applied sciences

MDPI

Article

Novel Prognostic Methodology of Bootstrap Forest and Hyperbolic Tangent Boosted Neural Network for Aircraft System

Shuai Fu * and Nicolas P. Avdelidis

IVHM Centre, School of Aerospace, Transport and Manufacturing, Cranfield University, Bedford MK43 0AL, UK; np.avdel@cranfield.ac.uk

* Correspondence: felix.fu@cranfield.ac.uk

Featured Application: This research work can be utilised to integrate prognostic methodologies in order to enhance the longevity of crucial aircraft systems and assist maintenance strategies in reducing operational and environmental risks while ensuring normal functioning.

Abstract: Complex aviation systems' integrity deteriorates over time due to operational factors; hence, the ability to forecast component remaining useful life (RUL) is vital to their optimal operation. Data-driven prognostic models are essential for system RUL prediction. These models benefit run-to-failure datasets the most. Thus, significant factors that could affect systematic integrity must be examined to quantify the operational component of RUL. To expand predictive approaches, the authors of this research developed a novel method for calculating the RUL of a group of aircraft engines using the N-CMAPSS dataset, which provides simulated degradation trajectories under real flight conditions. They offered bootstrap trees and hyperbolic tangent NtanH(3)Boost(20) neural networks as prognostic alternatives. The hyperbolic tangent boosted neural network uses damage propagation modelling based on earlier research and adds two accuracy levels. The suggested neural network architecture activates with the hyperbolic tangent function. This extension links the deterioration process to its operating history, improving degradation modelling. During validation, models accurately predicted observed flight cycles with 95–97% accuracy. We can use this work to combine prognostic approaches to extend the lifespan of critical aircraft systems and assist maintenance approaches in reducing operational and environmental hazards, all while maintaining normal operation. The proposed methodology yields promising results, making it suitable for adoption due to its relevance to prognostic difficulties.

Keywords: prognostic and health management; neural network; remaining useful life; aircraft systems; bootstrap forest

Citation: Fu, S.; Avdelidis, N.P. Novel Prognostic Methodology of Bootstrap Forest and Hyperbolic Tangent Boosted Neural Network for Aircraft System. *Appl. Sci.* **2024**, *14*, 5057. https://doi.org/10.3390/app14125057

Academic Editors: Rosario Pecora and Juan P. Torreglosa

Received: 9 April 2024
Revised: 31 May 2024
Accepted: 4 June 2024
Published: 10 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The effectiveness of aircraft systems in industrial performance is contingent upon reliability, availability, and safety considerations, necessitating the need for improved maintenance assistance. The conventional notions of preventive and corrective techniques are being supplanted by emerging concepts such as predictive and proactive maintenance [1,2].

The fundamental maintenance principles encompass fault detection, failure diagnosis, and response development, which are essential for perceiving phenomena, comprehending them, and taking appropriate action. In contrast to merely comprehending a newly developed incident, such as a failure, it is advantageous to proactively foresee it in order to promptly initiate appropriate measures. This might be characterised as a prognostic procedure. According to the International Organisation for Standardisation (ISO), prognostics refers to the process of estimating the time till failure and assessing the associated risk for one

or more failure modes, both current and potential. An alternative way to express this concept is to estimate a system's lifespan or remaining useful life (RUL), taking into account its present condition and prior performance [3,4]. Prognostic maintenance methodologies have the potential to reduce costs and improve dependability.

Numerous methodologies have been suggested to facilitate the anticipation of RUL and prognosis. However, traditional limitations such as data accessibility, system intricacy, implementation prerequisites, and the presence of monitoring equipment hinge on the selection of an appropriate strategy. There is a lack of consensus regarding a universally accepted set of metrics suitable for prognostics. The process of selecting appropriate metrics for prognostics is inherently uncertain for both researchers and practitioners in the field of condition-based monitoring (CBM). Therefore, when utilising decisional metrics to determine preventive actions, prognostic tools must consider this inherent uncertainty.

There are a multitude of potential methodologies for data analysis in the context of predictive maintenance. The most widely favoured options are enumerated as follows. The physics-based methodology necessitates a precise physical representation of system behaviour, encompassing both normal and defective states [5]. By comparing the data obtained from sensors with the model's predictions, one can infer the health of the system [6]. Physical techniques encompass the utilisation of physics-of-failure models. One approach to failure mode analysis entails integrating experiments, observation, geometry, and condition monitoring data in order to assess the extent of damage caused by a specific failure mechanism.

Data-driven methods use historical data to make predictions about a system's future state or identify similar trends in the past in order to estimate RUL [7]. The aforementioned factors encompass statistical methodologies, reliability functions, and artificial intelligence (AI) techniques [8]. The following is a list of the subsequent prognostic approaches based on the data:

- The various techniques employed in the field of artificial intelligence include neural networks, fuzzy logic, decision trees, support vector machines (SVMs), anomaly detection algorithms, reinforcement learning, classification, clustering, and Bayesian methods.
- Academic research and practical applications commonly employ various conventional numerical techniques, such as wavelets, Kalman filters, particle filters, regression, and statistical methods.
- There are various statistical approaches utilised in research, such as the gamma process, hidden Markov model, regression-based model, relevance vector machine (RVM), and numerous more.

Data-driven approaches rely on the analysis and utilization of historical run-to-failure data. These strategies are frequently employed to make estimations by relying on a pre-established threshold for failure. The use of a wavelet packet decomposition technique and hidden Markov models (HMMs) can be beneficial, as the incorporation of time-frequency information enables more accurate outcomes compared to solely considering temporal factors. Nevertheless, the methodologies that rely on historical data for asset lifespan prediction necessitate a comprehensive understanding of the asset's physical characteristics [9,10]. This paper employs the N-CMAPSS dataset, which provides simulated run-to-failure data for aircraft engines under real flight conditions.

This paper describes a new way to make predictions that combines bootstrap forest and hyperbolic tangent-boosted neural networks. It is meant to fill in the gaps in current methods for predicting when aircraft systems will need maintenance. Despite the success of traditional prognostic approaches, their capacity to manage complex, nonlinear deterioration patterns in aviation components remains restricted. The proposed method improves the accuracy and reliability of predictions by combining the collective power of bootstrap forests with the adaptive learning skills of hyperbolic tangent neural networks. This hybrid model greatly enhances the accuracy of predictions for the RUL in different

operational settings. This is a crucial improvement compared to previous models, which often face difficulties in dealing with the changing characteristics of aircraft system data. It offers a more precise and dependable approach to forecasting the RUL of crucial components. This facilitates the implementation of improved maintenance methods and helps to ensure safer and more efficient aircraft operations. This work specifically focuses on modelling the deterioration of selected fault types, which flow and efficiency modulation can accurately represent. It acknowledges that this limitation is important to ensure the precision and application of the model.

The structure of the current paper is as follows: the first chapter offers a comprehensive review of the prognostic methods currently employed in the field of RUL prediction. Chapter two provides an overview of the suggested data-driven prognostic approaches and their underlying assumptions. Chapter three presents a comparative examination of a hyperbolic tangent integrated neural network, which uses datasets from aviation engines and other data-driven techniques. Chapter four provides an overview of the RUL prediction's results. The conclusion is outlined in chapter five.

2. Proposed Data-Driven Approaches

2.1. Bootstrap Forest

Bootstrap forest pertains to the statistical methodology called bootstrap aggregation, or bagging, which is utilised on a collection of decision trees inside an ensemble model, resulting in the averaging of several trees [11]. The training of each decision tree involves the utilisation of a bootstrap sample derived from the training data. The splitting process in each tree entails the evaluation of a randomly chosen set of predictors. Through the utilisation of this methodology, a multitude of models possessing restricted individual capacities are combined to produce a more resilient and impactful model. The final prediction for an observation is derived by computing the average of the expected values associated with that observation across all decision trees.

The predictive capacity of a bootstrap forest is attained by aggregating the anticipated response values from many decision trees, which are then averaged [12]. Each individual tree in the model is nurtured by employing a bootstrap sample that is derived from the training data. A bootstrap sample is defined as a subset of observations that is selected randomly and with replacement from a given dataset. Additionally, the predictors are sampled at each split point within the decision tree. The construction of the decision tree is accomplished by utilising the recursive partitioning technique [13]. The subsequent steps outline the methodology utilised for the fitting of the training set [14].

- In order to enhance the robustness of the analysis, it is advisable to use a bootstrap sampling strategy for the selection of data for each individual tree.
- The individual decision tree is constructed using a recursive partitioning algorithm. To perform each split, a set of predictors is randomly chosen, and the order of the predictors within the set is randomised. The procedure of partitioning should be iterated until a predetermined termination criterion, as delineated in the bootstrap forest configuration, is met.
- The iterative procedure of executing steps one and two will persist until the desired quantity of trees specified is attained, or unless early stopping is activated.

Boosting refers to the methodology of constructing a substantial tree through the iterative fitting of a series of smaller decision trees, known as layers. The tree at each stratum is composed of a limited number of splits [15]. The condition of the tree is determined by evaluating the residuals of the preceding levels, enabling each layer to rectify the fit for inadequately fitting data inherited from the preceding layers. The ultimate forecast for an observation is obtained by aggregating the individual predictions for that observation across all levels.

The bootstrap forest generates an ensemble model consisting of many decision trees, which are developed in a sequential manner to form an additive model [16]. Each tier of

the tree is composed of a limited number of splits. The recursive fitting process mentioned is employed to fit each layer. The sole distinction lies in the fact that the fitting process concludes upon reaching a predetermined number of divisions. The predicted value for an observation in a leaf of a given tree is determined by calculating the mean of all observations included within that leaf. Figure 1 below shows the first branch of the first tree of the proposed 100 distributed bootstrap forests.

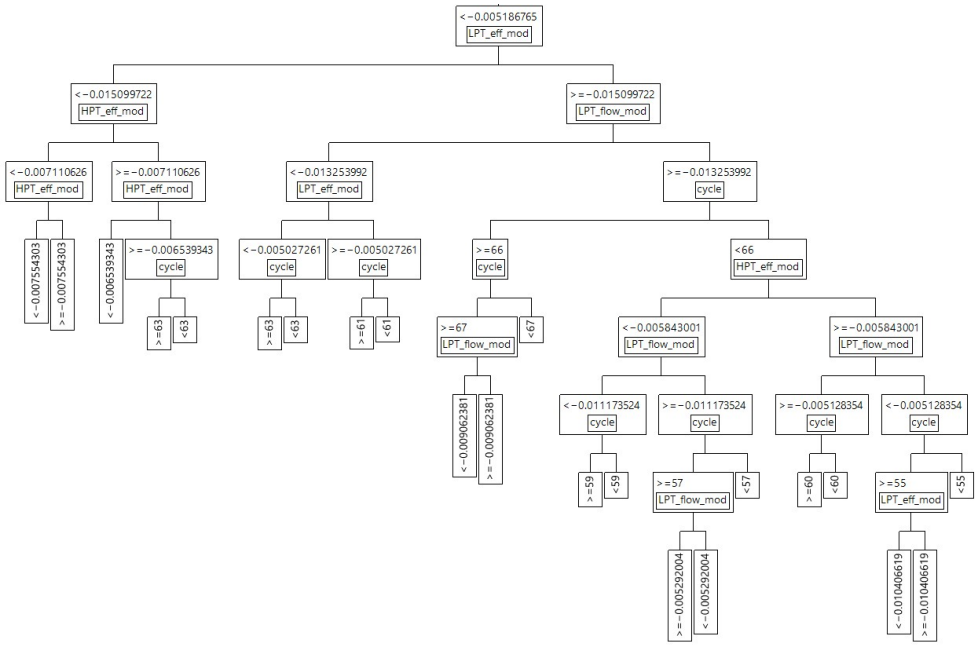


Figure 1. Illustration of the initial branch of a complete bootstrap forest, which is used for predicting the RUL. Each branch in the forest has a comparable structure.

The following steps delineate the methodology for the process of training:

1. To begin, it is recommended to apply an initial layer.
2. Compute the residuals. The numbers mentioned are obtained by subtracting the expected mean of observations contained inside a leaf from their actual numerical values.
3. Employ a layer on the remaining values.
4. The construction of the additive tree is necessary. In order to get the total expected value across several layers for a given observation, it is necessary to aggregate the projected values obtained from each individual layer.
5. The aforementioned procedure, as outlined in steps two through four, ought to be iteratively replicated until the appropriate quantity of layers has been attained. In the event that a validation measure is utilised, it is advisable to persist with the iterative process until the addition of another layer ceases to result in an enhancement of the validation statistics.

The probability estimates for the decision tree and bootstrap forest approaches are computed using the prob statistic [17]. Equation (1) used to compute the probability for the i^{th} response level at a specific node is defined as follows:

$$Prob_i = \frac{n_i + prior_i}{\sum(n_i + prior_i)} \quad (1)$$

where the summation is taken over all response levels; the variable n_i represents the number of observations at the node for the i^{th} response level; and the variable $prior_i$ denotes the prior probability for the i^{th} response level, which is derived using the following Equation (2):

$$prior_i = \lambda p_i + (1 - \lambda) P_i \quad (2)$$

where p_i is the $prior_i$ from the parent node, P_i represents the $Prob_i$ from the parent node, and λ is a weighting factor that is assigned as a fixed value so as to ensure that the predicted probabilities are not equal to zero.

The comprehensive forecast is derived by consolidating the individual projections for a certain observation across all levels. The model's overall fit can be improved by progressively incorporating layers and fitting them to the residuals derived from the preceding layers [18]. The existing configuration solely supports binary responses within category variables. The offsets of linear logits are used to indicate the residuals fit at each layer for a categorical answer. The final prediction is derived by employing a logistic transformation on the collective sum of the linear logits across all layers.

2.2. Hyperbolic Tangent Neural Network

Neural networks can be defined as a mathematical function that manipulates a set of derived inputs, commonly known as hidden nodes. The concealed nodes can be represented as non-linear transformations of the initial inputs. The researcher is provided with the choice to define how many hidden layers they prefer, where each layer can be customised to include any desired quantity of hidden nodes. The neural network architecture encompasses a fully connected multi-layer perceptron, consisting of either one or multiple layers. Neural networks are utilised in order to predict one or more response variables by employing a flexible function of the input variables. Neural networks demonstrate robust predictive ability in scenarios where there is no need to explicitly define the functional structure of the response surface or establish a clear description of the relationship between the input variables and the response variable [19]. Figure 2 displays the proposed neural network design that showcases 45 inputs derived from the dataset, with the only output being the prediction result for RUL.

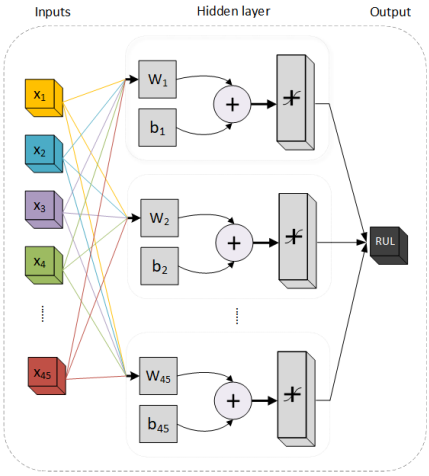


Figure 2. Display of a neural network diagram with 45 inputs.

The activation function is a mathematical operation that applies a non-linear transformation to the weighted sum of the input variables, sometimes denoted as x variables. The function used for the response entails either a linear combination in the case of continuous responses or a logistic transformation for nominal or ordinal responses [20]. One significant advantage of utilising a neural network model is its capacity to efficiently record and depict a wide range of response surfaces [21]. By employing an ample quantity of hidden nodes and layers, it becomes feasible to approximate any given surface with a high degree of accuracy. One significant constraint linked to neural network models is the difficulty of comprehending their results. The behaviour in question can be attributed to

the inclusion of intermediate layers, which differ from the typical regression model where a straight link exists from the x variables to the y variables.

2.3. Hidden Layer Structure

The neural network possesses the capacity to adapt to either a single or multiple layer configuration. The addition of nodes in the initial layer, or the inclusion of an extra layer, improves the flexibility of the neural network. It is feasible to integrate an unlimited number of nodes into either layer. The nodes in the second layer exhibit a dependency on the x variables. The operational capabilities of the initial layer nodes are contingent upon the second layer nodes. The dependent variables, also known as y variables, are defined by the functions performed by the nodes in the first layer.

This research paper examines the process of function approximation through the use of feedforward artificial neural networks, specifically focusing on the restriction of connections just between adjacent layers. In the subsequent section, we present a formal exposition of the notion of a neural network and the associated nomenclature.

Let L be an element of the set of natural numbers, and let $l_0, \ldots, l_L \in \mathbb{N}$ also be elements of the set of $\sigma: \mathbb{R} \longrightarrow \mathbb{R}$ be an activation function and define the parameter space in Equation (3) [22]:

$$\Theta = \bigcup_{L \in \mathbb{N}} \bigcup_{l_0, \ldots, l_L \in \mathbb{N}} \bigtimes_{k=1}^{L} \left(\mathbb{R}^{l_k \times l_{k-1}} \times \mathbb{R}^{l_k} \right) \tag{3}$$

for any $\theta \in \Theta$, it is defined $\theta_k \coloneqq (W_k, b_k)$, where W_k and b_k are matrices. Additionally, it is defined A_k as a function that maps from $\mathbb{R}^{l_{k-1}} \longrightarrow \mathbb{R}^{l_k}: x \mapsto W_k x + b_k: x \mapsto W_k x + b_k$, where x is the input and the output is given by A_k. The formula is defined for all values of k such that $1 \leq k \leq L$. Furthermore, it denotes Ψ_θ as a function that maps from $\mathbb{R}^{l_0} \to \mathbb{R}^{l_L}$, $x \mapsto \Psi_\theta(x)$ in Equation (4):

$$\psi_\theta(x) = \begin{cases} A_1 x \\ (A_L \circ \sigma \circ A_{L-1} \circ \sigma \circ \cdots \sigma \circ A_1)(x) \end{cases} \tag{4}$$

where σ is applied elementwise. The term Ψ_θ is used to denote the instantiation of the neural network linked to the parameter θ, which consists of L layers and has widths $(l_0, l_1, \ldots, l_L)$. The initial $L-1$ layers are commonly denoted as hidden layers. For each integer k such that $1 \leq k \leq L$, it is defined that the width of the layer k as l_k. Additionally, the weights and biases associated with layer k are denoted as W_k and b_k, respectively. The width of Ψ_θ is formally defined as the value among the set of lengths $\max(l_0, l_1, \ldots, l_L)$. When the value of $L = 2$, it is appropriate to refer to Ψ_θ as a deep neural network. Therefore, it can be observed that a shallow neural network consists of a single hidden layer, while deep neural networks are characterised by the presence of two or more hidden layers. This paper will primarily examine neural networks that employ the hyperbolic tangent as an activation function. The hyperbolic tangent function, denoted as $\sigma(x)$, is defined as

$$\sigma(x) \coloneqq \tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \tag{5}$$

for $x \in \mathbb{R}$. The networks shall be denoted as tanh neural networks. While it is possible to apply concepts to various smooth activation functions, concentrating on a specific activation function enables the establishment of precise and explicit bounds without compromising the clarity of the arguments [23]. It is worth mentioning that the findings are specifically applicable to the sigmoid or logistic activation function, which may be understood as a transformed version of the hyperbolic tangent function with shifts and scaling. This concludes by revisiting the fundamental features of neural network calculus that will be employed throughout the subsequent discussion, albeit without explicit mention.

It is noteworthy to state that in certain circumstances, there are two often employed activation functions, namely the linear and Gaussian functions, which typically exhibit

linearity and Gaussian distribution characteristics [24]. The term linear pertains to a mathematical concept that characterises a correlation, or the notion of the identity function. The process of transforming the linear combination of the x variables is not executed. The utilisation of the linear activation function is frequently observed in conjunction with one of the non-linear activation functions. In this particular instance, the linear activation function is situated within the second layer, whereas the non-linear activation functions are situated within the first layer. This methodology demonstrates its efficacy when the aim is to first reduce the dimensionality of the x variables and subsequently utilise a nonlinear model for the y variables [25]. When contemplating a continuous dependent variable y, if the model exclusively utilises linear activation functions, the resultant model for y can be mathematically represented as a linear amalgamation of the independent variables x. When confronted with a dependent variable y that is either nominal or ordinal in nature, it is possible to simplify the model by employing logistic regression.

The term Gaussian pertains to a mathematical distribution that is named in honour of Carl Friedrich Gauss [26]. The Gaussian function, often known as the normal distribution, is a mathematical function that finds widespread application in many academic disciplines such as mathematics, physics, and statistics. This alternative should be employed when there is a wish to observe the behaviour of the radial basis function or when the response surface exhibits a Gaussian (normal) form [27]. The Gaussian function is commonly represented as e^{-x^2}, where x is a linear combination of the X variables.

2.4. Prognostic Problem

The degradation pattern consists of multivariate time-series data obtained from condition monitoring sensors $X_{s_i} = [x_{s_i}^{(1)}, \ldots, x_{s_i}^{(m_i)}]^T$, along with the accompanying RUL values $Y_i = [y_i^1, \ldots, y_i^{m_i}]^T$. These measurements were collected from a fleet of N units, denoted as $(i = 1, \ldots, N)$. Every observation $x_{s_i}^{(t)} \in R^p$ consists of a vector containing p raw measurements that were taken under specific operating conditions $\omega_i^{(t)} \in R^s$. The duration of the sensory signal for the i^{th} unit is denoted as m_i, which may vary among different units. The cumulative length of the dataset in question is $m = \sum_{i=1}^{N} m_i$. In a more concise manner, the available dataset is denoted as $D = \{W_i, X_{s_i}, Y_i\}_{i=1}^{N}$. Considering the provided scenario, the objective is to develop a predictive model $\mathcal{G}$ that can accurately estimate the RUL $(\hat{Y})$ on a test dataset consisting of M units $D_{T*} = \{X_{sj*}\}_{j=1}^{M}$. These units are represented by multivariate time-series data $X_{sj*} = [x_{sj*}^1, \ldots, x_{sj*}^{k_j}]$, which captures readings from various sensors [28]. The cumulative length of the test data is $m_* = \sum_{j=1}^{M} k_j$.

3. Prediction Dataset and Boosting Process

The NASA N-CMAPSS dataset offers simulated degradation trajectories of a fleet of turbofan engines, which are designed to mimic run-to-failure scenarios under real flight conditions. The baseline health status of these engines are unknown. The dataset comprises eight distinct sets of simulated data collected from a total of 128 individual engines. These units are subject to five distinct failure modes, all of which have the potential to impact the flow and efficiency of the rotating sub-components. Table 1 presents a comprehensive summary of flight classes and their corresponding failure modes, as shown by the available datasets [28].

Table 1. Overview of the aircraft turbine dataset.

Number	Engine	Flight Class	Failure Quantity	Fan		LPC		HPC		HPT		LPT	
				E	F	E	F	E	F	E	F	E	F
DS01	10	1,2,3	1							Y			
DS02	9	1,2,3	2							Y		Y	Y
DS03	15	1,2,3	1							Y		Y	
DS04	10	2,3	1	Y	Y								
DS05	10	1,2,3	1					Y	Y				
DS06	10	1,2,3	1			Y	Y	Y	Y				
DS07	10	1,2,3	1									Y	Y
DS08	54	1,2,3	1	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Figure 3 depicts the intricate connections between various important operating factors, such as altitude (alt), throttle-resolver angle (TRA), flight cycles, total temperature, and the relative temperature decrease at the fan inlet (T2). This figure illustrates the interplay and impact of these variables, presenting data obtained from a dataset comprising six aircraft engines operating under different flight conditions. This representation emphasises the complex and interconnected aspects of our analysis and the detailed relationships that our model takes into account during the predictive process.

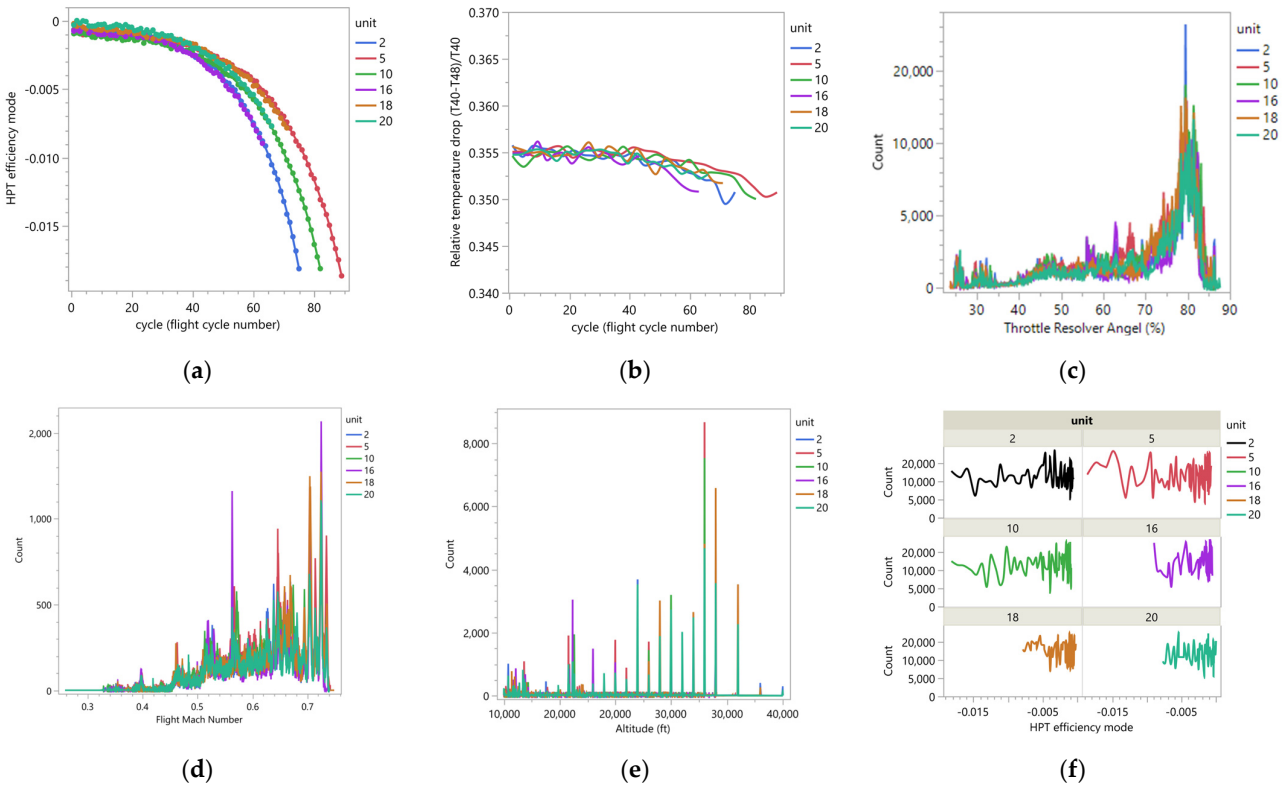


Figure 3. Interconnected relationships between scenario descriptors such as Mach number, throttle resolver angle (TRA), altitude (alt), and operative parameters such as fight cycles, HPT efficiency, total temperature, and its relative temperature drop. (**a**) Cycle versus HPT efficiency. (**b**) Cycle versus relative temperature drop. (**c**) Throttle resolver angel time series variations. (**d**) Engine Mach number variations. (**e**) Flight altitude changes. (**f**) Engines HPT efficiency variation against individual engine units.

The dataset used for this research is referred to as DS01 and DS02, which depict simulated examples of typical flight cycles, as evidenced by the traces of the scenario-

descriptor variables. Each simulated flight cycle consists of recordings of varying durations that record the climb, cruise, and descent phases of flight when the aircraft's altitude reaches 10,000 feet. These recordings correspond to the numerous flight paths undertaken by the aircraft. The flight trajectories of the remaining units of the fleet exhibit similarities. Both the training dataset and the validation dataset are contained within each data file. The dataset comprises six distinct types of variables: the operative conditions ω, the measured signals x_s, the virtual sensors x_v, the engine health parameters θ, the RUL label, and the auxiliary data such as unit number u, flight class number Fc, flight cycle number c, and the binary health state. Additionally, the parameters ω, x_s, x_v, and θ contain specified variable names. A summary of numerous variables can be found in [28].

The bootstrap forest encompasses the concepts of in-bag (IB) and out-of-bag (OOB) observations. The training set observations used to form a tree are referred to as IB observations. Observations in the training set that are not used in constructing a tree are referred to as OOB observations. The model offers the RASE values, which are the mean values for both IB and OOB observations, calculated over all trees. The OOB RASE is calculated for each tree by taking the square root of the sum of squared errors divided by the number of OOB observations. The data used to train an individual tree in the bootstrap method is randomly selected with replacement. In cases where the sampling procedure is intended to encompass all observations, the incorporation of replacement in the sampling approach leads to an expected proportion of unused observations equivalent to the reciprocal of the mathematical constant e. Every tree consists of over 1000 splits that include rankings, losses, and observations associated with OOB and IB data. The tree that exhibits the lowest OOB loss is assigned Rank 1. Table 2 presents the first 21 per-tree summaries of the bootstrap forest training processes.

Table 2. The description of the initial 21 branch trees in the bootstrap forest training operations.

Tree	Splits	Rank	OOB Loss	OOB Loss/N	RSquare	IB SSE	IB SSE/N	OOB N	OOB SSE	OOB SSE/N
1	1069	19	348,428.61	0.179873	0.9996	588,249.73	0.1117613	1,937,081	345,175.5	0.1781936
2	1079	75	1,129,529.9	0.5835069	0.9988	1,942,589.7	0.3690718	1,935,761	1,125,875.5	0.5816191
3	1137	48	657,177.05	0.3395718	0.9993	1,118,063.7	0.2124204	1,935,311	653,461.42	0.3376519
4	1239	82	1,428,064.1	0.737793	0.9985	2,447,363.9	0.4649736	1,935,589	1,423,022.7	0.7351885
5	1037	9	301,566.33	0.1557748	0.9997	508,044.11	0.0965231	1,935,912	297,525.88	0.1536877
6	1139	58	773,725.04	0.3993993	0.9992	1,328,241.4	0.252352	1,937,222	769,885.82	0.3974174
7	1313	97	2,657,672	1.3728297	0.9973	4,515,957.2	0.8579847	1,935,908	2,651,303.7	1.3695401
8	1089	16	328,339.82	0.169505	0.9997	555,723.73	0.1055817	1,937,051	323,897.8	0.1672118
9	1147	78	1,180,913.6	0.6095186	0.9988	2,026,344.8	0.3849844	1,937,453	1,176,315.3	0.6071452
10	1017	18	347,915.85	0.1795507	0.9996	587,667.48	0.1116507	1,937,703	345,131.72	0.1781138
11	1301	96	2,520,584.1	1.3021806	0.9974	4,315,572.9	0.8199138	1,935,664	2514,463	1.2990183
12	1121	45	621,068.46	0.3207995	0.9994	1,047,427.3	0.1990002	1,936,002	616,356.03	0.3183654
13	1053	14	320,291.75	0.1654456	0.9997	548,369.07	0.1041844	1,935,934	316,321.09	0.1633946
14	1197	50	690,715.86	0.3568014	0.9993	1,185,768.5	0.2252836	1,935,855	686,576.1	0.354663
15	1041	17	331,125.41	0.1710198	0.9997	556,257.31	0.1056831	1,936,182	326,746.89	0.1687584
16	1079	38	527,248.26	0.2724227	0.9995	900,814.16	0.1711453	1,935,405	524,716.96	0.2711148
17	1121	71	947,161.31	0.4894759	0.9990	1,630,965.7	0.3098665	1,935,052	942,616.88	0.4871274
18	1095	31	463,948.05	0.239569	0.9995	786,811.25	0.1494859	1,936,595	460,580.81	0.2378302
19	1053	3	212,820.83	0.1099359	0.9998	351,663.75	0.0668124	1,935,863	209,515.7	0.1082286
20	1053	7	282,493.4	0.1458563	0.9997	486,220.64	0.0923768	1,936,793	279,462.69	0.1442915
21	1077	11	313,612.15	0.1619483	0.9997	533,892.63	0.101434	1,936,495	310,544.71	0.1603643

OOB loss is a metric that quantifies the overall predicted inaccuracy of the decision tree when it is applied to the rows that were not used for training. Smaller numbers imply greater prediction accuracy. Among the 21 trees, tree 19 has shown the highest prediction accuracy. Therefore, we prioritise and choose tree 19 and its splits to be included in the fusion with the suggested neural networks. In the context of a continuous response, the predicted value for a specific observation is obtained by computing the average of its predicted values across the ensemble of individual trees. Within a categorical framework, the projected likelihood of an observation is determined by taking the average of its expected

probabilities across the collection of individual trees. The categorisation of the observation is determined by assigning it to the level that corresponds to its highest projected likelihood. The cumulative validation is a graphical representation of the fit statistics for the validation set, illustrating their fluctuations in response to the number of trees. Accessibility is contingent upon the successful deployment of validation. The R-Square statistic serves as the sole measure of adequacy when evaluating a continuous response variable.

4. Discussion and Results

The methodology employed in generating the dataset involves the identification and analysis of the failure modes associated with the sub-components of the main engine. Figure 4 depicts a comprehensive methodological profiler for predicting the RUL of engines, which clearly showcases the positive, neural, and negative properties and profilers that influence the entire RUL prediction process.

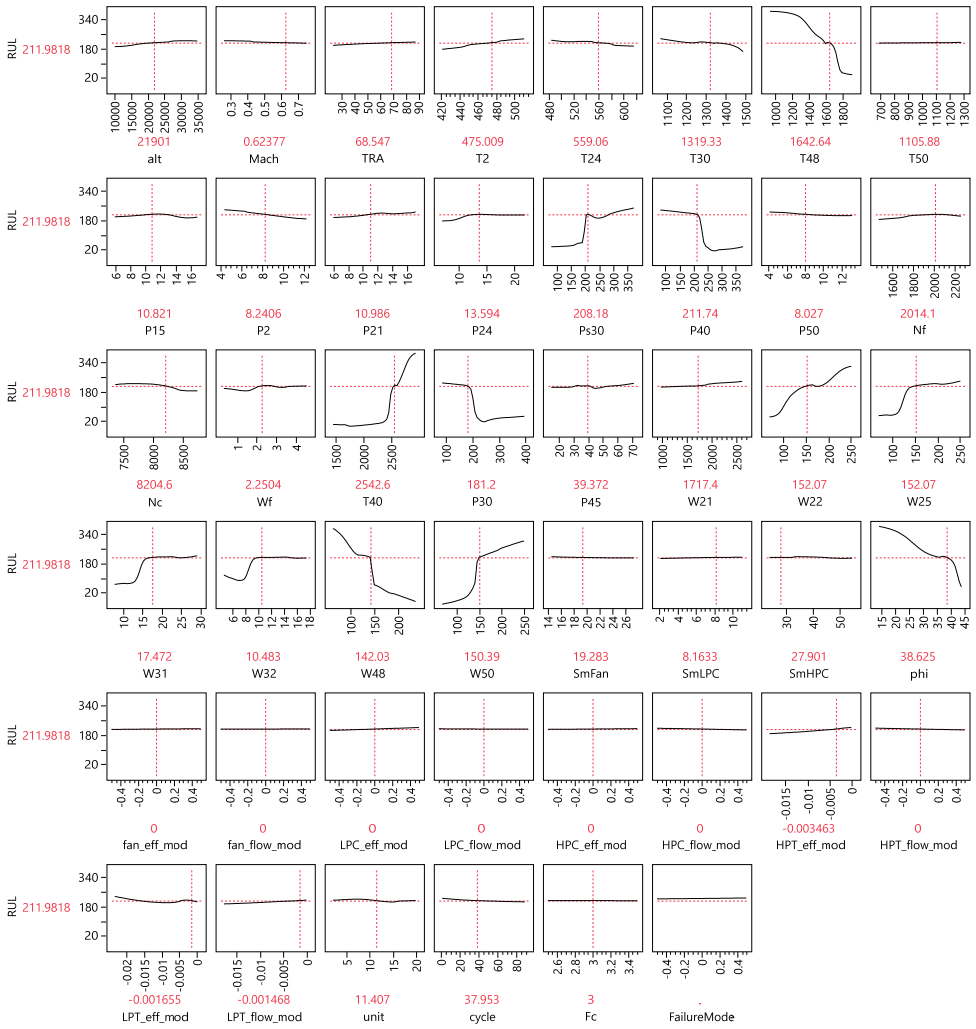


Figure 4. Depiction of the visual representation of prediction profiler traces that involve numerous inputs.

The components that exhibit a consistent decrease are the fan, low pressure compressor (LPC), high pressure compressor (HPC), low pressure turbine (LPT), and high-pressure turbine (HPT). The simulation of deterioration effects involves the modification of the flow capacity and efficiency of the engine sub-components, which are denoted by the engine health parameters θ. The given quantity of hyperbolic tangent boosted nodes is three, while a sequence of models that are additive in nature and scaled by the boosting learning rate is set to 0.1. The penalty method is mathematically defined as the square of a certain value, where the number of iterations is fixed at one. Figure 5 presents visual depictions of the highest rated response contours for pairs of variables, obtained by extracting notable positive and negative features through profiler manipulation. The interactive contour profiling capability is used to visually improve response surfaces. Every contour is accompanied by a dashed line that shows the direction of higher response values, giving a distinct indicator of orientation.

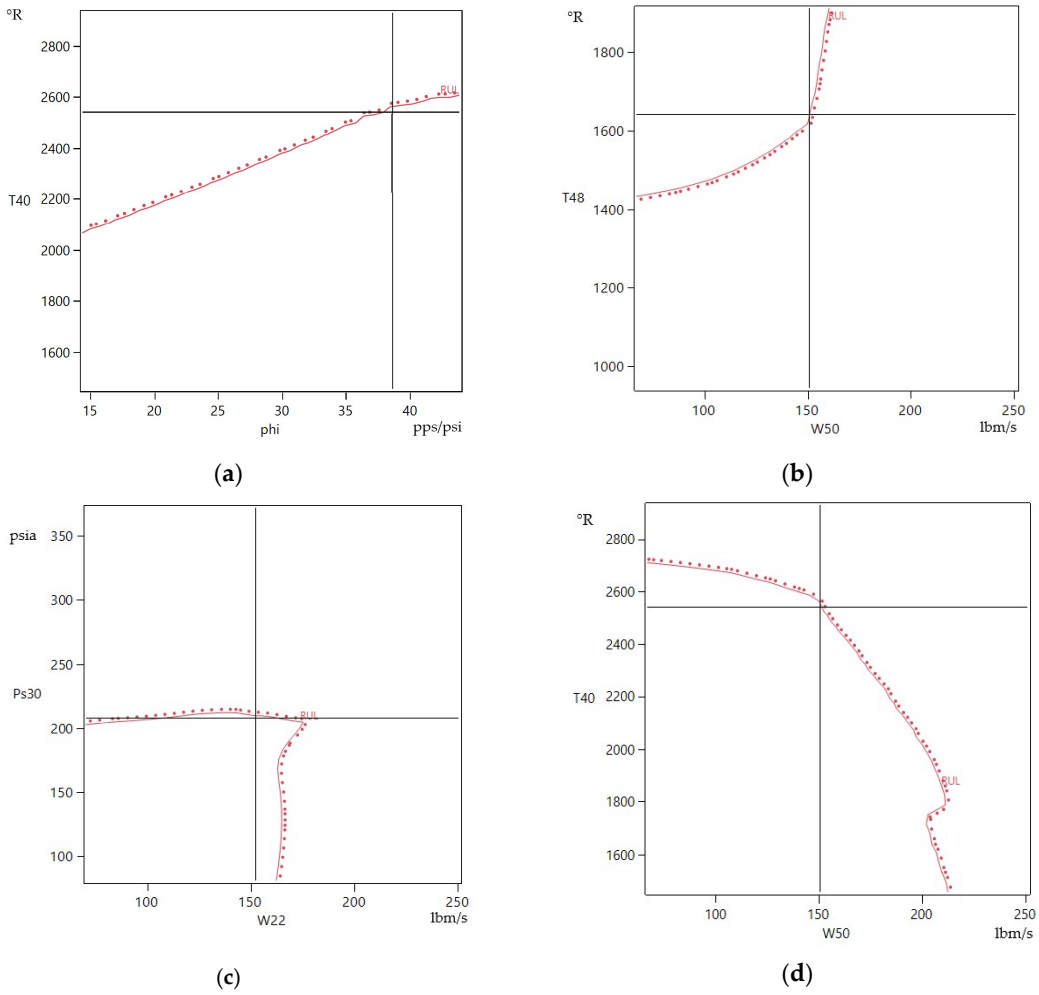


Figure 5. Representation of the most highly rated extracted features used for predicting RUL. (**a**) Interrelated profiler relationship between ratio of fuel flow to Ps30 (phi) and total temperature at burner outlet (T40); (**b**) Interrelated profiler relationship between flow out of LPT (W50) and total temperature at HPT outlet (T48); (**c**) Interrelated profiler relationship between flow out of LPC (W22) and static pressure at HPC outlet (Ps30); (**d**) Interrelated profiler relationship between W50 and T40.

The prediction profiler displays profile traces for every x variable. A profile trace refers to the anticipated outcome when a single variable is altered while keeping all other variables at their present levels. The prediction profiler recalculates the profiles and projected reactions instantaneously when you modify the value of an x variable. The vertical

dashed line for each x variable represents its current value or configuration. The value shown above each factor name is the current value of the corresponding x variable. The horizontal dashed line represents the present forecasted value of each y variable based on the current values of the x variables. To fully understand how the x factors are affecting the expected values, some variables show profiles with positive slopes, while others display negative slopes. For instance, the variable T40 has a favourable upward trend. This implies a clear relationship between the total temperature at the burner output T40 and the anticipated median value. Specifically, as the total temperature rises, the predicted median value similarly increases. T48 is the total temperature at the outlet of the HPT. It has a negative slope, indicating that as the temperature rises, the median value declines.

The fitting procedure of the neural network incorporates a validation technique. Various validation methods can be utilised, such as holdback, K-fold, and the incorporation of a validation column. To facilitate the development of a model, a series of steps are taken. The model parameters are subjected to a penalty, and the penalties imposed on these parameters are optimised using the validation set. The validation method is chosen as a random holdback, which means that the original data is divided into training and validation sets in a random manner, and the proportion of the original dataset to be allocated as the validation set, often known as the holdback. The process of random selection in this study is grounded in the principles of stratified sampling, which involves dividing the model factors into distinct strata. This approach aims to provide training and validation sets that exhibit a higher degree of balance compared to those obtained through simple random sampling. Figure 6 displays plots that compare the actual values with the predicted values for the training and validation procedures. The plots use the bootstrap forest method shown in red, the boosted tree method represented in green, and the neural boosted method represented in blue. The data points exhibit a strong correlation with the line, suggesting a high degree of relationship between the anticipated values and the actual values.

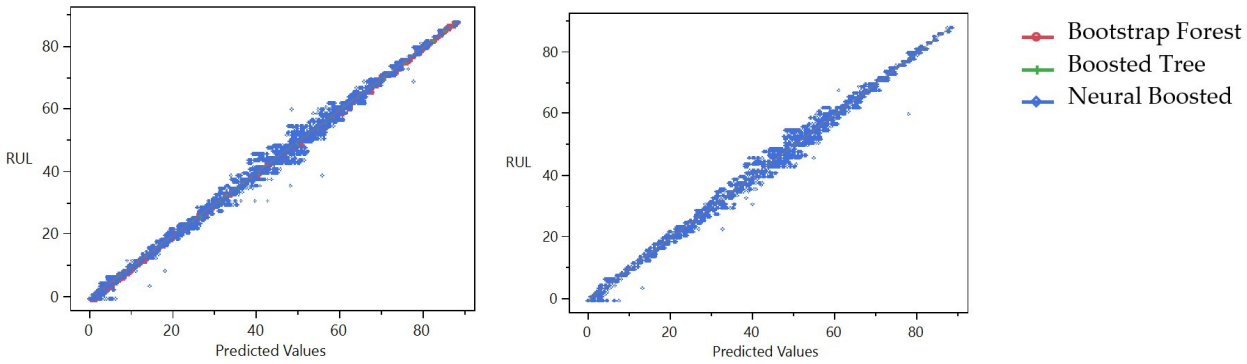


Figure 6. Illustration of the prediction outcomes and their accompanying metrics, which include the bootstrap forest, boosted tree, and neural boosted network. (**left**) Process of training. (**right**) The validation process.

Boosting refers to the procedure of constructing a substantial neural network model by training a series of smaller models. Each of the smaller models is fitted using the scaled residuals of the preceding model. The models are aggregated to create the overarching final model. The technique employs validation to determine the appropriate number of component models to fit, ensuring that it does not exceed the set limit. Boosting often exhibits faster computational speed compared to training a single, big model. Nevertheless, the fundamental model needs to consist of a single layer with 1 to 2 nodes. The advantage of quicker fitting might be negated if a substantial quantity of models is supplied. Learning rates approaching 1 lead to quicker convergence on a final model but also have a greater inclination to overfit data. In order to address the issue of neural networks being

prone to overfitting, the fitting procedure includes a penalty on the probability. The penalty term is represented as $\lambda p(\beta i)$, where λ represents the regularisation parameter, and $p(\,)$ denotes the penalty function, which is a function of the estimated parameters. The utilisation of the validation method is employed in order to ascertain the optimal value for the penalty parameter. Table 3 provides an overview of several penalty techniques.

Table 3. Overview of several penalty mechanisms used for differentiation.

Method	Penalty Function	Description
Squared	$\sum \beta_i^2$	This approach is recommended when a significant number of x factors are believed to be influential in enhancing the predictive capacity of the model.
Absolute	$\sum \lvert\beta_i\rvert$	If one is faced with a substantial number of x variables and suspects that only a subset of them significantly contribute to the predictive capacity of the model, either of the following approaches can be employed.
Weight Decay	$\sum \frac{\beta_i^2}{1+\beta_i^2}$	
No Penalty		Does not employ a penalty. This option is suitable for situations where there is a substantial volume of data and expeditious completion of the fitting procedure is desired. Nevertheless, this alternative may result in models exhibiting inferior predictive accuracy compared to models that incorporate a penalty.

The authors of this research article conduct a comparative analysis between the proposed methodology and established prediction approaches, such as K Nearest Neighbours (kNNs) and boosted tree. The neural boosted model has outstanding performance, as shown by its consistently high training RSquare values throughout several folds and its higher validation RASE score. The mean validation RSquare for neural boosted is 0.9968, and the mean RASE for neural boosted is 1.2711. The elapsed time data indicates that the neural boosted models exhibited the longest duration for fitting. Table 4 displays the comprehensive comparison of forecast outcomes in relation to various methods.

Table 4. A compilation of the outcomes obtained from comparing predictions.

(a) Training Process				(b) Validation Process			
Method	**N**	**RSquare**	**RASE**	**Method**	**N**	**RSquare**	**RASE**
K Nearest Neighbours	5,263,447	1.0000	0.1386	Neural Boosted	1754307	0.9968	1.2711
Bootstrap Forest	5,263,447	1.0000	0.1468	Bootstrap Forest	-	-	-
Boosted Tree	5,263,447	0.9999	0.1603	Boosted Tree	-	-	-
Neural Boosted	3,509,140	0.9968	1.2688	K Nearest Neighbours	-	-	-

Each of the training, validation, and test sets is provided with a range of prognostic measures. The prognostic measures vary based on whether the response is categorical or continuous. For continuous response scenarios, the cross-validation statistics consist of a generalised R^2, an entropy R^2, the (negative) loglikelihood, the root mean squared error (RMSE), and the mean absolute deviation. A list of measures of fit for both training and validation processes is summarised in Table 5 as follows.

Table 5. Training and validation metrics of fit.

Measures	Description
Generalised R^2	Generalised R^2 extends the concept of the coefficient of determination (R^2) to models that do not fit within the framework of linear regression. The adjusted coefficient of determination quantifies the percentage of variation accounted for by the model while taking into account the number of predictors. It is a useful metric for assessing the goodness of fit between different models.
Entropy R^2	It is a statistical measure commonly employed in the analysis of categorical data. The concept of entropy from information theory serves as the foundation for this assessment, which measures the decrease in uncertainty resulting from the model. Higher numbers suggest a model that effectively explains the variability in the categories.
R^2	Gives the RSquare for the model.
RASE	Gives the root average squared error. When the response is nominal or ordinal, the differences are between 1 and p, where p is the predicted probability of the response level.
Mean Abs Dev (MAD)	It quantifies the average absolute difference between expected and observed values, irrespective of their direction. It is a reliable indicator of the extent to which these faults are distributed.
Misclassification rate (MR)	The misclassification rate (MR) is a metric employed in classification problems to quantify the proportion of inaccurate predictions out of the total number of predictions made. It is a simplistic approach to assessing the frequency of the model's inaccuracies.
-Loglikelihood	Provides the negative of the loglikelihood
SSE	Gives the error sums of squares. It is only available when the response is continuous.
Sum Freq	It represents the complete analysis of observations. When a neural network variable Freq is supplied, the Sum Freq function sums the frequency column values.

Table 6 displays the results of the suggested NtahH(3)NBoost(20) measure of fit for both the training and validation phases. The R^2 values are remarkably high for both the training (0.9967847) and validation (0.9967703) datasets, suggesting that the NtanH(3)NBoost(20) model accounts for almost all of the variability in the target variable. The marginal decline in the validation R^2 indicates a remarkably high degree of model precision and ability to generalise. The RASE scores for both training (1.2688147) and validation (1.2710539) are comparably low, indicating that the model's predictions closely match the actual data points. The congruity between the RASE values of the training and validation sets suggests consistent and reliable performance across various datasets. The SSE represents the sum of the squared differences between the predicted values and the actual values. The SSE for the training data is 5649332.2, and the SSE for the validation data is 2834219.7. The correctness of the model is supported by these figures, as well as the strong R^2 values.

Table 6. Prediction results for NtanH(3)NBoost(20) on both training and validation datasets.

(a) Training Process		(b) Validation Process	
Measures	**Value**	**Measures**	**Value**
R^2	0.9967847	R^2	0.9967703
RASE	1.2688147	RASE	1.2710539
MAD	0.8863745	MAD	0.8882592
-LogLikelihood	5814721.2	-LogLikelihood	2910018
SSE	5649332.2	SSE	2834219.7
Sum Freq	3509140	Sum Freq	1754307

The congruity in RASE and MAD values between the training and validation stages implies that the model exhibits strong generalisation capabilities towards unfamiliar input. This is a crucial element of prognostic models, as it indicates their dependability in practical situations where the model needs to generate precise forecasts on data that it has

not been exposed to during its training. Due to the low SSE, the model's predictions closely align with the actual observations, providing more evidence of its efficiency.

To determine the effectiveness and strength of the suggested methodology, a comparison analysis was performed against existing methodologies described in the literature as follows:

- A method was presented by [5] that uses data fusion with stage division to forecast the RUL of a system. This method, called an ensemble RUL prediction method, has been shown to achieve considerable gains in accuracy. The utilisation of bootstrap forests and hyperbolic tangent neural networks in our approach yields similar results while providing increased adaptability and resilience in managing nonlinear deterioration patterns.
- Ref. [2] conducted a comprehensive analysis of predictive maintenance for defence fixed-wing aircraft, emphasising the importance of flexible and precise prognostic models. The hybrid model we have developed successfully integrates numerous prediction algorithms to ensure both high accuracy and reliability.
- Ref. [29] criticised the reliability prediction techniques used in avionics applications, highlighting the constraints of conventional methods in dynamic and intricate settings. The suggested neural boosted model overcomes these constraints by providing a prediction framework that is more dynamic and adaptive.
- The study conducted by [30] investigated the use of multi-task learning to forecast the RUL in different working settings. The results showed that advanced learning approaches have the potential to enhance the accuracy of RUL predictions. The findings of our study demonstrate that the combination of bootstrap forests and neural networks can produce comparable or better performance, especially when dealing with diverse operational scenarios.
- The significance of feature fusion and similarity-based prediction models for precise estimation of RUL was highlighted by [1]. Our suggested model improves upon current approaches by using hyperbolic tangent functions and bootstrap aggregation, resulting in increased prediction accuracy.

Table 7 demonstrates the enhanced performance of the suggested methodology compared to established models through statistical comparisons.

Table 7. Statistical comparison between the current model and the existing ones.

	R^2	RASE	SSE
NtanH(3)NBoost(20)	Training: 0.9967847 Validation: 0.9967703	Training: 1.2688147 Validation: 1.2710539	Training: 5649332.2 Validation: 2834219.7
[1]	Average R^2: 0.994	-	-
[30]	Average R^2: 0.995	-	-
[2]	-	Average: 1.35	-
[3]	-	Average: 1.45	-
[6]	-	-	Average: 6,000,000

In summary, the NtanH(3)NBoost(20) model shows outstanding effectiveness in predicting aircraft systems, as supported by strong performance indicators. This indicates that the model possesses the capacity to significantly enhance predictive maintenance protocols. Further research might focus on comprehending the -LogLikelihood values within a broader context, exploring the implications of Sum Freq, and comparing the effectiveness of the model with existing prognostic methods to strengthen its practical importance in the field.

5. Conclusions

This paper presents a novel data-driven prognostic algorithm designed exclusively for the aviation industry, with an emphasis on monitoring and managing the health of gas turbine engines. It demonstrates a significant enhancement in predictive accuracy compared to previous methods by using the NASA N-CMAPSS dataset, which provides simulated degradation trajectories. Furthermore, this paper closely simulates the operational conditions experienced during commercial aircraft flights. The approach effectively connects the engine's deterioration process with its operational history, allowing for a detailed comprehension of the different ways it can fail.

The investigation into the physical methodology of turbofan engines uncovers a potential for more enhancement, indicating that although our existing model achieves significant accuracy, including more precise physics-based models could lead to improvements. The proposed model represents a notable departure from conventional models that rely on engines with lifespans of hundreds of cycles. It incorporates a sophisticated combination of bootstrap forest and hyperbolic tangent boosted neural networks and has been developed and researched extensively.

Overall, the NtanH(3)NBoost(20) model represents a significant advancement in the precision and dependability of data-based predictions for aircraft systems. The models demonstrated a high level of accuracy, correctly predicting observed flight cycles during validation with an accuracy rate of 95–97%. Additionally, it paves the way for future improvements and developments. The encouraging outcomes achieved prompt continuous improvement of both the model itself and the fundamental physical techniques, with the goal of developing even more accurate and hybrid prognostic models in the future. This is to strengthen predictive maintenance procedures, which will ultimately lead to safer, more efficient, and more dependable aircraft operations through ongoing improvement and adaptation.

Author Contributions: Conceptualisation, S.F.; methodology, S.F.; software, S.F.; validation, S.F.; formal analysis, S.F.; writing—original draft preparation, S.F.; writing—review and editing, S.F. and N.P.A.; supervision, N.P.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research has received funding from the European Commission under the Marie Skłodowska Curie program through the H2020 ETN MOIRA project (GA 955681).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are publicly available from the NASA prognostics data repository and mirrored on the Prognostics Health Management Society website.

Acknowledgments: The authors gratefully acknowledge the H2020-MSCA-ITN-2020 MOIRA project and the research team at IVHM Centre, Cranfield University, UK, for supporting this research.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Li, Y.; Wang, Z.; Li, F.; Li, Y.; Zhang, X.; Shi, H.; Dong, L.; Ren, W. An Ensembled Remaining Useful Life Prediction Method with Data Fusion and Stage Division. *Reliab. Eng. Syst. Saf.* **2024**, *242*, 109804. https://doi.org/10.1016/J.RESS.2023.109804.
2. Scott, M.J.; Verhagen, W.J.C.; Bieber, M.T.; Marzocca, P. A Systematic Literature Review of Predictive Maintenance for Defence Fixed-Wing Aircraft Sustainment and Operations. *Sensors* **2022**, *22*, 7070. https://doi.org/10.3390/s22187070.
3. Pandian, G.P.; Das, D.; Li, C.; Zio, E.; Pecht, M. A Critique of Reliability Prediction Techniques for Avionics Applications. *Chin. J. Aeronaut.* **2018**, *31*, 10–20. https://doi.org/10.1016/J.CJA.2017.11.004.
4. Zhang, Z.; Chen, X.; Zio, E.; Li, L. Multi-Task Learning Boosted Predictions of the Remaining Useful Life of Aero-Engines under Scenarios of Working-Condition Shift. *Reliab. Eng. Syst. Saf.* **2023**, *237*, 109350. https://doi.org/10.1016/J.RESS.2023.109350.

5. Fu, S.; Avdelidis, N.P. Prognostic and Health Management of Critical Aircraft Systems and Components: An Overview. *Sensors* **2023**, *23*, 8124. https://doi.org/10.3390/s23198124.
6. Li, X.; Teng, W.; Peng, D.; Ma, T.; Wu, X.; Liu, Y. Feature Fusion Model Based Health Indicator Construction and Self-Constraint State-Space Estimator for Remaining Useful Life Prediction of Bearings in Wind Turbines. *Reliab. Eng. Syst. Saf.* **2023**, *233*, 109124. https://doi.org/10.1016/J.RESS.2023.109124.
7. Lin, Y.H.; Ding, Z.Q.; Li, Y.F. Similarity Based Remaining Useful Life Prediction Based on Gaussian Process with Active Learning. *Reliab. Eng. Syst. Saf.* **2023**, *238*, 109461. https://doi.org/10.1016/J.RESS.2023.109461.
8. Ali, S.; Abuhmed, T.; El-Sappagh, S.; Muhammad, K.; Alonso-Moral, J.M.; Confalonieri, R.; Guidotti, R.; Del Ser, J.; Díaz-Rodríguez, N.; Herrera, F. Explainable Artificial Intelligence (XAI): What We Know and What Is Left to Attain Trustworthy Artificial Intelligence. *Inf. Fusion* **2023**, *99*, 101805. https://doi.org/10.1016/J.INFFUS.2023.101805.
9. Okoh, C.; Roy, R.; Mehnen, J. Predictive Maintenance Modelling for Through-Life Engineering Services. *Procedia CIRP* **2017**, *59*, 196–201. https://doi.org/10.1016/J.PROCIR.2016.09.033.
10. Chen, J.; Li, D.; Huang, R.; Chen, Z.; Li, W. Aero-Engine Remaining Useful Life Prediction Method with Self-Adaptive Multimodal Data Fusion and Cluster-Ensemble Transfer Regression. *Reliab. Eng. Syst. Saf.* **2023**, *234*, 109151. https://doi.org/10.1016/J.RESS.2023.109151.
11. Bootstrap Aggregating. Available online: https://en.wikipedia.org/w/index.php?title=Bootstrap_aggregating&oldid=1223807758 (accessed on 11 March 2024).
12. Aria, M.; Cuccurullo, C.; Gnasso, A. A Comparison among Interpretative Proposals for Random Forests. *Mach. Learn. Appl.* **2021**, *6*, 100094. https://doi.org/10.1016/J.MLWA.2021.100094.
13. Boosting (Machine Learning). Available online: https://en.wikipedia.org/w/index.php?title=Boosting_(machine_learning)&oldid=1222981999 (accessed on 11 March 2024).
14. Chou, Y.-C.; Chuang, H.H.-C.; Chou, P.; Oliva, R. Supervised Machine Learning for Theory Building and Testing: Opportunities in Operations Management. *J. Oper. Manag.* **2023**, *69*, 643–675. https://doi.org/10.1002/joom.1228.
15. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3147–3155.
16. Altman, N.; Krzywinski, M. Points of Significance: Ensemble Methods: Bagging and Random Forests. *Nat. Methods* **2017**, *14*, 933–934. https://doi.org/10.1038/nmeth.4438.
17. Zhao, C.; Wu, D.; Huang, J.; Yuan, Y.; Zhang, H.-T.; Peng, R.; Shi, Z. BoostTree and BoostForest for Ensemble Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 8110–8126. https://doi.org/10.1109/TPAMI.2022.3227370.
18. Lee, T.-H.; Ullah, A.; Wang, R. Bootstrap Aggregating and Random Forest. In *Macroeconomic Forecasting in the Era of Big Data*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 389–429.
19. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *J. Big Data* **2021**, *8*, 53. https://doi.org/10.1186/s40537-021-00444-8.
20. Ezzat, D.; Hassanien, A.E.; Darwish, A.; Yahia, M.; Ahmed, A.; Abdelghafar, S. Multi-Objective Hybrid Artificial Intelligence Approach for Fault Diagnosis of Aerospace Systems. *IEEE Access* **2021**, *9*, 41717–41730. https://doi.org/10.1109/ACCESS.2021.3064976.
21. Chong, D.J.S.; Chan, Y.J.; Arumugasamy, S.K.; Yazdi, S.K.; Lim, J.W. Optimisation and Performance Evaluation of Response Surface Methodology (RSM), Artificial Neural Network (ANN) and Adaptive Neuro-Fuzzy Inference System (ANFIS) in the Prediction of Biogas Production from Palm Oil Mill Effluent (POME). *Energy* **2023**, *266*, 126449. https://doi.org/10.1016/J.ENERGY.2022.126449.
22. De Ryck, T.; Lanthaler, S.; Mishra, S. On the Approximation of Functions by Tanh Neural Networks. *Neural Netw.* **2021**, *143*, 732–750. https://doi.org/10.1016/j.neunet.2021.08.015.
23. Radhakrishnan, A.; Belkin, M.; Uhler, C. Wide and Deep Neural Networks Achieve Consistency for Classification. *Proc. Natl. Acad. Sci. USA* **2023**, *120*, e2208779120. https://doi.org/10.1073/pnas.2208779120.
24. Dubey, S.R.; Singh, S.K.; Chaudhuri, B.B. Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark. *Neurocomputing* **2022**, *503*, 92–108. https://doi.org/10.1016/j.neucom.2022.06.111.
25. Jagtap, A.D.; Karniadakis, G.E. How Important Are Activation Functions in Regression and Classification? A Survey, Performance Comparison, and Future Directions. *arXiv* **2022**, arXiv:2209.02681.
26. List of Things Named after Carl Friedrich Gauss. Available online: https://en.wikipedia.org/w/index.php?title=List_of_things_named_after_Carl_Friedrich_Gauss&oldid=1223714204 (accessed on 22 March 2024).
27. Surucu, O.; Gadsden, S.A.; Yawney, J. Condition Monitoring Using Machine Learning: A Review of Theory, Applications, and Recent Advances. *Expert. Syst. Appl.* **2023**, *221*, 119738. https://doi.org/10.1016/J.ESWA.2023.119738.
28. Chao, M.A.; Kulkarni, C.; Goebel, K.; Fink, O. Aircraft Engine Run-to-Failure Dataset under Real Flight Conditions for Prognostics and Diagnostics. *Data* **2021**, *6*, 5. https://doi.org/10.3390/data6010005.

29. Tsui, K.L.; Chen, N.; Zhou, Q.; Hai, Y.; Wang, W. Prognostics and Health Management: A Review on Data Driven Approaches. *Math. Probl. Eng.* **2015**, *2015*, 793161. https://doi.org/10.1155/2015/793161.
30. Shi, J.; Yu, T.; Goebel, K.; Wu, D. Remaining Useful Life Prediction of Bearings Using Ensemble Learning: The Impact of Diversity in Base Learners and Features. *J. Comput. Inf. Sci. Eng.* **2020**, *21*, 021004. https://doi.org/10.1115/1.4048215.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.



https://dspace.org/

Novel prognostic methodology of bootstrap forest and hyperbolic tangent boosted neural network for aircraft system

Fu, Shuai; Avdelidis, Nicolas P.

2024-06-02
MDPI
https://doi.org/10.3390/app14125057
Fu S, Avdelidis NP. (2024) Novel prognostic methodology of bootstrap forest and hyperbolic tangent boosted neural network for aircraft system. Applied Sciences, Volume 14, Issue 12, June 2024, Article number 5057

Downloaded from a DSpace repository. https://dspace.org/