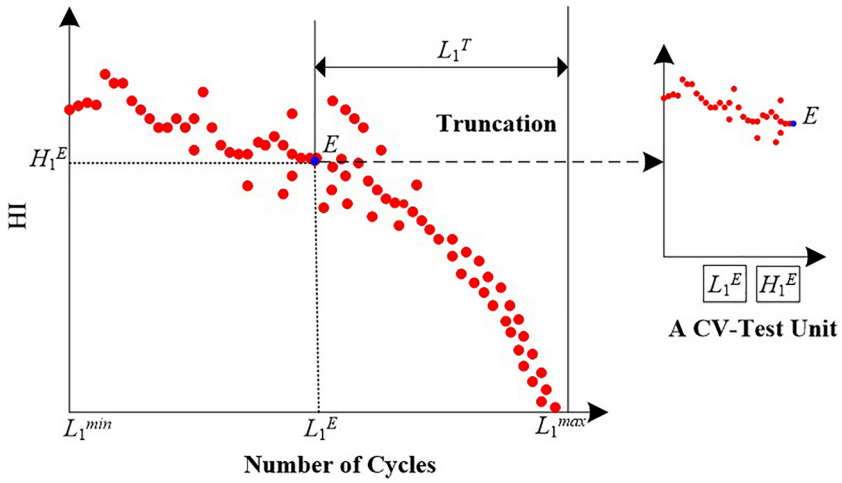

Fig. 2 Generating new training data points from the original training data for training unit ID-1

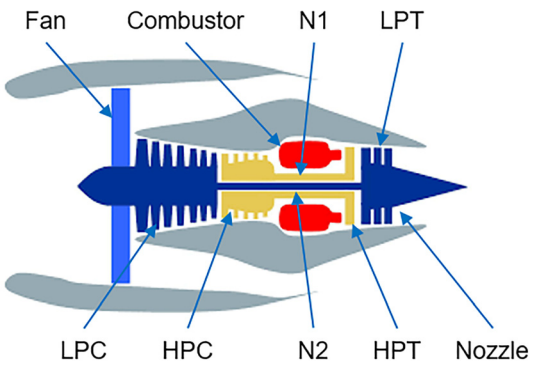

Fig. 3 Simplified diagram of the aircraft engine simulated in C-MAPSS [4]

maximum number of cycles. 2 illustrates how to generate the new training data points for the training unit ID-1. A truncation point E (L_1^T, H_1^E) with L_1^E cycles is randomly selected between the minimum and maximum number of cycles ($L_1^{\min}$ and $L_1^{\max}$). H_1^E is the HI associated with the truncation point E. The new training data points can be defined as the HI values corresponding to cycle $L_1^{\min}$ to cycle L_1^E. The RUL associated with these health indicators is $L_1^T = L_1^{\max} - L_1^E$. Similarly, new training data points can be defined by randomly truncating the time series for the remaining training units.

Step 2: Predict the RULs using the ensemble learning algorithm. To avoid overfitting, *k*-fold CV is conducted. The CV-test units are randomly grouped into *k* disjoint subsets. Each subset contains approximately the same number of units. In cross-validation, only one subset of the CV-test units from the *k* subsets is used for testing; the remaining units are used for training. To improve the performance of the ensemble learning algorithm, the PSO and SQP methods are used to determine the optimal weights for the base learners.

Model validation is performed to evaluate the performance of the predictive model trained by the ensemble learning algorithm. Each base learning algorithm predicts the RUL of the validation or test units. The final prediction is generated using the following weighted-sum function:

$$\hat{L}_t^P = \sum_{j=1}^{J} w_j \hat{L}_t^j(\mathbf{x}_t, \mathbf{X}) \tag{4}$$

where $\mathbf{x}_t$ is a validation or test unit; $\hat{L}_t^P$ is the ensemble RUL for $\mathbf{x}_t$; w_j ($j = 1, 2, \ldots, J$) is the optimum weight assigned to the *j*th algorithm member; $\hat{L}_t^j(\mathbf{x}_t, \mathbf{X})$ denotes the predicted RUL of $\mathbf{x}_t$ by the *j*th algorithm member trained by the dataset $\mathbf{X}$. The predicted RULs by the base learners are aggregated by taking a weighted sum to produce the final prediction. It should be noted that a positive weight would be assigned to an effective base learner in the ensemble. If any base learner does not help to improve the performance, it would be assigned a zero-value weight. In other words, this base learner would not be considered at all in the ensemble.

4 Case Study

4.1 Data Description. In this section, the ensemble learning-based prognostic approach is demonstrated on the FD004 dataset provided by the NASA Prognostics Data Repository. The run-to-failure data in the FD004 dataset were generated by the C-MAPSS tool developed by NASA. The ensemble learning-based prognostic approach is used to model the performance degradation behavior in the high pressure compressor (HPC) and fan modules due to wear. C-MAPSS models a generalized exponential wear behavior $\hat{w} = Ae^{B(t)}$ where A and $B(t)$ denote the amplitude and exponential parameters. The exponential degradation behavior ignores microlevel degradation characteristics but retains macro-level degradation characteristics.

A simplified diagram of the aircraft engine is shown in Fig. 3. The health condition data were collected from the simulations under six different combinations of altitude, throttle resolver angle, and Mach number and two failure modes caused by wear. The dataset consists of 249 training units and 248 test units with 57,522 observations for training and 41,214 observations for testing.

Table 3 lists more details about the six operational conditions. Table 4 lists 21 output variables of the C-MAPSS tool. It should be noted that not all of these variables could be measured in real-world applications. In the simulations performed by the C-MAPSS tool, time series of observables change from some undefined initial condition to a failure threshold. The training dataset includes trajectories that ended at the failure threshold, while the test dataset includes trajectories that end prior to the failure threshold. The number of cycles to failure of the training dataset ranges between 128 and 543 cycles, whereas the RULs of the test dataset range between 6 and 195 cycles.

4.2 Variable Selection. In the model training stage, not all of the variables are useful. Taking into account some variables may even reduce prediction accuracy because these variables may not be correlated to the degradation behavior of aircraft engines. To select the most effective variables, RFs were used to measure the

Table 3 Six operational conditions of the FD004 dataset

Operational Condition	Altitude (Kft)	Mach Number	Throttle resolver angle (deg)
1	42	0.8400	100
2	35	0.8400	100
3	25	0.6200	60
4	20	0.7000	100
5	10	0.2500	100
6	0	0	100

Table 4 Data description [4]

Symbol	Description	Symbol	Description
T2	Total temperature at fan inlet (°R)	Ps30	Static pressure at HPC outlet (psia)
T24	Total temperature at LPC outlet (°R)	Phi	Ratio of fuel flow to Ps30 (pps/psi)
T30	Total temperature at HPC outlet (°R)	NRf	Corrected fan speed (rpm)
T50	Total temperature at LPT outlet (°R)	NRc	Corrected core speed (rpm)
P2	Pressure at fan inlet (psia)	BPR	Bypass ratio
P15	Total pressure in bypass-duct (psia)	farB	Burner fuel-air ratio
P30	Total pressure at HPC outlet (psia)	htBleed	Bleed Enthalpy
Nf	Physical fan speed (rpm)	Nf_dmd	Demanded fan speed (rpm)
Nc	Physical core speed (rpm)	PCNfR_dmd	Demanded corrected fan speed (rpm)
EPR	Engine pressure ratio (P50/P2)	W31	HPT coolant bleed (lbm/s)
		W32	LPT coolant bleed (lbm/s)

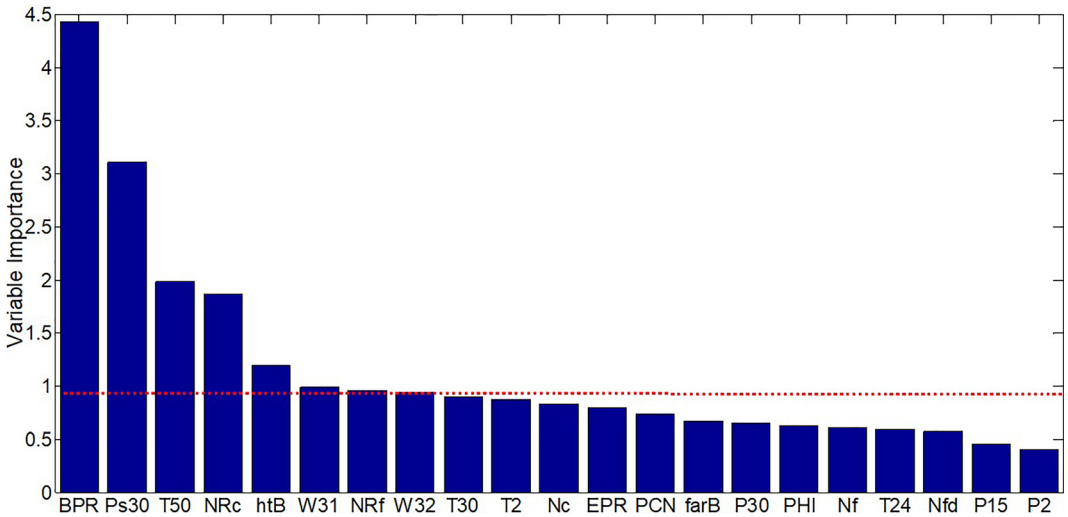

Fig. 4 Variable importance of 21 variables

importance of measurement variables with respect to their performance on prediction accuracy.

Figure 4 shows the variable importance. Based on this criterion, the most important variable is bypass ratio (BPR) (The variable importance is 4.43). The least important variable is P2 (The variable importance is 0.40). In this study, the variables with variable importance greater than a threshold of 0.95 were selected. As shown in Fig. 4, the measurement variables, including BPR, Ps30, T50, NRc, htB, W31 and NRf, were selected for training the predictive models. The variable importance values of these variables are 4.43, 3.11, 1.98, 1.86, 1.20, 0.99, and 0.96. This result is partially consistent with that of Ref. [26] where three measurement variables (T50, htB, and W31) were selected according to the log-normal distributions of the measurement data. In addition, 11 variables were selected in Ref. [27] based on a consistent increasing or decreasing trend is present in the variable measurements. As shown in Table 5, five common variables are selected in both variable selection methods. Therefore, seven variables selected by RFs are used to train the predictive models [5,28]. If a smaller threshold such as 0.9 was selected, more variables will be selected. However, taking into account more variables in model training will increase training time. To balance the trade-off between the number of variables and computational efficiency, a threshold of 0.95 was used and seven variables were selected for training the predictive models.

Figure 5 shows the health indices of 249 training aircraft engines. These health indices were transformed from the original data using the T-matrix transformation and different number of variables. In the T-matrix transformation, N_0 was set to 90% of N. As shown in Fig. 5(*a*), relatively large variations in the health indices were observed at the beginning of the degradation processes of the training units when the original 21 variables were used to compute HI. In addition, sudden decreases in the health indices were observed at the end of the degradation processes of the training units. However, the variations in the health indices should be relatively small because the C-MAPSS tool models a gradual degradation process due to wear. These observations indicate that some redundant variables might be used for computing

Table 5 Variable selection for the base learning algorithms

Number of variables	Variables
11	T24, T30, T50, P30, Nf, PHI, NRf, NRc, htB, W31, and W32
7	BPR, Ps30, T50, NRc, htB, W31 and NRf
3	T50, htB and W31

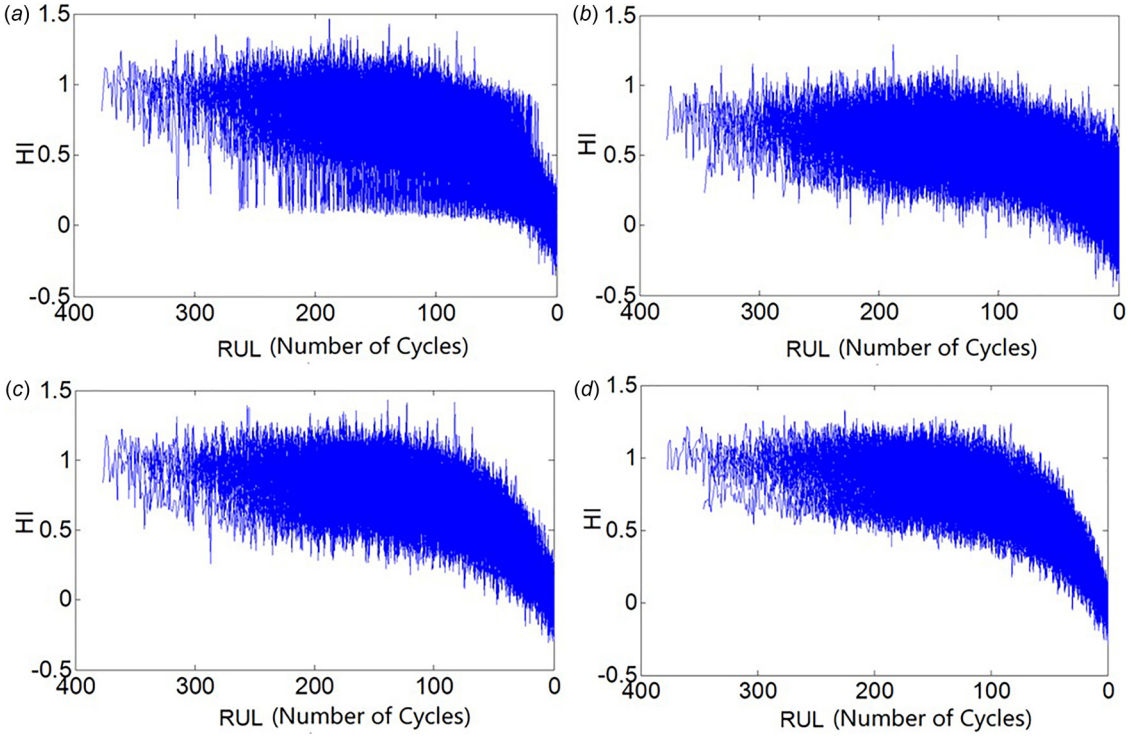

Fig. 5 Health indices associated with 249 training units when using (*a*) 21 variables, (*b*) 3 variables [26], (*c*) 11 variables [27], and (*d*) 7 variables selected by RFs

Table 6 Parameter settings for the base learners

Base learner	Parameter
RFs	Number of trees: 500; Stopping threshold: 5
CART	Stopping criterion: less than ten observations in the nodes
RNN	Number of nodes in the hidden layer: 8; Number of nodes in the output layer: 1
AR	AR order: 30
ANFIS	Number of rules: 15; Number of memberships: 15
RVM	Gaussian kernel width: 0.8; Prior variance $\sigma = 0.1$
EN	Shrinkage $\alpha = 0.5$; Regularization $\lambda = 100$

the health indices. As shown in Figs. 5(*b*)–5(*d*), the health indices were computed using 11, 7, and 3 variables instead of 21 variables. When seven variables were used to compute the health indices, the smallest variations in the health indices were observed. Therefore, seven variables were selected for computing the health indices and training predictive models.

4.3 Optimal Weights for Base Learners. The PSO and SQP methods are used to determine the optimal weights for the base learners. PSO is a population-based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling [16]. SQP is an iterative method for constrained nonlinear optimization [28]. The objective of the optimization methods is to minimize three types of errors, including root mean square error (RMSE), relative error (RE), and S-score (see Eqs. (5)–(7)). RMSE and RE measure the deviations between the predicted and actual RULs. An S-score, initially introduced in 2008 PHM Data Challenge, measures the performance of a model by taking into account whether the model overestimates and underestimates the RULs.

$$\text{RMSE} \quad \varepsilon_{\text{RMSE}} = \sqrt{\text{E}[(\hat{\mathbf{y}} - \mathbf{y})^2]} \tag{5}$$

$$\text{RE} \quad \varepsilon_{\text{RE}i} = |\hat{y}_i - y_i|/y_i \tag{6}$$

$$\text{S}-\text{score} \quad \varepsilon_{\text{CV}i} = \begin{cases} \exp(-d_i/13), & d_i < 0 \\ \exp(d_i/10), & d_i \geq 0 \end{cases}, (d_i = \hat{y}_i - y_i) \tag{7}$$

where $\text{E}(\cdot)$ denotes the expectation operator, y_i is the actual RUL of the ith unit, $\hat{y}_i$ is the predicted RUL of the ith unit, $\hat{\mathbf{y}}$ is the

Table 7 S-score and RE of the predictive models trained on different number of variables

	Number of variables							
	21		11		7		3	
Base learner	S-score	RE	S-score	RE	S-score	RE	S-score	RE
RFs	69.411	0.377	29.222	0.508	51.167	0.493	675.30	1.404
CART	3.95×10^5	0.252	3.955×10^5	0.295	112.677	0.303	1.640×10^3	0.444
RNN	3.605×10^7	7.294	3.406×10^7	7.269	3.388×10^7	7.275	1.717×10^7	6.987
AR	2.502×10^5	5.260	2.502×10^5	5.260	2.502×10^5	5.260	2.502×10^5	5.260
ANFIS	1.668×10^3	1.005	2.096×10^3	1.055	3.184×10^3	1.093	3.833×10^4	3.563
RVM	1.533×10^3	1.403	1.594×10^3	1.380	1.513×10^3	1.376	2.640×10^3	1.217
El-Net	2.858×10^7	4.024	1.166×10^5	1.015	1.152×10^5	1.025	1.168×10^5	1.000
Average	9.326×10^6	2.802	4.975×10^6	2.397	4.893×10^6	2.404	2.512×10^6	2.839

Table 8 RMSE of the predictive models trained on different number of variables

	RMSE of the base learners							
Number of variables	RFs	CART	RNN	AR	ANFIS	RVM	El-Net	Average RMSE
21	25.578	21.647	139.427	95.030	38.592	36.862	142.817	71.4219
11	22.658	25.943	138.859	95.030	43.203	36.794	76.901	62.7697
7	24.380	20.950	139.230	95.030	43.406	36.574	76.879	62.3499
3	35.269	30.481	133.928	95.030	67.103	42.627	76.897	68.7621

Table 9 Optimal weights for base learners and prediction accuracy

			Prediction error		
Number of variables	Method	Weight vector **w**	S-score	RE	RMSE
21	SQP	[0.398, 0.315, 0.077, 0, 0.062, 0, 0.148]	5.669	0.288	15.754
	PSO	[0.398, 0.316, 0.075, 0, 0.065, 0, 0.146]	5.672	0.281	15.731
11	SQP	[0.594, 0.207, 0.005, 0, 0, 0, 0.194]	3.560	0.305	14.733
	PSO	[0.595, 0.207, 0.004, 0, 0, 0, 0.194]	3.560	0.304	14.733
7	SQP	[0.407, 0.408, 0, 0, 0, 0, 0.184]	5.673	**0.257**	**14.281**
	PSO	[0.430, 0.469, 0, 0, 0, 0, 0.100]	**3.400**	0.284	14.908
3	SQP	[0.488, 0.310, 0, 0, 0, 0.078, 0.124]	34.793	0.800	22.050
	PSO	[0.488, 0.311, 0, 0, 0, 0.077, 0.115]	34.794	0.800	22.063

matrix form of all predicted RULs, and $\bar{y}$ is the mean value of the actual RUL vector **y**, $d_i = \hat{L}_i - L_i^{\mathrm{T}}$, $\hat{L}_i$ and L_i^{T} denote the predicted and actual RUL of the ith CV-test unit. Tenfold cross-validation was conducted on 249 CV-test units in the FD004 dataset. The parameter settings for the based learners are shown in Table 6.

Table 7 shows the prediction errors associated with seven base learners. The prediction errors for the CV-test data vary depending on the base learners. The smallest RE is 0.252, which is achieved by CART using 21 variables. The average RE when using seven variables is 2.404, which is comparable to the average RE (i.e., 2.397) when using 11 variables, and is less than the average REs (i.e., 2.802 and 2.839) when using 21 and 3 variables. The smallest S-score is 29.222, which is achieved by RFs using 11 variables. The average S-score when using seven variables is 4.893×10^6, which is comparable to the average S-score (i.e., 2.512×10^6) when using three variables, and is less than the average S-scores (i.e., 9.326×10^6 and 4.975×10^6) when using 21 and 11 variables. In addition, the CART with seven variables made significant improvements over the other three variable selection methods with respect to S-score value. The RMSE results are provided in Table 8 where the proposed variable selection method produced the smallest RMSE (20.95) with CART. The average RMSE over all of the base learners demonstrated better performance of the proposed selection method than the other three selection methods.

To improve prediction accuracy, the proposed ensemble learning method combines the base learners by assigning a weight to each base learner. The PSO technique was used to determine the optimum weight vector. To evaluate the performance of PSO, the SQP method was also used to determine the weight vector. As shown in Tables 7–9, both PSO- and SQP-based ensemble learning algorithms outperform the individual base learners significantly in terms of S-score, RE, and RMSE when training the predictive models using 21, 11, 7, and 3 variables. As shown in Table 9, the predictive models trained by PSO- and SQP-based ensemble learning algorithms using seven variables achieved the best performance in terms of in terms of S-score, RE, and RMSE. For instance, the PSO-based ensemble learning algorithm achieved an S-score of 5.672 using 21 variables, whereas the PSO-based ensemble learning algorithm achieved an S-score of 3.400 using 7 variables. Similarly, the SQP-based ensemble learning algorithm achieved a RE of 0.288 and a RMSE of 15.754 using 21 variables, whereas the SQP-based ensemble learning algorithm achieved an RE of 0.257 and an RMSE of 14.281 using seven variables. The reason why training the predictive models using seven variables achieved the best performance is that the redundant variables that are not correlated to the degradation behavior of aircraft engines were not included in the model training process. In addition, based on the weight vector determined by both PSO and SQP algorithms, some of the base learners were not used to train the predictive models. For example, when training

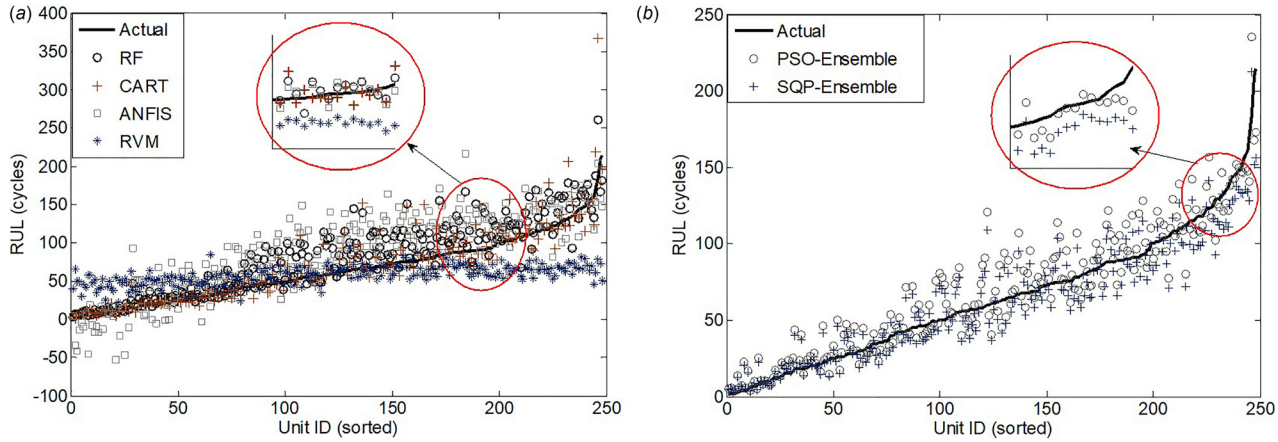

Fig. 6 RUL prediction performance on CV-test data with seven variables: (*a*) base learners and (*b*) ensemble learning

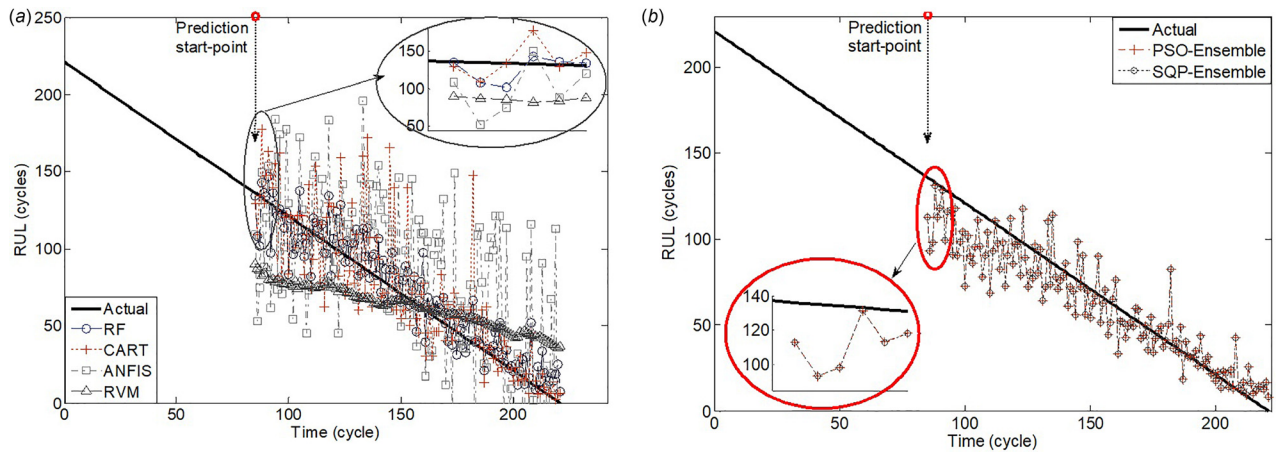

Fig. 7 RUL predictions for one CV-test unit: (*a*) base learners and (*b*) ensemble learning

the predictive models using seven variables, only RFs, CART, and El-Net were combined by the PSO- and SQP-based ensemble learning algorithms because the weights assigned to other base learners are zero. By determining the optimal weight vector, the ensemble learning algorithms selected the base learners with the best performance.

Figure 6 shows the performance of the base learners and ensemble learning algorithms for all of the 249 CV-test units. Figure 6(*a*) provides the prediction performance of four base learners (i.e., RFs, CART, ANFIS, and RVM). The prediction results of the other three algorithms (RNN, AR and El-Net) are not provided in Fig. 6(*a*) due to large errors. Figure 6(*b*) compares the RUL prediction results of the ensemble learning using SQP and PSO optimization. The closer the predicted RULs to the actual RUL curve, the more accurate are the predicted RULs. It can be observed in Fig. 6(*b*) that the PSO-based method slightly outperformed the SQP-based method because predicted RULs by the former (circles) were slightly closer to the actual RUL than that of the latter (crosses). As shown in Figs. 6(*a*) and 6(*b*), both PSO- and SQP-based ensemble learning algorithms outperformed the base learners.

Figure 7 shows more details on the performance of the predictive model of an individual aircraft engine unit (i.e., unit ID-35) from the beginning to end of life of the entire lifecycle. The total number of cycles of unit ID-35 was 221. The predictive model started to predict RUL after 85 cycles. As shown in Fig. 7(*a*), most of the predictions produced by four of the base learners were late predictions. Late predictions refer to the cases where the predicted RULs are greater than the actual RULs. Since the objective of predicting the RUL of aircraft engines is to avoid failures, it is important to predict failures early as compared to predicting failure late. As shown in Fig. 7(*b*), when training the predictive model using the ensemble learning algorithm, most of the predicted RULs were early predictions. In addition, by comparing Fig. 7(*b*) with Fig. 7(*a*), the predictive model trained by the ensemble learning algorithm achieved an S-score of 3.670 and a RMSE of 16.110, whereas the predictive model trained by CART achieved an S-score of 518.892 and a RMSE of 25.367. In addition, Fig. 7(*b*) shows that the predictive model trained by the ensemble learning algorithm achieved very high prediction accuracy when the engine unit is close to the end of life.

4.4 Model Validation and Performance Comparison. The test dataset was used to evaluate the performance of the predictive model trained on the training dataset. Figure 8 shows the prediction results using the base learners and ensemble learning method. It has been observed that prediction accuracy is very high for those test units that have longer RULs, whereas prediction accuracy is relatively low for those test units that have shorter RULs. For example, the actual RULs for test units ID-22 and ID-121 are 11 and 41 cycles. The predicted RULs for the test units are 16 and 40, respectively. The predictive model is very accurate for these test units. However, the actual RULs for test unit ID-246 are 194 cycles, whereas the predicted RUL is 126 cycles. Similarly, the actual RUL for test unit ID-204 is 151 cycles, whereas the predicted RUL is 88 cycles. The predictive model is not accurate for these test units. This is because test unit ID-22 (The number of

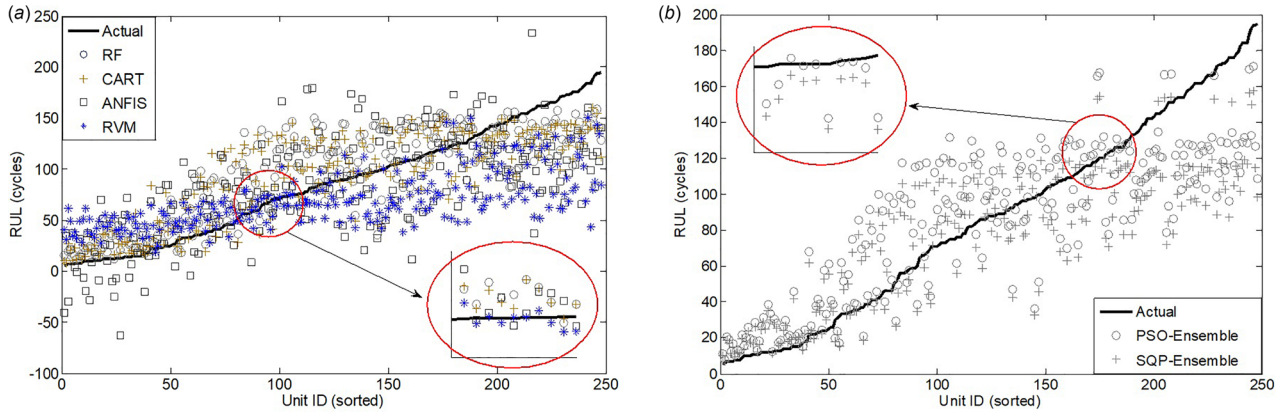

Fig. 8 RUL prediction for 248 test units using (*a*) base learners and (*b*) ensemble learning

Table 10 Performance of the predictive models on the test dataset

	Performance metrics		
Prognostic approach	S-score	RE	RMSE
RFs	75.266	0.485	105.065
CART	74.115	0.505	101.897
RNN	2.010×10^7	4.175	121.579
AR	6.469×10^{11}	6.997	220.301
ANFIS	3.003×10^4	0.732	45.208
RVM	448.695	0.803	44.519
El-Net	1.344×10^5	1.012	102.372
SQP-based Ensemble	28.958	0.373	31.622
PSO-based Ensemble	26.382	0.420	29.512

cycles that has been observed is 158) and ID-121 (The number of cycles that has been observed is 194) were operating in the late stage of the degradation process. However, test units ID-246 (The number of cycles that has been observed is 29) and ID-204 (The number of cycles that has been observed is 19) were operating in the early stage of the degradation process.

Table 10 lists the S-score, RE, and RMSE values of the predictive model for the test dataset. Both PSO- and SQP-based ensemble learning methods outperform the base learners significantly. For example, the mean S-score, RE, and RMSE values of PSO- and SQP-based ensemble learning methods are 28.958, 0.373, 31.622 and 26.382, 0.420, and 29.512, respectively. Moreover, a comparative study between the ensemble learning methods and 16 other methods was conducted. These methods include multilayer perceptron (MLP), SVR, extra-randomized trees, KNN, SVM, least absolute shrinkage and selection operator (LASSO), extremely learning machine (ELM), hierarchical ELM, relevance vector regression (RVR), gradient boosting (GB), similarity-based inference (SBI), discriminating shapelet extraction, deep belief network (DBN), deep convolutional neural network (CNN), multi-objective deep belief networks ensemble, and deep CNN without rectified labels. As shown in Table 11, both PSO- and SQP-based ensemble learning methods outperform MLP, SVR, extra-randomized trees, KNN, SVM, LASSO, ELM, hierarchical ELM, RVR, GB, SBI, discriminating shapelet extraction, DBN, and deep CNN in terms of S-score and RMSE. In addition, both PSO- and SQP-based ensemble learning methods are comparable with two deep learning algorithms, including the multi-objective deep belief networks ensemble and deep CNN without rectified labels.

5 Conclusions and Future Work

This paper has presented an ensemble learning-based prognostic approach to degradation modeling and RUL prediction of aircraft engines. A variable selection approach using RFs was used

Table 11 Performance comparisons between the ensemble learning methods and other existing methods

	FD004	
Prognostic approach	S-score	RMSE
MLP [29]	2.266×10^4	77.37
SVR [29]	1.496×10^3	45.35
ETR [30]	1.395×10^3	40.01
KNN [30]	945.147	54.44
SVM [30]	569.041	59.96
LASSO [30]	505.231	40.70
ELM [30]	489.575	38.43
Hierarchical ELM [30]	425.374	37.98
RVR [29]	106.855	34.34
GB [30]	71.847	29.01
SBI [31]	69.2928	NA
Discriminating shapelet extraction [31]	37.7097	NA
DBN [30]	32.0746	29.88
Deep CNN [29]	31.815	29.16
Multi-objective DBNs ensemble [30]	26.442	28.66
Deep CNN without rectified labels [32]	NA	29.44
Proposed method with SQP optimization	**28.958**	**31.62**
Proposed method with PSO optimization	**26.382**	**29.51**

to determine an optimal set of variables/features. The ensemble learning algorithm combined RFs, CART, RNN, AR model, ANFIS, RVM, and EN using the parallel ensemble technique. The PSO and SQP algorithm were used to optimize the prediction performance by assigning an optimal weight to each base learner. The predictive model trained by the ensemble learning algorithm has been demonstrated on the data generated by the C-MAPSS. The experimental results have shown that the ensemble learning-based prognostic approach can predict the RUL of the aircraft engine with very high accuracy. The PSO and SQP optimization methods selected not only the most accurate base learner (i.e., stronger learner) but also several less accurate base learners for fusion. In the future, we will parallelize the training process of the ensemble learning-based prognostic approach. In addition, a topic of particular interest is to explore base learner diversity. As alluded to above, different base learners were chosen depending on the number of variables. Variable diversity and base learner diversity are important aspects that deserve more attention. Finally, the fusion method itself calls for further investigation. It should be expected, for example, that a time-dependent fusion method might perform even better. We will also test and validate the proposed approach on other C-MAPSS datasets.

Acknowledgment

The research reported in this paper is partially supported by the University of Central Florida (UCF) and the Digital Manufacturing and Design Innovation Institute (DMDII). Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the UCF and the DMDII.

Nomenclature

AR = autoregressive
ANFIS = adaptive network-based fuzzy inference system
BPR = bypass ratio
C-MAPSS = commercial modular aero-propulsion system simulation
CART = classification and regression tree
CV = cross-validation
EN = elastic net
EPR = engine pressure ratio
HPC = high pressure compressor
HPT = high-pressure turbine
HI = health index
LPC = low pressure compressor
LPT = low pressure turbine
PSO = particle swarm optimization
PPS = pound-mass per second
RFs = random forests
RUL = remaining useful life
RNN = recurrent neural networks
RVM = relevance vector machine
SQP = sequential quadratic optimization
TRA = throttle resolver angle

References

[1] Lee, S., Ma, Y.-S., Thimm, G., and Verstraeten, J., 2008, "Product Lifecycle Management in Aviation Maintenance, Repair and Overhaul," Comput. Ind., **59**(2–3), pp. 296–303.
[2] Wu, D., Jennings, C., Terpenny, J., Gao, R. X., and Kumara, S., 2017, "A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests," ASME J. Manuf. Sci. Eng., **139**(7), p. 071018.
[3] Li, Z., Wu, D., Hu, C., and Terpenny, J., 2017, "An Ensemble Learning-Based Prognostic Approach With Degradation-Dependent Weights for Remaining Useful Life Prediction," Reliab. Eng. Syst. Saf. (in press).
[4] Saxena, A., Goebel, K., Simon, D., and Eklund, N., 2008, "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation," International Conference on Prognostics and Health Management (PHM), Denver, CO, Oct. 6–9, pp. 1–9.
[5] Schwabacher, M., and Goebel, K., 2018, "A Survey of Artificial Intelligence for Prognostics," AAAI Fall Symposium, Arlington, VA, Oct. 18–20, pp. 107–114.
[6] Wang, P., and Gao, R. X., 2016, "Markov Nonlinear System Estimation for Engine Performance Tracking," ASME J. Eng. Gas Turbines Power, **138**(9), p. 091201.
[7] Mosallam, A., Medjaher, K., and Zerhouni, N., 2016, "Data-Driven Prognostic Method Based on Bayesian Approaches for Direct Remaining Useful Life Prediction," J. Intell. Manuf., **27**(5), pp. 1037–1048.
[8] Liu, K., and Huang, S., 2016, "Integration of Data Fusion Methodology and Degradation Modeling Process to Improve Prognostics," IEEE Trans. Autom. Sci. Eng., **13**(1), pp. 344–354.
[9] Nieto, P. G., Garcia-Gonzalo, E., Lasheras, F. S., and de Cos Juez, F. J., 2015, "Hybrid PSO–SVM-Based Method for Forecasting of the Remaining Useful Life for Aircraft Engines and Evaluation of Its Reliability," Reliab. Eng. Syst. Saf., **138**, pp. 219–231.
[10] Khelif, R., Chebel-Morello, B., Malinowski, S., Laajili, E., Fnaiech, F., and Zerhouni, N., 2017, "Direct Remaining Useful Life Estimation Based on Support Vector Regression," IEEE Trans. Ind. Electron., **64**(3), pp. 2276–2285.
[11] Yu, J., 2017, "Aircraft Engine Health Prognostics Based on Logistic Regression With Penalization Regularization and State-Space-Based Degradation Framework," Aerosp. Sci. Technol., **68**, pp. 345–361.
[12] Hu, C., Youn, B. D., Wang, P., and Yoon, J. T., 2012, "Ensemble of Data-Driven Prognostic Algorithms for Robust Prediction of Remaining Useful Life," Reliab. Eng. Syst. Saf., **103**, pp. 120–135.
[13] Ramasso, E., and Gouriveau, R., 2014, "Remaining Useful Life Estimation by Classification of Predictions Based on a Neuro-Fuzzy System and Theory of Belief Functions," IEEE Trans. Reliab., **63**(2), pp. 555–566.
[14] Chen, H., 2011, "A Multiple Model Prediction Algorithm for CNC Machine Wear PHM," Int. J. Prognostics Health Manage., **2**, p. 129.
[15] Goebel, K., Eklund, N., and Bonanni, P., 2006, "Fusing Competing Prediction Algorithms for Prognostics," IEEE Aerospace Conference, Big Sky, MT, Mar. 4–11, p. 10.
[16] Kennedy, J., 2011, "Particle Swarm Optimization," *Encyclopedia of Machine Learning*, Springer, Boston, MA, pp. 760–766.
[17] Breiman, L., 2001, "Random Forests," Mach. Learn., **45**(1), pp. 5–32.
[18] Prasad, A. M., Iverson, L. R., and Liaw, A., 2006, "Newer Classification and Regression Tree Techniques: Bagging and Random Forests for Ecological Prediction," Ecosystems, **9**(2), pp. 181–199.
[19] Loh, W. Y., 2011, "Classification and Regression Trees," Wiley Interdiscip. Rev.: Data Min. Knowl. Discovery, **1**(1), pp. 14–23.
[20] Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., and Siegel, D., 2014, "Prognostics and Health Management Design for Rotary Machinery Systems—Reviews, Methodology and Applications," Mech. Syst. Signal Process., **42**(1–2), pp. 314–334.
[21] Čerňanský, M., Makula, M., and Beňušková, Ľ., 2007, "Organization of the State Space of a Simple Recurrent Network before and After Training on Recursive Linguistic Structures," Neural Networks, **20**(2), pp. 236–244.
[22] Akaike, H., 1969, "Fitting Autoregressive Models for Prediction," Ann. Institute Stat. Math., **21**(1), pp. 243–247.
[23] Jang, J.-S., 1993, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," IEEE Trans. Syst., Man, Cybern., **23**(3), pp. 665–685.
[24] Tipping, M. E., 2001, "Sparse Bayesian Learning and the Relevance Vector Machine," J. Mach. Learn. Res., **1**, pp. 211–244.
[25] Zou, H., and Hastie, T., 2005, "Regularization and Variable Selection Via the Elastic Net," J. R. Stat. Soc.: Ser. B (Stat. Methodology), **67**(2), pp. 301–320.
[26] Fang, X., Paynabar, K., and Gebraeel, N., 2017, "Multistream Sensor Fusion-Based Prognostics Model for Systems With Single Failure Modes," Reliab. Eng. Syst. Saf., **159**, pp. 322–331.
[27] Yan, H., Liu, K., Zhang, X., and Shi, J., 2016, "Multiple Sensor Data Fusion for Degradation Modeling and Prognostics Under Multiple Operational Conditions," IEEE Trans. Reliab., **65**(3), pp. 1416–1426.
[28] Nocedal, J., and Wright, S. J., 2006, *Sequential Quadratic Programming*, Springer, New York.
[29] Babu, G. S., Zhao, P., and Li, X.-L., "Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life," *Database Systems for Advanced Applications* (Lecture Notes in Computer Science, Vol. 9642), S. Navathe, W. Wu, S. Shekhar, X. Du, X. Wang, and H. Xiong, eds., Springer, Cham, Switzerland, pp. 214–228.
[30] Zhang, C., Lim, P., Qin, A., and Tan, K. C., 2017, "Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics," IEEE Trans. Neural Networks Learn. Syst., **28**(10), pp. 2306–2318.
[31] Malinowski, S., Chebel-Morello, B., and Zerhouni, N., 2015, "Remaining Useful Life Estimation Based on Discriminating Shapelet Extraction," Reliab. Eng. Syst. Saf., **142**, pp. 279–288.
[32] Li, X., Ding, Q., and Sun, J.-Q., 2018, "Remaining Useful Life Estimation in Prognostics Using Deep Convolution Neural Networks," Reliab. Eng. Syst. Saf., **172**, pp. 1–11.