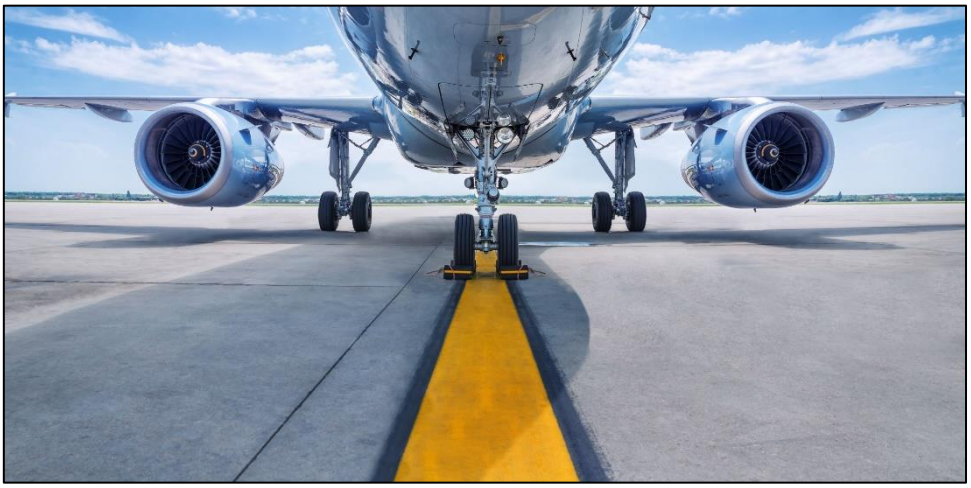
Machine Learning Prognostic for Aircraft Engine Under Real Flight Conditions



September 2022

CMI7420-2022

Carl Wilson [U0370630]

u0370630@unimail.hud.ac.uk

Contents

List of figures

List of tables

List of equations

Acknowledgements

The author would like to thank M. Chao, C. Kulkarni, K. Goebel, and O. Fink for providing such an interesting and comprehensive dataset, without which this work could not be completed. He would also like to thank G. Bargiannis for always making himself available for support and advice.

1. Introduction

Prognostics is defined, in the IEEE Standard Framework for Prognostics and Health Management [PHM] of Electronic Systems, as “the process of predicting an object system’s Remaining Useful Life [RUL] by predicting the progression of a fault given the current degree of degradation, the load history and the anticipated future operation” (IEEE, 2017).

The operational costs of aircraft are the most significant in its life-cycle, with maintenance costs being a significant contributor. These costs provide a very real challenge for airlines to have aircraft available and financially viable (Mofokeng et al., 2020).

Effective Prognostics and Health Management [PHM] systems can offer significant tangible benefits in cost avoidance whilst providing a positive Return On Investment [ROI] (Feldman et al., 2009), as well as increased safety, availability, reliability, and reduced waste material (IEEE, 2017). One of the crucial factors in a PHM system being effective is accuracy (Luna, 2021). Furthermore, aviation is a field where safety is paramount and thus an ideal candidate for PHM.

There have been ever increasingly clever and complex data-driven solutions that are quite pure to Machine Learning [ML], and more specifically - Deep Learning [DL] but offer little insight into the function map between input and output due to algorithms such as Artificial Neural Networks [ANN] and Support Vector Machines [SVM] being “black-box”. However, as will be presented in related works, there is a lack of explain ability and relation to first principles of mechanical engineering throughout many of the developed solutions.

The gap this study fills is making use of domain specific knowledge in the use of first principles of thermodynamics combined with cutting edge model explanation methods, to aid in the explain-ability and interpretability of the predictive models. Estimations will be made, with domain knowledge in mind, to determine a suitable subset of predictors and then benchmarked against using all available predictors. This will allow cutting edge techniques using algorithms with ever increasing levels of obfuscation to be explainable and interpretable within the context of the problem they solve.

2. Literature Review

2.1. Related Works

Approaches to prognostics and life estimations can be categorised into model-based approaches, those that rely on a mathematical and physics-based equations; and data-driven approaches, those that depend on a significant volume of data to create a function map of predictors and target labels.

Model-based approaches can be categorised as having a high degree of bias error and low degree of variance error; whereas the opposite is true for data-driven approaches. Where model-based approaches are heavily biased and struggle to capture all the complexities of low-level interactions – data-driven approaches specifically use these to build a picture. Thus, data-driven approaches have the challenge of generalisation and quality of predictions.

The NASA Prognostics Centre of Excellence [PCoE] published a Turbofan Engine Degradation Simulation Data Set (Saxena et al., 2008), known as CMAPSS, which has been extensively used in prognostics research for aircraft engines and features heavily in this background research. The data set was, however, limited in scope to several discrete operating modes per cycle and a snapshot of control parameters and measured sensor readings was provided by cycle and engine unit number. It also provided no insight into the starting condition of the engine, and at which point accelerated degradation starts to occur. A more recent dataset has been published (M. Chao, 2021), for the 2021 PHM Society Data Challenge (PHM, 2021), known as N-CMAPSS which addresses this shortcoming by actually simulating real flight mission profiles within each cycle and providing a distinction between normal and abnormal degradation.

Research into data-driven approaches has accelerated over the last decade or so with the advancement of algorithms and increasing computational power. A neural network that might have taken weeks to train previously can be completed over an order of magnitude faster (Schmidhuber, 2015) with modern graphics processing units [GPU]. A study by (Thoppil et al., 2021) reviewed different DL algorithms for machinery health prognostics using time series data. DL algorithms included Convolutional Neural Networks [CNN], Recurrent Neural Networks [RNN], Long Short-Term Memory networks [LSTM], Generative Adversarial Networks [GAN], and Auto Encoders [AE]. They were applied to simulated run to failure data, including mechanical bearings and the NASA CMAPSS datasets. The study found that the most popular algorithm applied to 1D time-series or 2D time-frequency signal was CNN. Similarly, the study found that the CMAPSS dataset was used for around one-third of the research investigated as part of the review.

An LSTM was developed by (Yuan et al., 2016) for aero-engines using the CMAPSS dataset, the results showed that it was acceptable to have a high error rate during the engines normal or healthy operation state, but during the abnormal degradation the RUL becomes critical and the resulting models accuracy increased gradually as the RUL decreased. The researchers used a Support Vector Machine [SVM] to determine at what point the fault occurred for each unit and then an LSTM to predict the remaining life. One assumption made by the researchers is that if multiple fault modes occur on the same engine, that they occur at the same time. It is unlikely this would be the case. A crucial element observed by (Li et al., 2018), when developing a CNN for the CMAPSS dataset, was the tuning of hyper-parameters in a network to manage overfitting and provide the most accurate but also highly generalising model. Hyper-parameters such as learning rate, optimiser selection, batch size, hidden layer number, number of neurons per layer as well as layer drop-out and activation functions all have a role to play.

A much more complex approach taken by (Kong et al., 2019) developed a hybrid DNN which combined CNN and LSTM to extract special and temporal features. They made use of the stall margins, and physical sensor measurements in the CMAPSS data to develop a 1D health-indicator [HI]. Then preselected data and HI were sequentially fed into the CNN layer and LSTM layer to extract high-level spatial features and long-term temporal dependency features. This research did make use of a variance threshold to remove features that were not obviously distinguishable so were not providing new information to the prognostic. The prognostic was benchmarked against four existing architectures, MLP, SVR, CNN and LSTM. They concluded that the hybrid model performed better on all scores including the NASA scoring function, with the hybrid approach being two orders of magnitude lower than the MLP and SVR but being slightly better than the CNN and LSTM models.

With the more recent N-CMAPSS dataset, the present research will make use of, the 2021 PHM challenge was won by (Lövberg, 2021) who developed a network of dilated convolutions with gated linear activations and residual skip connections. A dilated convolution is where the kernel's receptive field is extended by skipping input values of a given step size. An interesting approach taken here was to normalise the input data to the flight conditions, or scenario descriptors, *W.* This allowed the degradation signal to be identified from the noise of the flight conditions for each physical sensor. Another interesting approach taken here was to reduce the data frequency from 1Hz to 0.1Hz and to reduce the data fidelity from double precision to half-precision floating point values; thus, reducing the total memory and overhead to 7.5% and significantly reducing the computational overhead for faster model iteration.

The runner up in the same competition was (Nathaniel DeVol, 2021) who developed a CNN based on the inception architecture by (Szegedy et al., 2016). The inception architecture effectively has multiple CNN layers in parallel, of different sizes plus a single max pool layer which are all concatenated prior to the next inception layer. Due to the different sized CNN layers, different sized features can be identified, thus a shallower but wider network can be developed. The researcher concluded that a key limitation of the work was that the RUL estimations were not explainable, and the model was not able to explain why any specific RUL prediction was made. He also goes on to recommend that future work focus on the explain ability as well as using the model to isolate the failure component and failure mode, as the full N-CMAPSS dataset contains multiple datasets with different and combined failure modes.

A conclusion by (Thoppil et al., 2021) was: "The quality of acquired machine degradation data and the chosen deep learning algorithm are the major factors influencing the accuracy of the machinery health prognostics". To this end, it will be key to understand which parameters offer the most informational value to the prognostic, which should in turn help reduce mapping to noise and overfitting.

A study completed by (Khumprom et al., 2020) was motivated to understand how different dimensionality reduction techniques could simplify models and reduce the computational overhead, specifically on the CMAPSS dataset. A hyper-model was developed varying epoch, learning rate, momentum plus L1 and L2 regularisation. There were two types of feature selection employed: filtering and wrapper methods. The filtering methods included Pearson Correlation, Relief Algorithm, SVM, PCA and Deviation. While the wrapper techniques were forward selection, backward selection, and evolutionary selection. They concluded that the SVM and Relief Algorithm provided a feature selection, that did not develop a suitable prognostic. While the researchers also concluded that the evolutionary selection was the best overall performer, the Pearson Correlation filtering method was very close and could be argued to be better as it's RSME's were slightly worse, but the NASA scoring

function was better. The work was benchmarked against (Kong et al., 2019) which was considered state-of-the-art at the time of writing and found to be significantly worse.

There are also approaches which combine life estimation approachs, such as (Xu et al., 2014) who developed a combined, or fusion, approach. The data driven elements focussed on using Support Vector Regression [SVR] and RNNs with the CMAPSS dataset targeted RUL. The research observed that the RNN required significant volumes of data and was prone to overfitting. However, SVR struggled with large scale data sets but managed overfitting quite well as it has a strict theoretical and mathematical basis. An interesting approach taken by (Xu et al., 2014) was to refine the initial 21 predictor sensors down to just 7 key sensors which appeared to contain the bulk of the degradation information. The study concluded that the fusion approach was optimal, however a comparison of SVM and RNN showed that RNN was higher performing with a test PCC of 0.948 versus 0.918 and was also higher on other test metrics including MSE.

The study completed by (Xu et al., 2014) was also used as calibration for (Liu et al., 2015) who focussed on sensor selection methodology. The Shannon entropy equation was used to determine the measure of uncertainty of information contained in set or measure or a data series. The predictor sensor set was also reduced to a total of seven in this instance, with two sensors in common to the (Xu et al., 2014) Study: total temperature at LPT outlet and bypass ratio. The additional sensors of interest, selected in this instance, were total temperature at LPC outlet, physical fan speed, physical core speed, corrected core speed and bleed enthalpy. Multiple Linear Regression [MLR] and Gaussian Process Regression [GPR] were used to target RUL, with a conclusion that the entropy-based method was better performing than the former method of (Xu et al., 2014). However, it was noted that the researchers limited their performance results only to the accuracy of the last prediction per unit, i.e. at the point when RUL hits zero and used this absolute error as the benchmark with the entropy-based result being 1 and 2 and the former method being 6 and 6 with MLR and GPR respectively.

(Zhang et al., 2020) proposed a new metric which evaluated the trend of a sensor across a fleet of engines and used to ensure the most informative sensors were selected. These were then used with Functional Principal Component Analysis [FPCA] to model the engines health index and the combined with Bayesian inference to predict healthy prognostics. The paper discusses how SVR, ANN, Fuzzy Logic techniques are power for data mining complex datasets but predicted results can be hard to explain and cannot handle uncertainty, however, statistical approaches can. The researchers made reference to (Lei et al., 2018) and how HI's can be generated for multi-sensor data, in this instance the FPCA makes use of correlation, monotonicity and robustness for their sensor selection which was reduced to total temperature out at LPC, total temperature out at HPC, total temperature out at LPT, total pressure out at HPC, static pressure at HPC out, ratio of fuel flow to static pressure at HPC out, bypass ratio, HPT coolant bleed and LPT coolant bleed.

2.2. Evaluation

It is clear in the background research that minimal effort has been made to explain how the prognostics are making their predictions in the context of the domain, particularly with the CMAPSS and latterly the N-CMAPSS datasets. Quite often all predictor parameters available were automatically used with a given algorithm to develop the predictive model, with little effort to examine reducing the number of predictors and the complexity of the model – which could in turn help with explain-ability.

Even in papers which have focussed on feature selection methodologies, these have been heavy in their use of statistical methods for feature reduction and determining which features offer the most

informational value to the prognostic but lack any real focus on the domain context and system they're making predictions on.

This work will make use of the methodology employed by (Lövberg, 2021) in creating a healthy model to predict the physical sensor readings assuming normal degradation and then compared to the actual degradation in the sensor readings to understand which sensors are providing the most information to the algorithm for the prognostic. Most importantly, the feature selection will relate back to the context of the problem – which is an aerodynamic mechanical problem of engine performance degradation. The sensor selection will combine machine knowledge with statistical reduction to feed forward into the prognostic development.

The prognostic model itself will be built as a hyper-model to provide high accuracy, but also manage overfitting and produce a model which generalises well.

3. Project Management

3.1. Aims

The aim of this proposal is to apply a suitable machine learning algorithm to develop a prognostic model for aircraft engines under real flight conditions, that is both explainable and interpretable.

3.2. Objectives & Milestones

To achieve the aim of this project, there will be six key objectives and measures.

1. Data exploration: analysis of dataset, understanding and characterising the dataset, identification of pre-processing requirements and data structure for modelling.

   Outcome: understanding of data structure and prognostic modelling proposal.

2. Development of a suitable model to understand which predictors contain the bulk of the degradation information.

   Outcome: modelling technique for degradation understanding.

3. Reduction of predictor set to key predictors based on modelling from objective [2] and first principles of thermodynamics.

   Outcome: reduced predictor parameter set.

4. Modelling of prognostic based on reduced parameter set developed in objective [3].

   Outcome: trained and validated RUL prognostic based on reduced parameters.

5. Modelling of prognostic based on full parameter set for direct comparison to objective [4].

   Outcome: trained and validated RUL prognostic based on all parameters.

6. Comparison of models developed in objective [4] and objective [5] with SHAP analysis of both models to understand how the model is using different predictor variables.

   Outcome: comparison of models with SHAP analysis for explain ability.

3.3. Measures

The target measure for model development will be Remaining Useful Life [RUL], as an absolute value – making this a regression problem to solve.

Two key metrics will be used to evaluate the solution:

1. Root-mean-square-error [RMSE]

$$RMSE = \sqrt{\frac{1}{m_*}\sum_{j=1}^{m_*}(\Delta^{(j)})^2}$$

Equation 1: RMSE

2. NASA scoring function

$$s = \sum_{j=1}^{m_*} exp\left(\alpha\left|\Delta^{(j)}\right|\right)$$

Equation 2: NASA scoring function

Where m_* denotes the total number of test data samples, Δ^j is the delta between the estimated and the ground truth RUL of the j sample and α is $\frac{1}{13}$ if RUL is under-estimated and $\frac{1}{10}$ otherwise (M. Chao, 2021).

3.4. Deliverables

The key deliverable will be a research report capturing the development of the proposed solution, the code, test results and critical analysis of the success of the solution.

Full list of deliverables:

1. Report of implementation of research.
2. Models and scalers for degradation on each physical sensor channel.
3. Model and scaler for reduced parameter set prognostic.
4. Model and scaler for full parameter set prognostic.
5. Python code of research and prognostic.

3.5. Methodology

The hypothesis is that not all physical sensor channels will provide degradation information in the unhealthy state of engine performance degradation and that some physical sensor channels will provide more information than others. This research focusses on determining which physical sensors provide the most information and producing a prognostic model based on only these sensors, removing additional unrequired sensors to minimise noise in the prognostic. Through the reduction of sensors, it also stands that the model is much more explain-able in the context of the domain – engine RUL.

To achieve this the performance of each engine must be understood in its healthy state, based on the scenario descriptors which will form the sensor predictor model, mapping scenario descriptors to physical sensor readings as the target. This will be completed by developing a deep learning model of densely connected layers that can predict the expected physical sensor reading for a given scenario.

Using this physical sensor model, predictions can be made during the engines unhealthy state to predict what the sensor should read if the engine was still healthy. The delta between this value and the actual measurement in the dataset will highlight which sensors see the most monotonic degradation of the engine over time. The sensors with the highest level of degradation are the key sensors to include in the prognostic model. Using this degradation information alongside thermodynamic first principles a determination can be made regarding how a model will likely work.

From this reduced set of predictive parameters, a prognostic model will be developed, this will be a CNN based on: research, anticipated accuracy, and ease of implementation. It is important to note the focus of this research is explain-ability of the prognostic and understanding how and why it makes the predictions it does rather than pushing the state of the art on accuracy – this has been presented in detail in the background.

To understand the performance of the reduced parameter set model a second prognostic using the full set of physical sensor predictors will be trained, validated, and tested. The performance of this model can then be compared to the baseline reduced parameter set model to understand if the reduced parameter set model has all the crucial degradation information. This helps validate that the reduced parameter set model performs well.

Finally, SHAP analysis will be used on both models to understand which parameters are key contributors to the predictions made as well as a direct comparison between the two models to understand which, if any, predictors might have improved the reduced parameter set model.

3.6. Gantt Chart

Task Name	Duration	Start	Finish	Predecessors
Proposal Draft	5 days	Mon 28/02/22	Fri 04/03/22	
Literature Review	15 days	Mon 07/03/22	Fri 25/03/22	1
Break for Assignment	10 days	Mon 28/03/22	Fri 08/04/22	2
Complete Scoping	5 days	Mon 11/04/22	Fri 15/04/22	3
Exploratory Data Analysis	5 days	Mon 18/04/22	Fri 22/04/22	4
Feature Reduction	20 days	Mon 25/04/22	Fri 20/05/22	5
Prognostic Development	20 days	Mon 23/05/22	Fri 17/06/22	6
All Parameter Prognostic	15 days	Mon 20/06/22	Fri 08/07/22	7
Comparative Testing	5 days	Mon 11/07/22	Fri 15/07/22	8
SHAP Analysis	20 days	Mon 18/07/22	Fri 12/08/22	9
Documentation	20 days	Mon 15/08/22	Fri 09/09/22	10
Code Clean Up	20 days	Mon 15/08/22	Fri 09/09/22	10
Project Complete	0 days	Fri 09/09/22	Fri 09/09/22	11,12

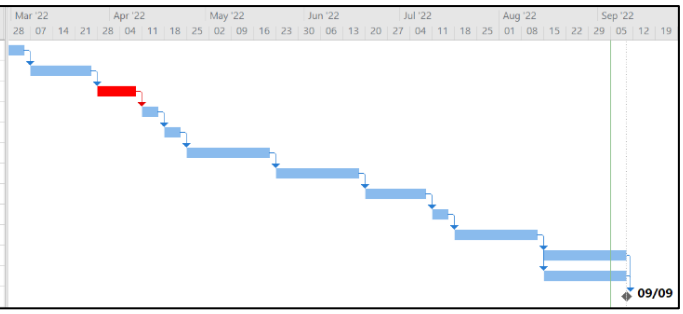


Figure 1: Gantt Chart of Project Plan

3.7. Critical Path

The project is sequential with parallel pathing of documentation and code clean up towards the back end of the project. There will be a short break at the end of March and start of April, for the researcher to focus on a taught module assignment before resuming, the aim is to complete the literature review ahead of this and then resume by completing the scoping. The project is targeting a September 2022 submission, with a start of March 2022 due to logistical challenges with the researcher's original project.

3.8. Risk Register

ID	Risk	Category	Probability	Impact	Risk	Mitigation	Probability	Impact	Risk
1	Time required to train models on large dataset	Project	9	9	81	A	3	9	27
2									
3									
4									

3.9. Mitigation Plan

A. Model iterations and hyper-parameter space exploration to be completed with a data frequency reduction from 1 Hz to 0.1 Hz, similar to (Lövberg, 2021). This will allow quicker iteration, with low deterioration in performance due to long term trends [inter-cycles] being more important than intra-cycle data.

4. Problem Overview & Data Description

4.1. Problem Overview

The challenge is to predict the RUL of a series of turbofan aircraft engines, given each engines historical missions described through a series of scenario descriptors and physical sensor measurements.

The dataset has been provided courtesy of the NASA Ames Prognostics Data Repository (M. Chao, 2021) and is a collection of realistic turbofan engine run to failure simulations developed synthetically with Commercial Module Aero-Propulsion System Simulation [CMAPSS] system model developed by NASA.

The dataset consists of eight subsets, each subset contains:

- Between 9 and 54 individual engines [or units].
- Covering between two and three different class of flights [or cycles], with class 1 being 1 to 3 hours in length, class 2 being 3 to 5 hours in length and class 3 being over 5 hours in length.
- Failure modes that can affect up to five different components: fan, low pressure compressor [LPC], high pressure compressor [HPC], high pressure turbine [HPT] and low-pressure turbine [LPT].
- Impacting on two performance parameters: efficiency and flow.

The dataset overview can be exampled in Table 1 below.

Table 1: Overview of complete dataset and subsets (M. Chao, 2021)

Name	# Units	Flight Classes	Failure Modes	Fan		LPC		HPC		HPT		LPT		Size
				E	F	E	F	E	F	E	F	E	F	
DS01	10	1, 2, 3	1							☑				7.6 M
DS02	9	1, 2, 3	2							☑		☑	☑	6.5 M
DS03	15	1, 2, 3	1							☑		☑	☑	9.8 M
DS04	10	2, 3	1	☑	☑									10.0 M
DS05	10	1, 2, 3	1					☑	☑					6.9 M
DS06	10	1, 2, 3	1			☑	☑	☑	☑					6.8 M
DS07	10	1, 2, 3	1									☑	☑	7.2 M
DS08	54	1, 2, 3	1	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	35.6 M

The CMAPSS model is a coupled system on nonlinear equations for calculating estimates of measured physical properties of a turbofan engine X_s, i.e. physical sensors, and non-observable properties of the engine X_v, i.e. virtual sensors. A schematic of CMAPSS model can be observed with Figure 2. The physical sensors monitoring the engine system can be observed in Table 3.

The inputs to CMAPSS are in one part scenario descriptors for the flight data W, including: altitude, flight Mach number and throttle-resolver angle. The second part is an unobservable set of health parameters ϑ which are quality parameters used to simulate the deterioration of the system performance. The scenario descriptors can be examined in Table 2. All channels are logged at a rate of 1 Hz. Another advantage that will be made use of with the newer dataset is that the cycle where a unit transitions from a healthy state to an unhealthy state is captured, with a portion of data available for the healthy state. This will be used to determine the key predictor variables, as will be

explained in more detail later. It should also be noted that noise has purposely been introduced to the physical sensor readings to simulate realistic data collection.

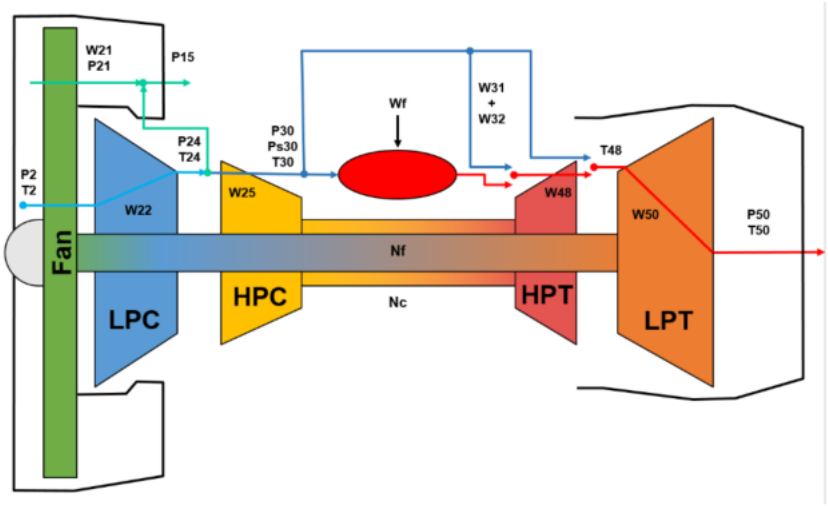


Figure 2: Schematic representation of the CMAPPS model (M. Chao, 2021)

The dataset this research will focus on is DS05, as this has one failure mode but has a failure mode in one of the earlier stages – the HPC. This allows the method developed to understand the knock-on impact of the failure and if the degradation in performance of the HPC can be measured in the performance parameters of the stages further downstream such as the HPT and LPT.

4.2. Turbofan Engines and Performance

Turbofan engines, or bypass turbojets, are efficient, quiet and reliable (El-Sayed, 2017) engines widely used in commercial aviation. They primarily consist of three sections: the fan and compressor, the combustion chamber, and the turbine. Turbofans are named as such because the bulk of the air drawn by the fan is bypassed around the engine core, see Figure 2, to provide the thrust.

A small portion of the air drawn in by the fan is channelled through the core where it is compressed, in this instance twice via two axial compressors the LPC and the HPC. The air is then mixed with fuel in the combustion chamber providing enthalpy into the system, which is extracted via a series of axial turbines – initially the HPT and then latterly the LPT. The turbine is then connected to the compressor via a common shaft which is used to translate the turbine work to the compressor and fan to continue to draw in more air at the fan.

Due to turbofans being axial devices, axial compressor and turbines are limited in pressure ratio, hence multiple stages each with a stack of rotor and stator rows are usually employed. It is not uncommon for each stage to have six or more rotor blade rows (Jansohn, 2013). Several turbomachinery equations are critical in developing the explain-ability of the prognostic model, to be able to understand the predictions and predictor parameters that are being used, it must first be understand the domain context.

The total to total [T-T] pressure ratio of a compressor is defined as the total outlet pressure divided by the total inlet pressure, as detailed below in equation 3.

$$PR = \frac{P_2}{P_1}$$

Equation 3: Compressor Pressure Ratio

The compressor work is defined as the product of the mass flow, in kgs^{-1} the specific heat-capacity of the gas C_p and the delta temperature rise over the device, as detailed below in equation 4.

$$W_c = \dot{m} \times C_p \times (T_2 - T_1)$$

Equation 4: Compressor Work

The compressor efficiency is defined as the product of the useful pressure ratio increase over the device and the inlet temperature, divided by the temperature rise across the device, as detailed below in equation 5.

$$\eta_c = \frac{T_1 \times PR^{\left(\frac{\gamma-1}{\gamma}\right)}}{T_2 - T_1}$$

Equation 5: Compressor Efficiency

It can be observed then, when examining equations 4 and 5 that temperature plays an important role in the performance of a compression device. As outlet temperature rises, T_2, efficiency and work drops.

Turbine work is defined as the gross useful power of the turbine, which is the product of turbines isentropic work and its total to static [T-S] efficiency, as detailed below in equation 6.

$$W_t = \eta_t \times W_{isen}$$

Equation 6: Turbine work

The isentropic work is a product of the mass flow, in kgs^{-1}, specific heat-capacity of the gas C_p, the turbine inlet temperature T_3 and the expansion ratio across the device, as detailed below in equation 7.

$$W_{isen} = \dot{m} \times C_p \times T_3 \times \left(1 - \left[\frac{1}{ER}\right]^{\frac{\gamma-1}{\gamma}}\right)$$

Equation 7: Turbine isentropic work

Where the expansion ratio is defined as the total inlet pressure of the device, P_3 divided by the static outlet pressure P_4 as detailed below in equation 8.

$$ER = \frac{P_3}{P_4}$$

Equation 8: Turbine expansion ratio.

Finally, a turbomachine is a work balance between compressor and turbine – the compressor can only deliver the work provided by the turbine as they're connected by a common shaft with only mechanical losses in between, i.e. bearing losses.

If a turbomachine is working as intended at the start of its service life, it can be assumed that the geometry of said turbomachines is as per the specification. It follows that if the performance changes, then the geometry of said machine must have changed to accommodate this shift as they are intrinsically, physically linked. This shift could be due to a change in operating conditions, such as moving from a cooler "warm-up" condition to a hotter "running" condition, whereby there is some

form of elastic deformation or thermal growth within the mechanical components driven by the temperature change. For instance (E. E. Halila, 1982) as cited in (Shih & Yang, 2014), see Figure 3, nicely illustrated the impact of running clearance between an actively cooled and an uncooled casing diameter and that an actively cooled case could significantly reduce the running clearances during operation. It will be crucial that this "healthy" state with normal performance variation and noise be initially understood, and the scenario descriptors mapped to the physical sensor readings.

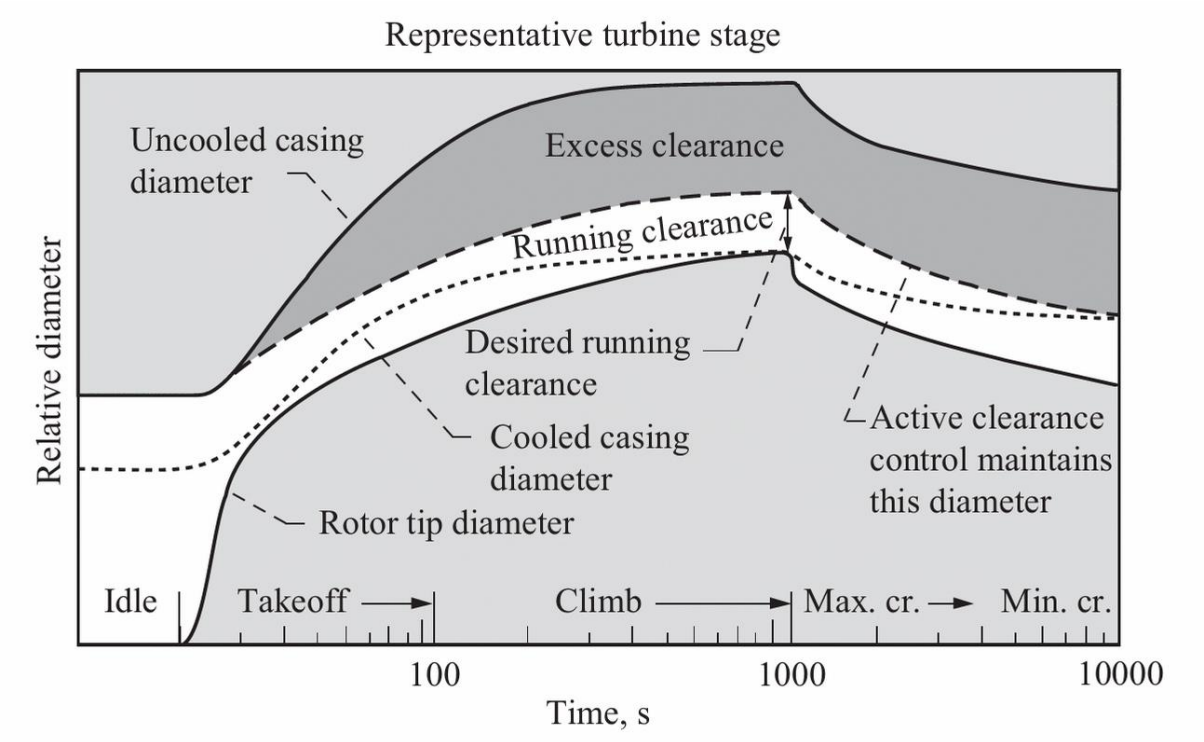


Figure 3: Effects of case cooling on HPT blade tip clearance during take-off (Shih & Yang, 2014)

This shift could also be due to irreversible damage propagation such as plastic deformation, cracking, or wear on the components, leading to a monotonic degradation of performance – such as those of interest in this report. When combined with observations by (Lattime & Steinetz, 2003) as cited in (Shih & Yang, 2014) of 0.25mm HPT tip clearance equating to ~1% specific fuel consumption and a ~10 °C increase in exhaust gas temperature – it can easily be seen that gradually increasing clearances through wear or degradation would impact the systems performance, as measured by the Health Index, and exhibit a performance degradation within the physical sensors X_s of the dataset. As (Jansohn, 2013) notes, there are many factors that can lead to gas turbine degradation – fouling, erosion, seal tip gap enlargement, hot section deformation, and others. The specific mechanism is not known or of interest in this research.

By understanding how the individual engines perform in the healthy state or $H_s = 1$, the engine can be characterised, and predictions made about future performance based on the flight cycle and scenario descriptors. Using these predictions and comparing against the actual sensor readings once the engine has taken on a failure mode and is in $H_s = 0$, such as efficiency degradation, the specific physical sensors which highlight the degradation and can be used for RUL predictions can be identified.

5. Exploratory Data Analysis

5.1. Data Structure

The first key table within the dataset are the scenario descriptors *W* for flight data. These are detailed in Table 2 and an example plot of each descriptor against time for one unit, during one cycle, [specifically Unit 5, Cycle] has been illustrated in Figure 4. It can be observed that Altitude and Mach number will be highly correlated, and that T2 and Altitude will also have a strong correlation but Throttle Resolver Angle [TRA] will likely be lower.

Table 2: Scenario descriptors for flight data, W. (M. Chao, 2021)

Id	Symbol	Description	Units
1	Alt	Altitude	ft
2	Mach	Flight Mach number	-
3	TRA	Throttle-resolver angle	%
4	T2	Total temperature at fan inlet	°R

The physical sensor measurements, X_s, is the second key set of variables for the development of the prognostic. They can be observed in Table 3. These predictor variables offer the insight into the health of the unit over time, with subtle, gradual monotonic increases or decreases anticipated to guide predictions on RUL. It is in these physical sensor measurements that we will also draw insight into *why* the model makes the predictions it does, and which sensor measurements carry the most informational value.

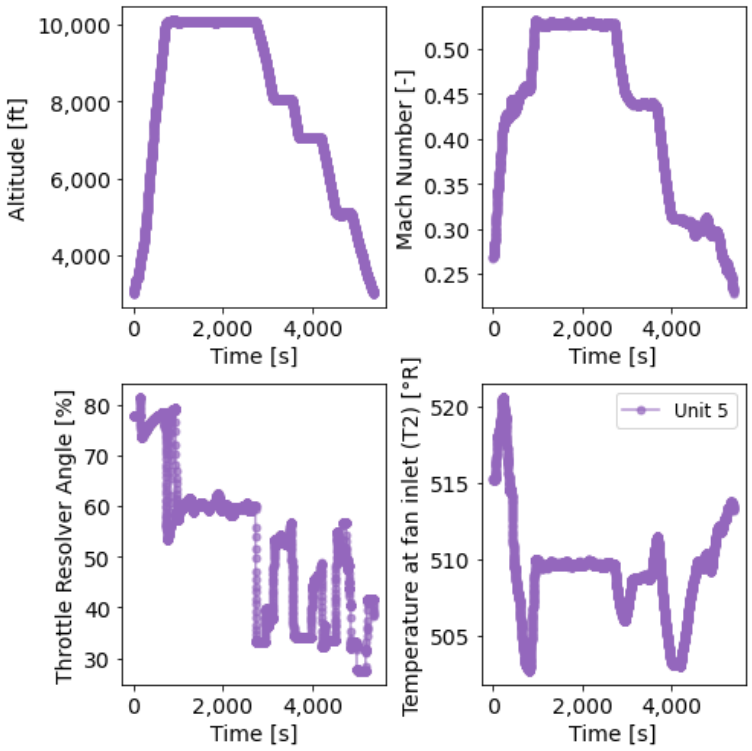


Figure 4: Scenario descriptors [W] for DS05, Unit 5, Cycle 5. (M. Chao, 2021)

The physical sensor measurements cover a series of standard temperature and pressure measurements after each key stage within the engine: the fan, LPC, HPC, HPT, LPT as well as fuel flow and shaft rotational speeds. As per the turbomachinery equations outlined in section 4.2, these

parameters will be crucial in assessing the performance health of the engine and isolating the monotonic degradation of the most informational channels to estimate RUL.

Table 3: Physical sensor measurements [Xs] for DS05, Unit5, Cycle 5. (M. Chao, 2021)

Id	Symbol	Description	Units
1	Wf	Fuel flow	pps
2	Nf	Physical Fan Speed	rpm
3	Nc	Physical core speed	rpm
4	T24	Total temperature at LPC outlet	°R
5	T30	Total temperature at HPC outlet	°R
6	T48	Total temperature at HPT outlet	°R
7	T50	Total temperature at LPT outlet	°R
8	P2	Total pressure at fan inlet	psia
9	P15	Total pressure in bypass-duct	psia
10	P21	Total pressure at fan outlet	psia
11	P24	Total pressure at LPC outlet	psia
12	Ps30	Static pressure at HPC outlet	Psia
13	P40	Total pressure at burner outlet	psia
14	P50	Total pressure at LPT outlet	psia

Figure 5 shows a series of scatter plots for each of the sensor measurements for unit 5, cycle 5, against time. Visually examining the physical sensor channels against time this way shows some very interesting characteristics. For instance, many of the temperature and pressure channels appear to have a high level of collinearity, for example: T24, T30, T48 and T50. It also seems apparent at which point the engine is in a cruise condition with little changes in temperature and pressure for an extended period.

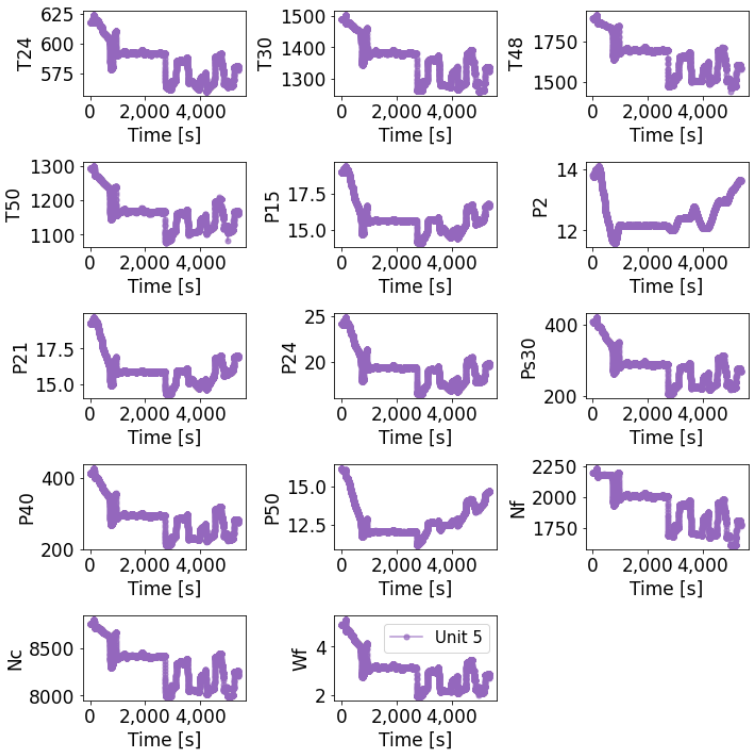


Figure 5: Physical sensor readings [Xs] for DS05, Unit 5, Cycle 5. (M. Chao, 2021)

The final key piece of information in the dataset is the auxiliary data which contains the unit number, flight cycle number, flight class and health state. While the first three parameters are key for indexing and tracing the fourth parameter of health state is crucial for the research contained within, as it is through understanding the engines performance in the healthy state and then monitoring the degradation in the unhealthy state that will lead to determining the key predictor variables and estimate the RUL. Figure 6 shows how the health state transitions from H_s 1 where the engine is healthy, to H_s 0 where the engine is unhealthy and has taken on a failure mode with abnormal and accelerated levels of degradation to critical functions.

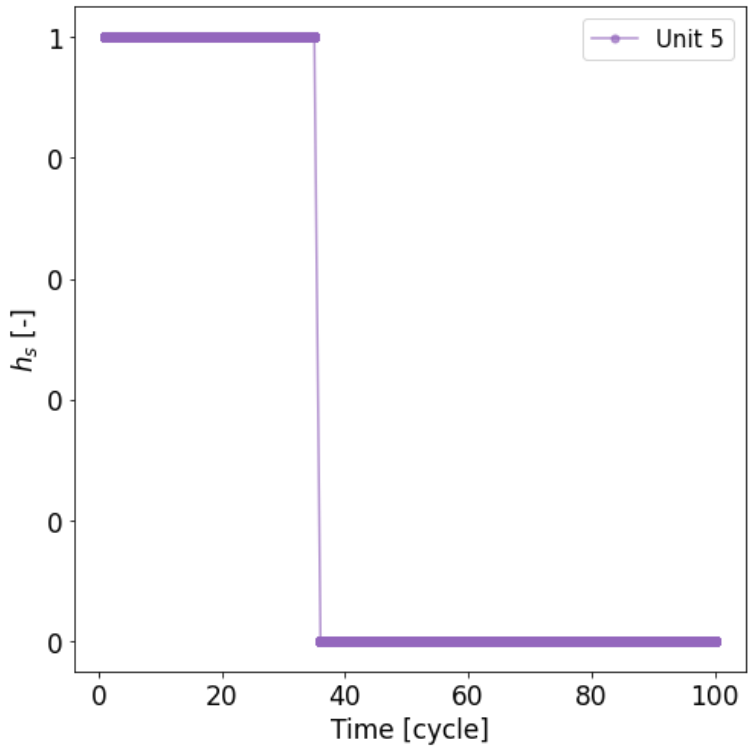


Figure 6: Health state of Unit 5, with a transition from Hs 1 to Hs 0 at cycle 35. (M. Chao, 2021)

5.2. Correlation of Scenario Descriptors to Physical Sensors

An initial examination of correlation between scenario descriptors and the physical sensor measurement channels, for cycles during H_s = 1, was completed using Spearman's Rank Order, to understand the interaction of all eighteen channels that could be used in the prognostic model. Initially only H_s 1 data was used to remove any potential noise due to degradation of sensor measurements anticipated during H_s = 0. It should be noted that all sample data points for all units and cycles were used in the correlation analysis. Figure 7 illustrates the correlations in a heat-map with strong correlations between several scenario descriptors and physical sensors.

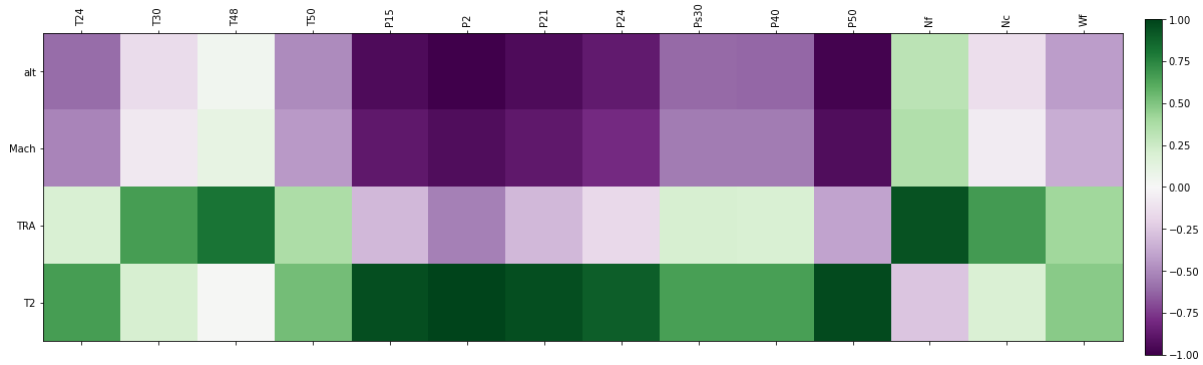


Figure 7: Spearman correlation between scenario descriptors and physical sensors in Hs 1.

An interesting observation is to complete another correlation analysis during the unhealthy state and then examine the delta between healthy correlations and unhealthy correlations, which can be observed in Figure 8. The scale has been significant reduced, but there are already key sensors starting to stand out such as Ps30, P40, T24, P24 and Wf.

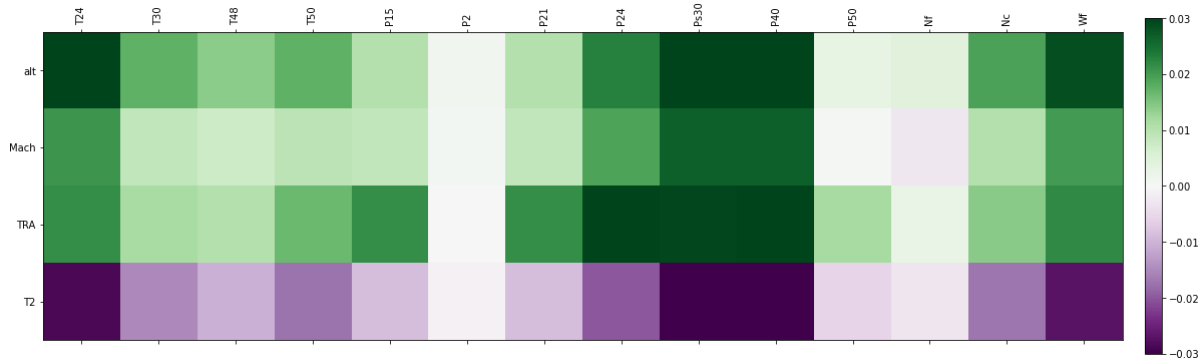


Figure 8: The delta Spearman correlation between scenario descriptors and physical sensors.

5.3. Structuring of the Data

The data is provided as a series of engine units, in DS05 there are 6 units available for training and 4 units for testing. Each unit has several flight cycles between 65, for unit 3, and 100 for unit 5. Each cycle then has scenario descriptors, physical sensor measurements and virtual sensor measurements captured at 1Hz for the entire flight cycle.

The target data is provided as a RUL and is also at the 1Hz resolution, however, is at a constant value for a given cycle. For example, unit 1 which has 80 cycles, has a RUL of 79 for each 1Hz observation throughout the entire cycle and it is at this level that the prediction is required to be provided. Thus, the RUL is a remaining number of cycles or flights or missions. An example has been outlined in Table 4, overleaf.

Table 4: Example data structure for Unit 1, Cycle 1 with RUL target data.

Unit	Cycle	t	Wf	Nf	Nc	…	P50	RUL
1	1	1	5.280657	2240.817081	8822.782067	…	1305.942703	79
1	1	2	5.279919	2240.682999	8822.565341	…	1305.955847	79
1	1	3	5.279245	2240.711263	8822.534731	…	1305.888172	79
1	…	…	…	…	…	…	…	…
1	80	11369	2.509348	1694.968199	8115.223114	…	14.447787	0
1	80	11370	2.517648	1695.666667	8116.793081	…	14.455600	0
1	80	11371	2.514809	1696.279190	8116.970648	…	14.454369	0

5.4. Modelling Proposal

There were two independent models developed for the purposes of this research:

- Scenario descriptors [W] as predictors and physical sensors [X_s] as targets during $H_s = 1$.
- Scenario descriptors [W] and a subset of physical sensors [X_s] as predictors and RUL as the target.

For the sensor predictive models, a fully connected deep neural network was used, as the target data [the physical sensors] are in the same time domain as the predictors [the scenario descriptors]. Lower complexity Machine Learning algorithms were considered, however due to the anticipated complexity of the interactions between predictors and targets and the volume of data available for training – it was deemed suitable to use a DNN.

Due to the multilevel dimensionality of each cycle having a time-series at 1Hz and the target RUL being at the cycle level, the data effectively has a higher number of dimensions than a structured 2d table of observations and predictors. Hence, the prognostic model was suited to either a many-to-one LSTM or 1D convolutional neural network to deal with dimensionality and structure of the data. It had been found in the background research that CNNs offered good performance (Li et al., 2018) for time series and time frequency problems and (Thoppil et al., 2021) found that CNNs were the most popular DL architecture employed in machine prognostics; hence a CNN will be used for the research in this project.

It is obvious that the dataset is relatively large, with 4.35M observations in the training set and 2.56M observations in the test set. For faster iterations of model training, particularly as hyper-models will be used that can have their hyper-parameters optimised as well as, the prognostic model will have the intra-series data sampled down from 1Hz to 0.1Hz significantly reducing the computational overhead. This method was used by (Lövberg, 2021) who showed that the inter-series information is more important for RUL predictions across a high number of cycles.

The model training was completed using a personal computer with an intel i9-10900k processor, 64GB DDR4 RAM and an nVidia RTX3080 10GB GPU; so most hyper-model optimisations were completed with less than 24hrs total computation time.

6. Physical Sensor Predictive Model

The four scenario descriptor variables were used to predict the fourteen physical sensors. The training data was first filtered based on the health state, ensuring only healthy data was used. The data was then split into training and testing based on unit, with units 1 to 4 and then 5 to 6 in the training and test splits, respectively.

A hyper-model DNN was created with the hyper parameters of number of hidden layers, number of neurons per layer, dropout rate and learning rate with an Adam optimiser. The loss was set as Mean Square Error [MSE] and all activations were set to ReLU, as the default for deep learning of complex maps. The input and output layers had a fixed number of neurons of 4 and 14 respectively. The hyper-model was tuned using a Bayesian optimisation, with each trial having 50 epochs, up to 200 trials being completed and optimising on the MSE of the 20% validation split.

Including the input and output layers, the total number of trainable parameters of the highest scoring model was 33,677,350 as the optimizer tended towards the maximum number of neurons available for 4 out of 5 of the layers it could tune, the architecture can be observed in Table 5. This indicates that there is likely more room for optimization if the hyper-model was given the room, but also that a relatively high complexity model was required – particularly when benchmarked against similar models which used 4 dense layers of 128 neurons (Lövberg, 2021).

Table 5: Sensor predictor model architecture.

Layer (type)	Output Shape	Param #
Input (Dense)	None, 4	20
Dense 1 (Dense)	None, 4096	20,480
Dense 2 (Dense)	None, 4	16,388
Dense 3 (Dense)	None, 4096	20,480
Dense 4 (Dense)	None, 4096	16,781,312
Dense 5 (Dense)	None, 4096	16,781,312
Dense 6 (Dense)	None, 14	57,358
	Total	**33,677,350**

In examining the results in detail, the Mean Absolute Percentage Error [MAPE] for each sensor, it can be observed that some sensors had a low error rate; for example, T30 has an average of ~0.08% whereas others had a relatively high error rate; for example, Wf at over 8%. To determine the generalisation of the model, a Mann-Whitney U test was conducted on each sensor to understand if the distribution of the training sample MAPE was the same as the test sample MAPE. The results can be reviewed in Table 6. It can be observed that several sensor predictions do not generalise well between the training and test set, specifically: T48, P2, P24, Nf and Wf. It should be noted that in each instance, except the sensor Wf, the model appeared to be overfitted.

Considering that the level of measured degradation with some sensors could be relatively small in comparison to the level of error, the level of generalisation and the high complexity of the model, it was concluded that the single model design for predicting physical sensor values was not suitable for the purpose of determining which sensors would provide the bulk of the degradation information.

Table 6: Hypothesis test for generalisation of single model.

Sensor	Average MAPE		U	P-Value	Conclusion
	Training Set	Test Set			
T24	0.28%	0.30%	2369	0.323	Accept null hypothesis
T30	0.08%	0.09%	2456	0.514	Accept null hypothesis
T48	0.10%	0.11%	2070	0.032	Reject null hypothesis
T50	0.23%	0.24%	2460	0.524	Accept null hypothesis
P15	1.02%	1.02%	2533	0.722	Accept null hypothesis
P2	3.68%	4.01%	1960	0.010	Reject null hypothesis
P21	1.19%	1.06%	3085	0.077	Accept null hypothesis
P24	1.24%	1.40%	1868	0.004	Reject null hypothesis
Ps30	0.47%	0.47%	2484	0.586	Accept null hypothesis
P40	0.48%	0.49%	2422	0.433	Accept null hypothesis
P50	3.79%	3.71%	2989	0.163	Accept null hypothesis
Nf	0.08%	0.10%	1943	0.009	Reject null hypothesis
Nc	0.03%	0.03%	2775	0.567	Accept null hypothesis
Wf	8.60%	8.35%	3196	0.028	Reject null hypothesis

To understand if the error was due to sensor noise, i.e. irreducible error, or modelling error – a separate model was initiated using the same four scenario descriptors as predictors but this time only mapping to P2 and P50 as they had relatively high MAPE scores. After only 8 hyper-model iterations using Bayesian optimization – the prediction error was already significantly reduced to <2% for both the development and test datasets.

It was determined that an alternate approach was to be taken whereby each individual sensor would have its own, albeit lower complexity, model. The lower complexity was driven primarily through a significant reduction in the number of artificial neurons per hidden layer. The comparison of hyper-models can be examined in Table 7.

Each sensor was trained using the same data split of units 1 to 4 for training and units 5 to 6 for testing, with 20% of the training set being held out for validation during training of each model and tuning of the hyper-parameters.

Table 7: Comparison of hyper-parameter ranges for single versus individual sensor models.

Hyper-parameter	Single Model		Individual Models	
	Min	Max	Min	Max
Number of Hidden Layers	1	5	1	5
Number of neurons per layer	4	4096	2	256
Drop out	0.05	0.50	0.05	0.50
Learning rate	1E-05	1E-03	1E-05	1E-03

The results were significantly improved, as detailed in Table 8. The highest average MAPE for training or testing out of all fourteen sensors was reduced to 0.312%. Sensor Wf specifically was reduced from 8.60% to 0.204% in the training set with the test set sample not being statistically significantly different from the training set. Only two sensors were rejected against the Mann-Whitney U test null hypothesis, they were T30 and P50. Examining them in more detail it can be observed that P50 performed better on the test set. Only sensor T30 was found to have a statistically significant difference between training and testing with signs of overfitting. This can be observed to be a mean difference of 0.025% which is suitably acceptable against the requirement.

Table 8: Hypothesis test for generalisation of individual sensor models.

Sensor	Average MAPE		U	P-Value	Conclusion
	Training Set	Test Set			
T24	0.069%	0.070%	2735	0.676	Accept null hypothesis
T30	0.130%	0.155%	1621	0.000	Reject null hypothesis
T48	0.164%	0.168%	2370	0.325	Accept null hypothesis
T50	0.102%	0.103%	2569	0.828	Accept null hypothesis
P15	0.130%	0.141%	2654	0.916	Accept null hypothesis
P2	0.147%	0.167%	2831	0.431	Accept null hypothesis
P21	0.109%	0.110%	2721	0.716	Accept null hypothesis
P24	0.166%	0.174%	2762	0.602	Accept null hypothesis
Ps30	0.282%	0.284%	2573	0.840	Accept null hypothesis
P40	0.290%	0.312%	2175	0.083	Accept null hypothesis
P50	0.155%	0.147%	3161	0.040	Reject null hypothesis
Nf	0.127%	0.137%	2133	0.058	Accept null hypothesis
Nc	0.114%	0.121%	2408	0.402	Accept null hypothesis
Wf	0.204%	0.206%	2547	0.762	Accept null hypothesis

Figures 9 and 10 nicely illustrate the performance jump in moving from a single model of four predictors and fourteen targets to individual models of four predictors and a single target. The difference is most notable on P2 and P50, however for sensors that already had reasonable performance of <2% have also seen a measurable improvement in predictive performance and generalisation.

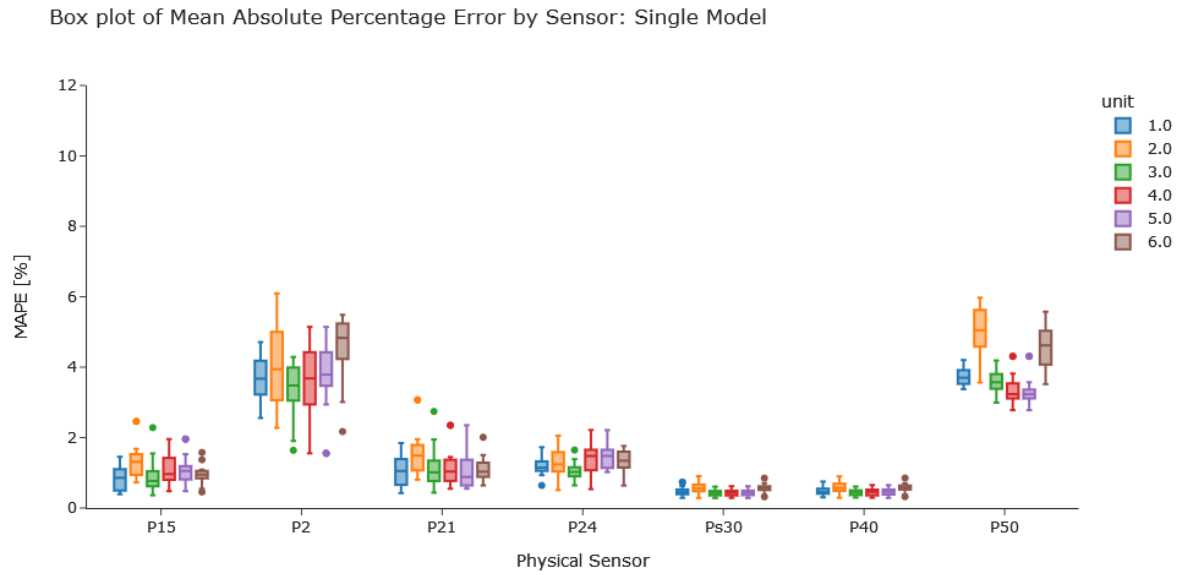


Figure 9: Boxplot of MAPE by sensor for each unit with a single model.

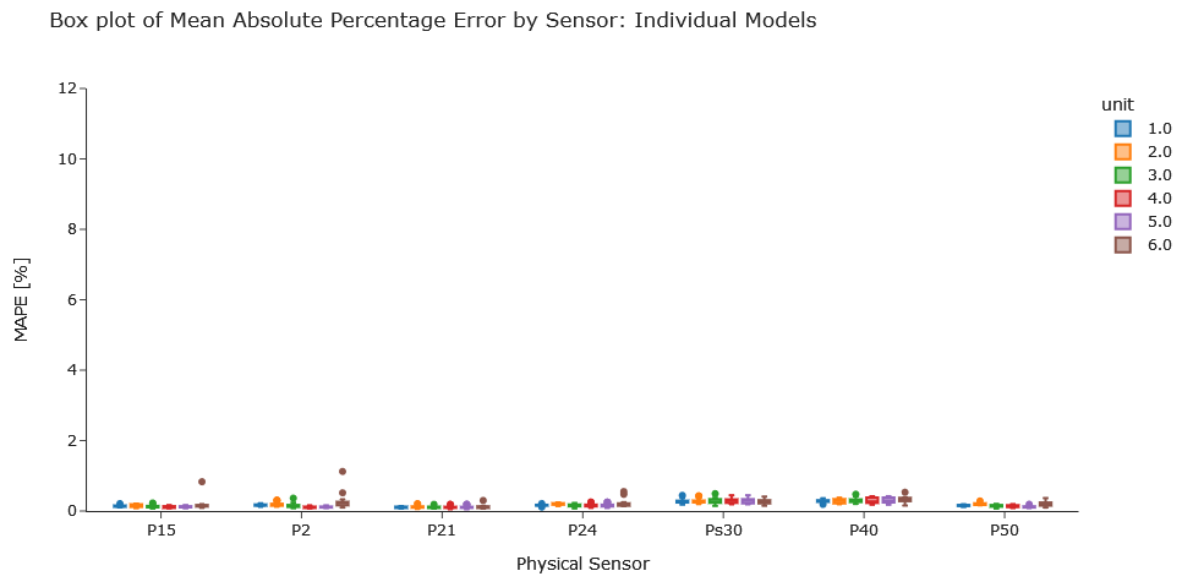


Figure 10: Boxplot of MAPE by sensor for each unit with individual sensor models.

7. Analysis of Predictors

Using the physical sensor predictor models developed in Chapter 5, predictions were made for the expected sensor value based on the flight scenario descriptors

A total of three tests were applied to each sensor to determine which sensors were expected to provide the bulk of the degradation information for the prognostic, this would then be combined with mechanical first principles to make a final selection. The first three tests were:

1. Visual examination of line plots for each sensor.
2. Mann-Whitney U test beginning vs. end of life.
3. Spearman Correlation of Mean Delta vs. Cycle.

For simplicity the visual examination was split into three categories: strong, weak or none. A strong trend would be on that showed around 2% pts mean delta between healthy and unhealthy operation for a given sensor across the life of the unit. A good example of this can be observed in Figure 11 for sensor T50.

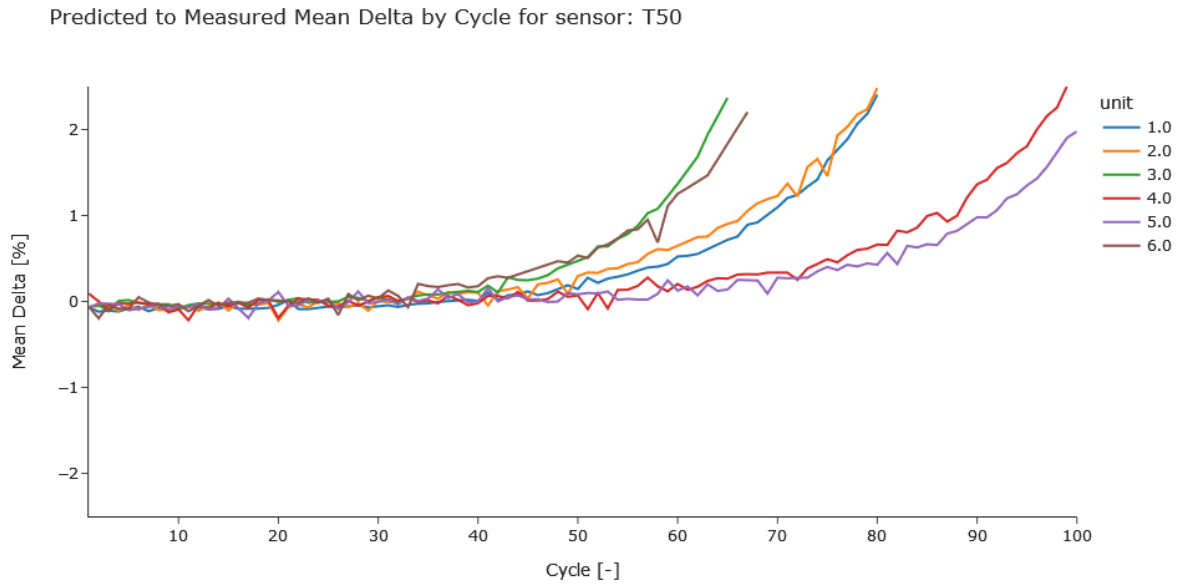


Figure 11: Line plot of Predicted to Measured Mean Delta by Cycle and Unit for Sensor T50.

A weak trend was determined to be that of a very low monotonic response over the life of the engine, perhaps only shifting around 0.25% pts mean delta between healthy and unhealthy operation, but an observable shift. Sensor T24, as can be observed in Figure 12, is a good example of a weak trend. Note how the y-axis limits of the line plot have been reduced from +/-2.5% to +/-0.5%.

Finally, a sensor with no trend would be one where the mean delta for the given sensor was zero over the life of the engine. Sensor P21 demonstrates this well, see Figure 13, despite the reduced y-axis limit of +/-0.5%. The only engine that might be exempt in this example is unit 3, the green line, where it appears as though the mean delta shifts up slightly in the last few cycles. However, this could also be noise as it is not seen in the other units.

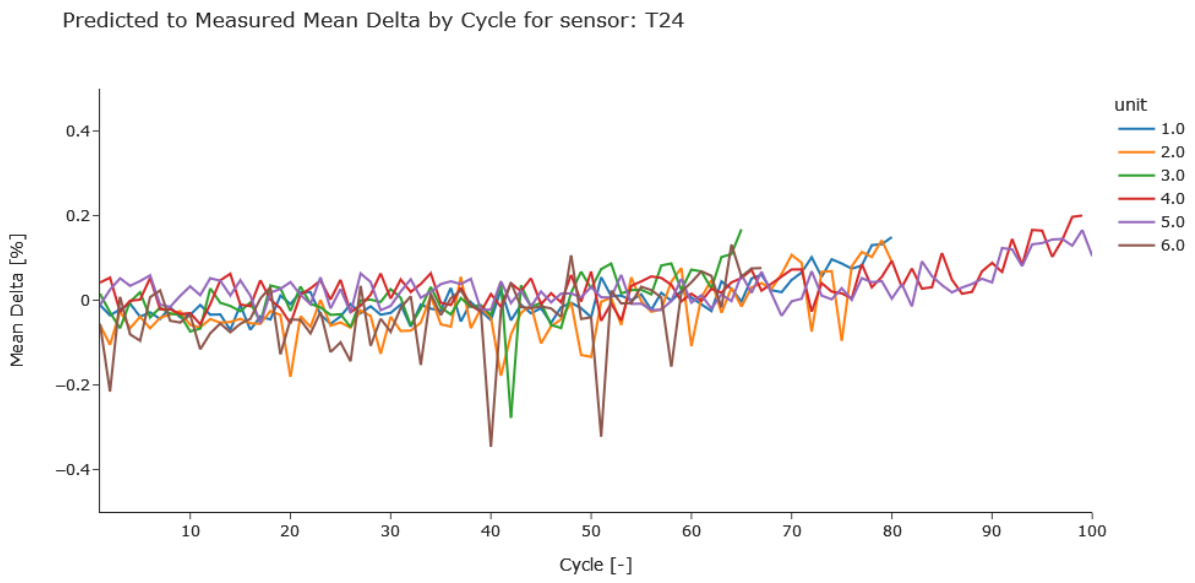


Figure 12: Line plot of Predicted to Measured Mean Delta by Cycle and Unit for Sensor T24.

In order to statistically prove that there was a significant shift between the start of a units life and the end, the first and last ten cycles by unit would be taken and used with a Mann-Whitney U test in order to understand if the populations were statistically significantly different to each other. The hypothesis test H_0 is that the samples are the same, with the alternate H_1 being that they are different.

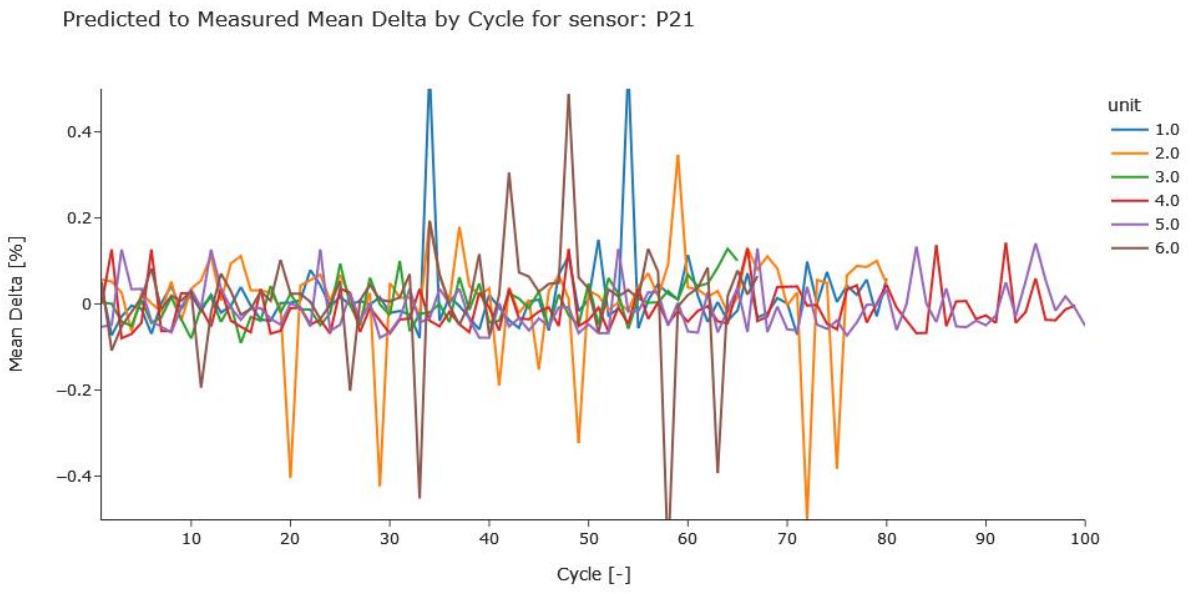


Figure 13: Line plot of Predicted to Measured Mean Delta by Cycle and Unit for Sensor P21.

A boxplot of all six units p-values by sensor was created to visualise the Mann-Whitney U test results, see Figure 14. As can be observed, several sensors have a definite difference in their first ten cycles versus their last ten cycles for each unit – such as T24, T30, T48 and T50. Others are potentially borderline such as P50 and Nc and then others appear to clearly not have a difference in

distribution such as Nf and P21. The results of the Mann-Whitney U test cannot be taken in isolation, they need to be considered in the context of the visual trends along with the final test which is a Spearman Correlation for the Mean Delta to cycle.

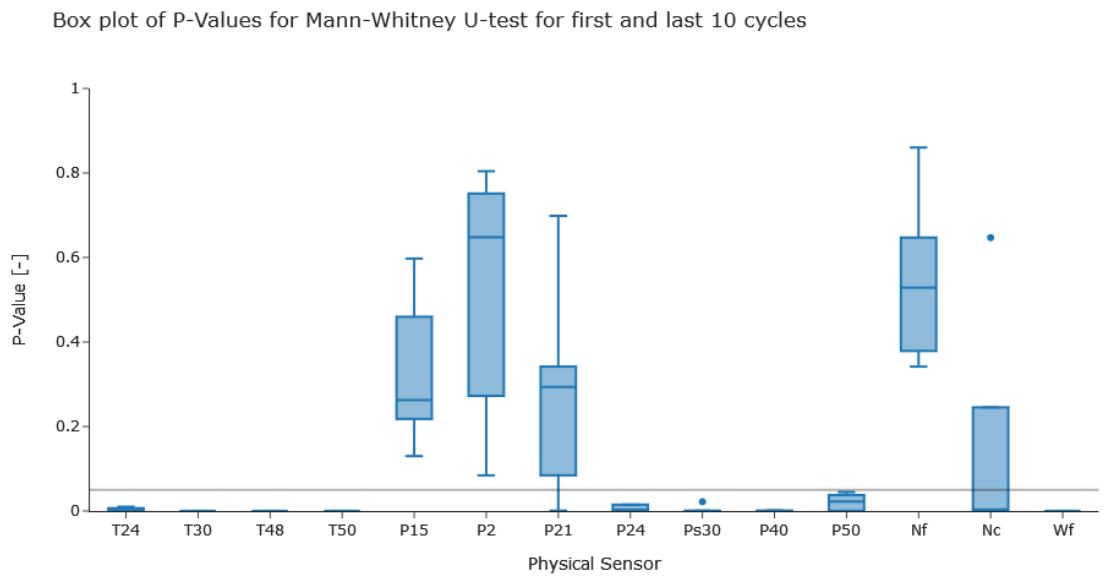


Figure 14: Box-plot of P-Values for first and last 10 cycles for each unit.

A Spearman ranked correlation was completed for mean delta in expected sensor reading versus measured sensor reading against cycle number, with the expectancy that the visual trends observed would be backed up by this last hypothesis test. This was completed for each of the 14 sensors and for each of unit. Again, a boxplot of p-values for all 6 units by sensor was created to visualise the results, see Figure 15.

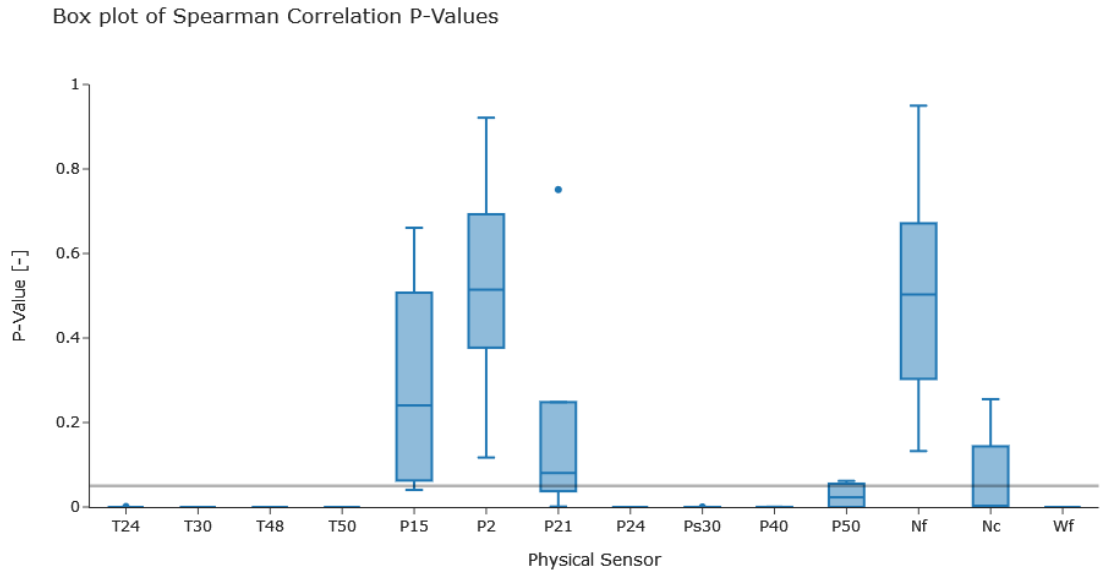


Figure 15: Box-plot of P-Values for Spearman Correlation of each unit.

The Spearman ranked correlation confirmed the visual observations with the Mann-Whitney results, with Nc and P21 still remaining slightly inconclusive as the distribution suggests that some units had a p-value below 0.05 whereas others did not. To decide on inclusion of these sensors a third boxplot was created of the Rho values, the correlation score, for any sensors with at least one unit with a p-value below 0.05, see Figure 16.

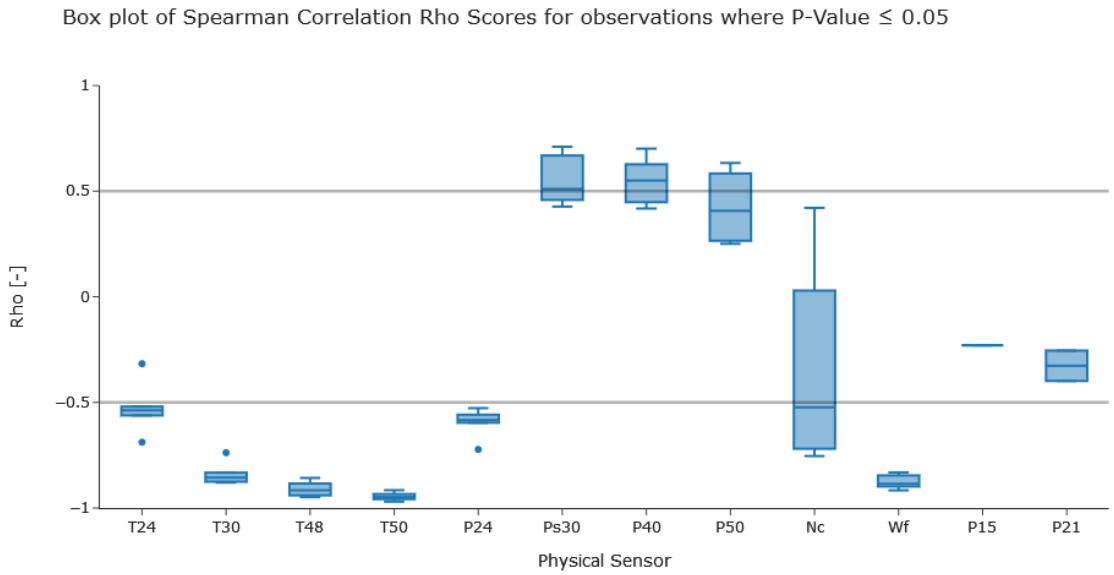


Figure 16: Box-plot of all sensors with P-Values ≤ 0.05.

Upon examining the boxplot of Rho values, P15 and P21 were ruled out due to having weak correlations. However, Nc was still inconclusive with a strong negative correlation for some units but then a weak positive correlation for others. A summary table of each sensor and test result including initial decision based on said tests is included as Table 9 below.

Table 9: Summary table of tests and decision by sensor.

Sensor	Description	Visual	Mann-Whitney U-test	Spearman's Correlation	Decision
Wf	Fuel flow	Strong	Reject	Reject	Include
Nf	Physical Fan Speed	None	Accept	Accept	Drop
Nc	Physical core speed	Weak	Accept	Accept	Drop
T24	Total temperature at LPC outlet	Weak	Reject	Reject	Include
T30	Total temperature at HPC outlet	Strong	Reject	Reject	Include
T48	Total temperature at HPT outlet	Strong	Reject	Reject	Include
T50	Total temperature at LPT outlet	Strong	Reject	Reject	Include
P2	Total pressure at fan inlet	None	Accept	Accept	Drop
P15	Total pressure in bypass-duct	None	Accept	Accept	Drop
P21	Total pressure at fan outlet	None	Accept	Accept	Drop
P24	Total pressure at LPC outlet	Weak	Reject	Reject	Include
Ps30	Static pressure at HPC outlet	Weak	Reject	Reject	Include
P40	Total pressure at burner outlet	Weak	Reject	Reject	Include
P50	Total pressure at LPT outlet	Weak	Reject	Reject	Include

At this point only five sensors have been dropped, they are Nf, Nc, P2, P15 and P21. Examining the physical sensors to be included in the model, it can be observed there is a clear trend to which sensors are included when considered in context of the failure mode and thermodynamics.

The sensors to be included can briefly be summarised as the temperature and pressure at the outlet of all four stages: LPC, HPC, HPT and LPT as well as fuel flow. This isn't surprising, as when one stage in multi-stage device such as this is affected, the temperature and pressure of all adjacent or downstream stages will also be affected.

It is understood that the failure mode originates in the HPC with it described as a flow and efficiency degradation. As compressor efficiency is a measure of useful charge air pressure for a temperature rise over the compressor, a gradual decline in efficiency would exhibit as a gradual rise in temperature at the outlet, sensor T30. Visually there is a strong positive rise in temperature over the life of each unit.

Moving to the HPT we also observe an increase in outlet temperature this along with the increased turbine outlet temperature of the LPT, which is the fourth and final stage in the engine, can be attributed to two things. First is the increase in outlet temperature of the HPC, due to its efficiency degradation, being the inlet of the burner – so a higher intake temperature. Secondly, due to the increase on power requirements to achieve the required air flow demand from the HPC, the additional fuel drives the combustion temperature and thus the isentropic turbine power up through turbine inlet temperature and mass flow. With the HPT working at the same efficiency, it extracts work at the same rate, hence the same temperature delta across the machine and thus an increase in outlet temperature.

Moving backwards through the system to the LPC, this stage will potentially be affected primarily on flow as the HPC will be creating back-pressure on the LPC. With said back-pressure, the PR across the stage will be reduced as the inlet pressure will not be changing. This impact will be greatly reduced in comparison to the later stages though.

The fuel flow drift over time is explainable in that as the high-pressure compressor efficiency degrades, the two turbines must deliver more power to deliver the same air flow to the burner. As the two turbines efficiency is not degrading or increasing, to deliver more power, the turbines isentropic work must increase either through an increase in inlet temperature, mass flow or expansion ratio. To increase mass flow and inlet temperature, more fuel must be added as the burn temperature is inversely proportional to the air to fuel ratio. This is ultimately the result of a performance degradation of an engine, the fuel economy reduces due to the increased fuel demand for the same power.

A composite approach can be taken combining the visual examination, the median Rho value from the correlation, the median U-score, and the thermodynamic rationalisation. The predictions for feature importance can be reviewed in table 10, below.

Table 10: Initial predictions for feature importance in 9-parameter model.

Sensor	Description	Visual	Median U-Score	Median Rho-Score	Rank
T50	Total temperature at LPT outlet	Strong	0.0	-0.946	1
T48	Total temperature at HPT outlet	Strong	0.0	-0.910	2
Wf	Fuel flow	Strong	0.0	-0.877	3
T30	Total temperature at HPC outlet	Strong	0.0	-0.840	4
P50	Total pressure at LPT outlet	Weak	88.0	0.354	5
P40	Total pressure at burner outlet	Weak	107.0	0.549	6
Ps30	Static pressure at HPC outlet	Weak	107.5	0.547	7
P24	Total pressure at LPC outlet	Weak	4.0	-0.595	8
T24	Total temperature at LPC outlet	Weak	2.5	-0.527	9

For ranks 1, 2, 3 and 4 the median U-score combined with the median ρ scores were used to rank. From here, consideration was given to both parameters and considering where the temperature measurement counterpart might be for each pressure sensor, in the initial ranking. For example, T50 was ranked 1, hence P50 was ranked 5. The two sensors that were the most challenging to rank were P24 and T24, which are the total pressure and temperature at the LPC outlet, respectively. This is because the impact on this stage should be relatively low, in comparison to the other stages due to being upstream of the faulty stage with the others all being downstream. Despite this, the median U-score and ρ scores were favourable in it being ranked reasonably high, despite the line plot of the degradation being weak. Hence, P24 and T24 were ranked 8 and 9, respectively, with the ranking between them being decided using the median ρ score as well as the expectancy of the backpressure from the HPC driving the degradation being observed.

8. Prognostic Model Development

The dataset was split, with units 1-6 available for training and units 7-10 for testing, the training data was then split again taking unit 6 as the validation sample to reduce over-fitting to the training set. Due to the size of the dataset and to allow faster training and optimisation of the network, the cycles were resampled to 1/10th of their original size – reducing the memory requirements and computational overhead significantly. A "many to one" data structure and architecture was selected, as the problem required a model that could predict the number of cycles left until a specific unit fails.

As the model required a consistent input shape, the longest cycle was determined and all other cycles shorter than the longest cycler were zero padded to make all cycles equal length. Each cycle was then constructed as a single observation for the network making the network "many to one" with a training shape of (424, 2000, 13) – 424 observations of 2000 length cycles each with 13 parameters: 4 scenario descriptors and 9 physical sensors. The validation shape was (67, 2000, 13); as it only consisted of Unit 6.

Consideration was given to the target data for the algorithm, with respect to the training data having two distinct health states. A health state with minimal or normal degradation and a health state with abnormal degradation upon which the RUL was required to be estimated. Figure 17 illustrates how the target data is provided, which presents a continuous linear degradation from the beginning of the unit's life until failure. However, this does not match with reality or what the model will be trained on. Theoretically the unit could have a significantly high RUL whist in a healthy state; however, this RUL is unknown. Early model iterations appeared to pick up on this trend. Instead, a compromise was set for training the model which fixed the RUL during the healthy phase at the starting point of the unhealthy state as per, Figure 17. The constant degradation will be referred to as the "true RUL" with the consideration for health states being referred to as the "piece-wise RUL".

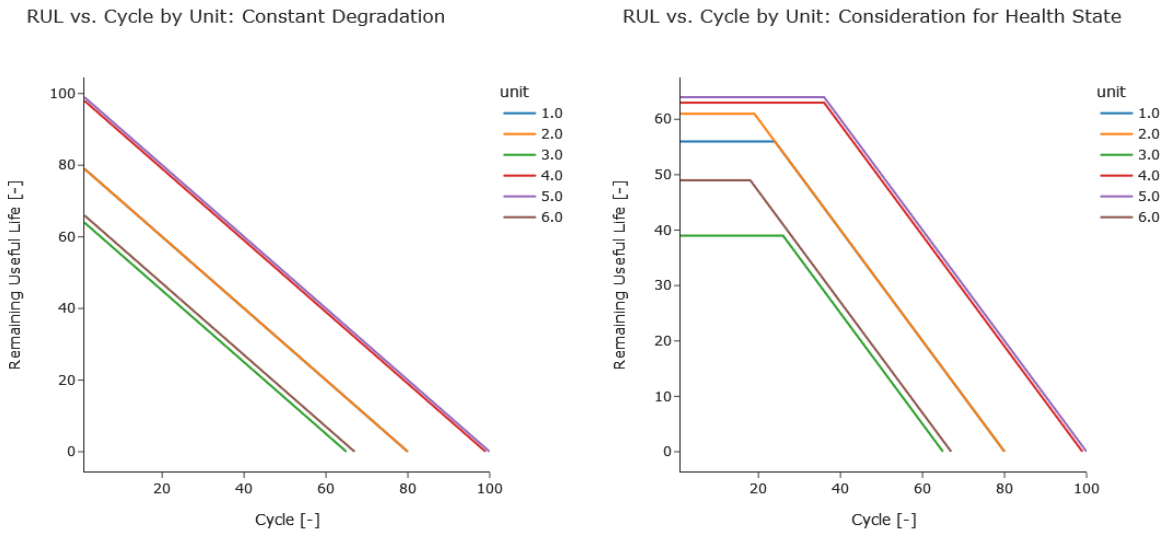


Figure 17: Target data with true target [left] and piecewise target [right] RUL.

A hyper model convolutional neural network was built using Keras (Chollet & others, 2015) as research had shown that convolutional neural networks (Thoppil et al., 2021) were powerful networks for timeseries predictions, with available configurations outlined in Table 11.

Table 11: Hyper-model configuration for 9-parameter model.

Layer	Type	Details
1	Conv1D	Number of filters: 2 to 64; step size 2 Kernel size: 2 or 4 Activation: ReLU
2	Conv1D	Number of filters: 512 to 2048; step size 8 Kernel size: 2 or 4 Activation: ReLU
3	MaxPooling1D	Pool size: 2
4	Dropout	Rate: 0.05 to 0.50; step size 0.01
5	Conv1D	Number of filters: 2 to 64; step size 2 Kernel size: 2 or 4 Activation: ReLU
6	MaxPooling1D	Pool size: 2
7	Dropout	Rate: 0.05 to 0.50; step size 0.01
8	Flatten	
9	Dense	Number of units: 2 to 64; step size 2
10	Dense	Number of units: 1

The Adam optimiser was used as it has historically shown good results with deep learning applications, specifically with CNNs (Li et al., 2018). The learning rate of the optimiser was also a hyper-parameter that could be tuned with a minimum value of $1x10^{-5}$ and a maximum of $1x10^{-3}$ with a step of $5x10^{-5}$. The loss metric was set as mean squared error [MSE].

The grid-search was configured to run a Bayesian Optimisation using Keras Tuner (O'Malley et al., 2019) with the tuners objective to minimise the MSE on the validation dataset [unit 6]. A fixed seed of 42 was used for repeatability and the optimiser was given 400 trials to tune the final model. The fit, or search method in this instance, ran to 250 epochs with a batch size of 8 to conserve memory and maximise performance.

Initial trials with hyper-models with much wider ranges of hyper-parameters were run, which included Boolean functions for the optimiser to set the number of layers within the hyper-model. It was quickly observed that the optimiser typically shifted to an initial convolutional layer with fewer filters with an immediate second convolutional layer with a much higher number of filters with a third and final convolutional layer after a max-pooling and dropout layer. Activation functions in the CNN were set as ReLU, which have historically been show to outperform sigmoidal activation functions in deep learning and regression tasks (Schmidhuber, 2015). After 400 optimisation runs, the model with the lowest MSE on the validation set was selected for scoring and further analysis.

A scatter plot of predicted RUL vs. ground truth RUL was created, the results can be examined in figure 18. It can be observed that a strong correlation exists, with an R^2 of 0.898 for units 1- 5. There is some scatter around mean line, particularly at higher RULs – this is due to the units being in healthy state at higher RULs.

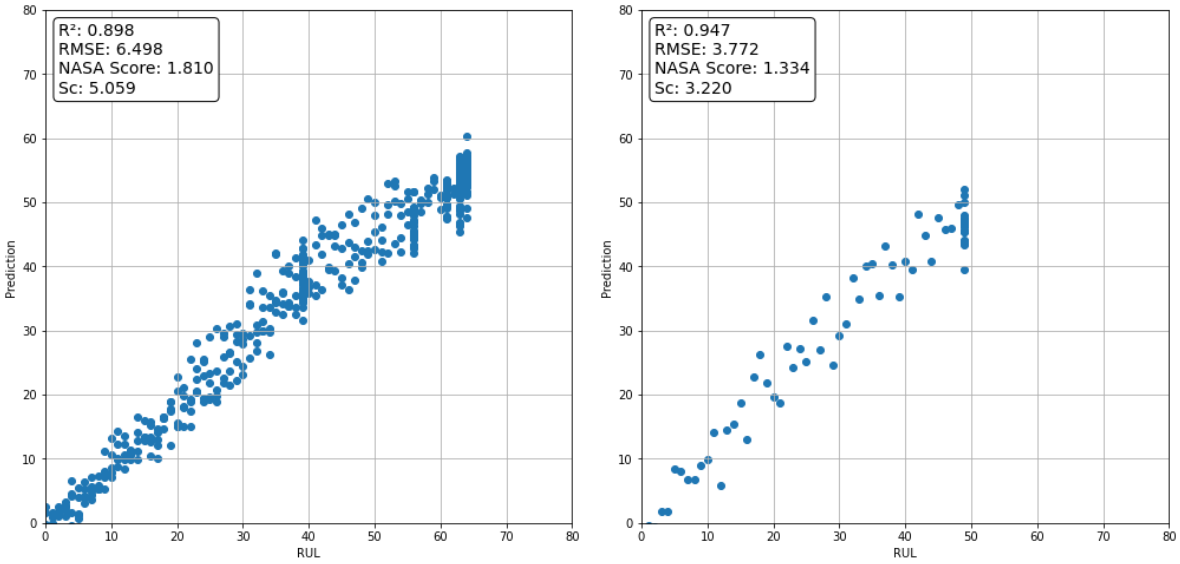


Figure 18: Scatter plot training [left] and validation [right] results for 9 parameter model.

The MSE for the training data was determined to be 6.498; but 3.772 for the validation set [unit 6]. The R^2 for the validation set was also higher, at 0.947 which indicates some slight overfitting towards the validation set but good generalisation overall.

Examining individual predictions, such as unit 1 in Figure 19, it is evident that the model is much more accurate in the unhealthy phase of the unit life in comparison to the healthy state, with RMSEs of 3.57 and 10.34 respectively. This trend continued for all six units in the training and validation datasets. It was noted that the model typically underpredicts, which is a useful bias caused by the modified target data to account for health state; as a prognostic the preference is to underpredict rather than overpredict and the NASA scoring function is configured to penalise underpredicting models to a lesser extent.

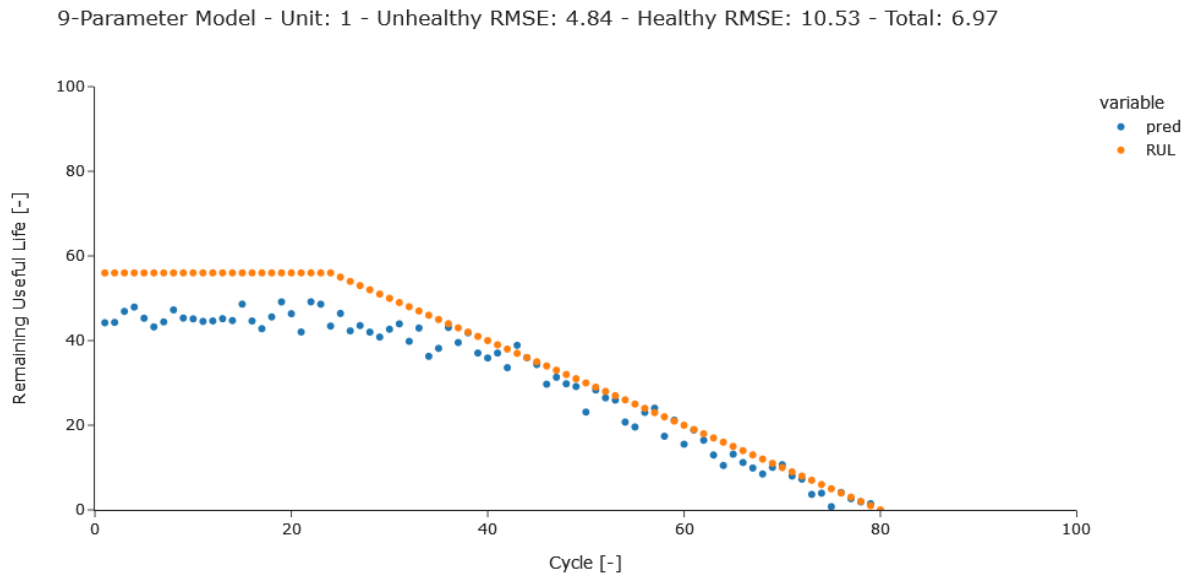


Figure 19: 9-Parameter model predictions versus ground truth RUL for Unit 1.

9. All Parameter Benchmark Model

The 14-parameter model, which would use all physical sensor measurements as predictors, was developed similarly to the 9-parameter model. The hyper-model was given much more range to operate in, to account for the higher number of predictor variables available. The hyper-model configuration can be observed in Table 12, illustrating the increased range in number of filters and kernel size.

The optimiser was given 400 model iterations to fine tune the hyper-parameters, again aiming to minimise the MSE. Figure 19 illustrates a scatter plot of predicted versus ground-truth RUL with a training R^2 of 0.875 and a MSE of 7.20. The optimisation runs also used a validation set to minimise against, which was again unit 6, the R^2 for the validation set was 0.946 and the MSE was 3.82 again indicating that there is some over-fitting towards the validation data.

Table 12: Hyper-model configuration for 14-parameter model.

Layer	Type	Details
		Number of filters: 2 to 2048 step size 2
1	Conv1D	Kernel size: 2, 3, 5, 7, 11, 13
		Activation: ReLU
		Number of filters: 512 to 2048 step size 2
2	Conv1D	Kernel size: 2, 3, 5, 7, 11, 13
		Activation: ReLU
3	MaxPooling1D	Pool size: 2
4	Dropout	Rate: 0.05 to 0.50; step size 0.01
		Number of filters: 2 to 64; step size 2
5	Conv1D	Kernel size: 2, 3, 5, 7, 11, 13
		Activation: ReLU
6	MaxPooling1D	Pool size: 2
7	Dropout	Rate: 0.05 to 0.50; step size 0.01
8	Flatten	
9	Dense	Number of units: 2 to 128; step size 2
10	Dense	Number of units: 1

An interesting observation with the 14-parameter model is the apparent “elbow” in Figure 20 between the predicted and ground-truth RUL, apparently showing the two different health states of the units.

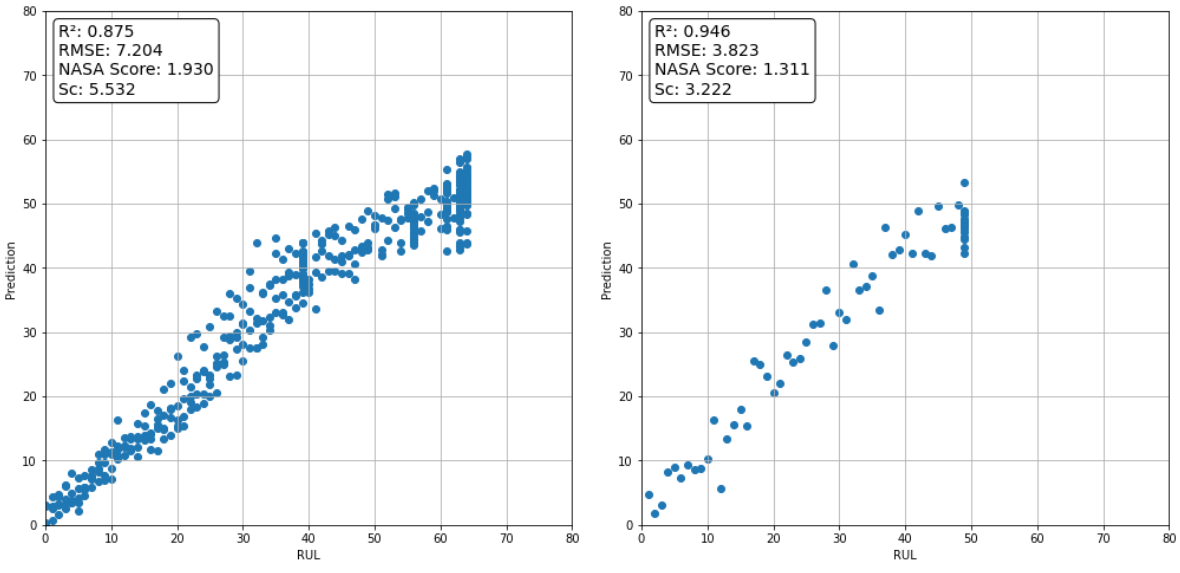


Figure 20: Scatter plot training [left] and validation [right] results for 14-parameter model.

10. Model Evaluation

10.1. Model Test Results

To quality check the models for good generalisation and as a formal test for the prognostic model, there were four units held back in the test data set for both the 9-parameter and the 14-parameter models. The test set consisted of units 7 – 10. A Mann-Whitney U test between training and test data sets RMSE scores were completed for both models, with results indicating that the models generalised well with p-values of 0.916 and 0.849 for the 9-Parameter and 14-Parameter models respectively.

All RMSE scores for each unit, both models and against both the piecewise and true target RUL have been tabulated in Table 13.

Table 13: RMSE results table for both models.

		9 Parameter			14 Parameter		
Target	Unit	Unhealthy	Healthy	Total	Unhealthy	Healthy	Total
Piecewise	Unit 1	4.84	10.53	6.97	3.57	9.81	6.06
Piecewise	Unit 2	4.94	10.23	6.52	4.60	9.97	6.23
Piecewise	Unit 3	2.88	3.00	2.92	4.54	2.51	3.88
Piecewise	Unit 4	5.13	10.49	7.48	5.69	12.76	8.86
Piecewise	Unit 5	4.20	9.83	6.73	5.14	12.53	8.49
Piecewise	Unit 6	3.73	3.90	3.77	4.01	3.22	3.82
Piecewise	Unit 7	8.86	21.00	13.29	7.39	18.91	11.69
Piecewise	Unit 8	2.86	4.50	3.58	3.29	6.79	4.94
Piecewise	Unit 9	3.09	5.35	3.81	3.58	4.92	3.98
Piecewise	Unit 10	4.93	5.11	5.01	5.11	4.07	4.68
True	Unit 1	4.84	23.53	13.26	3.57	22.63	12.50
True	Unit 2	4.94	20.27	10.55	4.60	20.03	10.33
True	Unit 3	2.88	15.37	9.80	4.54	13.05	8.84
True	Unit 4	5.13	29.68	18.12	5.69	32.10	19.63
True	Unit 5	4.20	28.88	17.42	5.14	31.46	19.07
True	Unit 6	3.73	12.56	7.10	4.01	12.18	7.04
True	Unit 7	8.86	33.60	19.05	7.39	31.53	17.57
True	Unit 8	2.86	23.15	15.53	3.29	25.44	15.99
True	Unit 9	3.09	14.72	7.97	3.58	14.48	8.01
True	Unit 10	4.93	17.71	12.30	5.11	19.16	13.26

Scatter plots of the ground truth RUL versus predicted RUL can be examined in Figure 21, for all 10 units including training, validation, and test versus the piecewise target and the true target for the 9-Parameter model.

Table 14: Mann-Whitney U test for 9-Parameter model versus 14-Parameter model.

Hs	U	P-Value	Conclusion
0 - Unhealthy	42.0	0.571	Do not reject null hypothesis
1 - Healthy	51.0	0.970	Do not reject null hypothesis
Combined	47.5	0.880	Do not reject null hypothesis

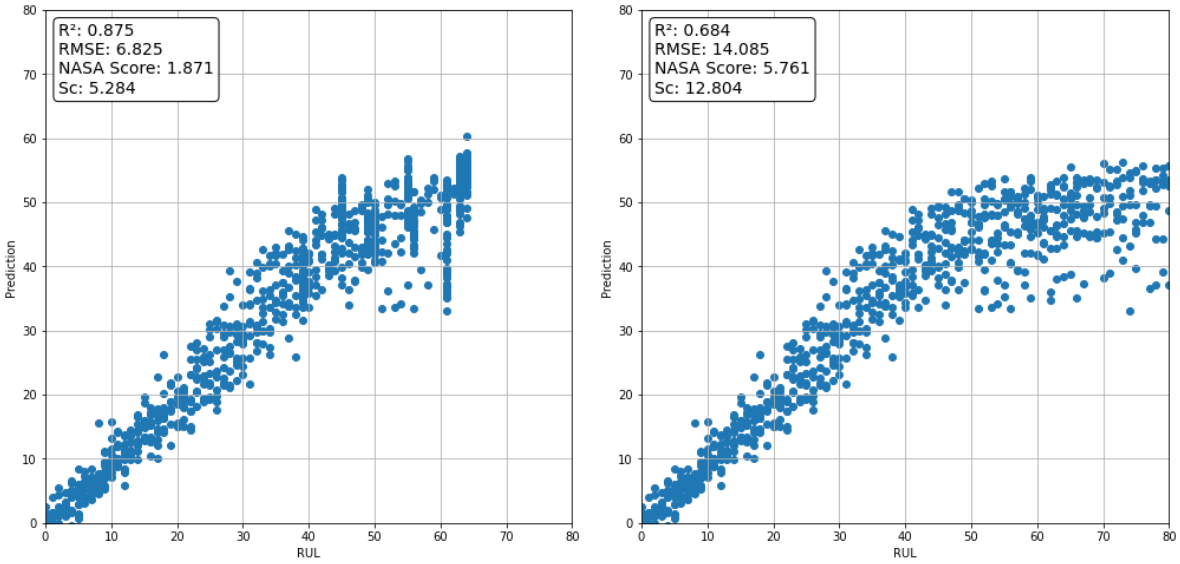


Figure 21: Scatter plot 9-parameter model versus piecewise [left] and true [right] targets.

The RMSE for both the 9-parameter and 14-parameter models were tabulated and split out between unhealthy and healthy, as well as scored against the piecewise and the true targets, in Table 14. This allowed the model performances to be understood in both health states. Table 13 was also translated into a boxplot, Figure 22.

It can easily be observed that the distributions for the two models are very similar, particularly during the healthy state. The 14-parameter model appears to have a slightly wider distribution during the unhealthy state with the outlier in the 9-parameter model coming into the distribution of the 14-paramter model. It should, however, be noted that the sample sizes here are small and should be interpreted with care.

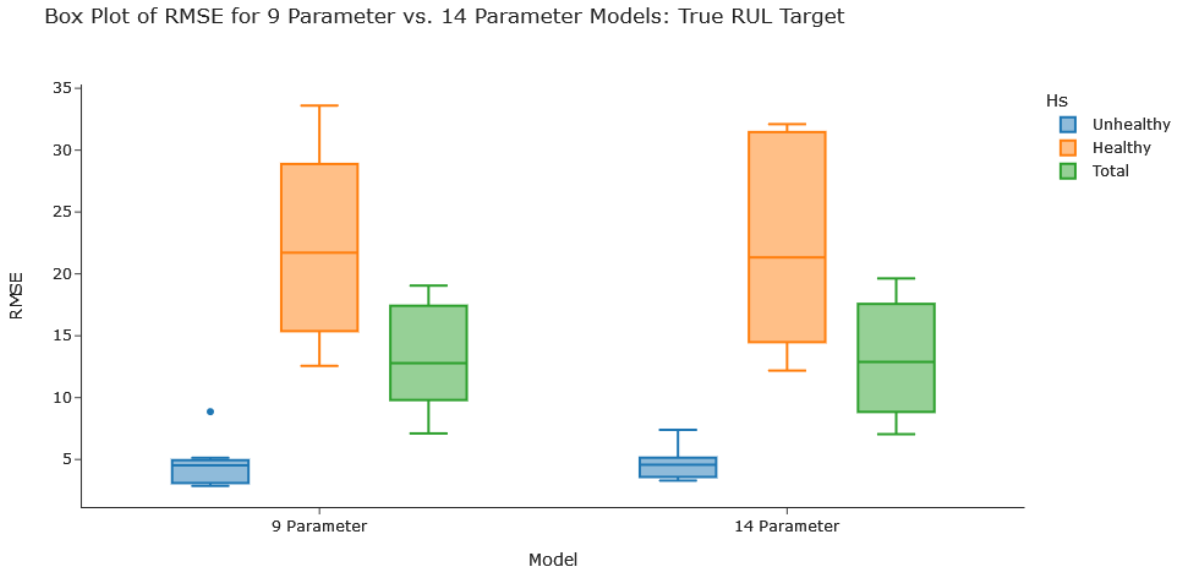


Figure 22: Boxplot of RMSE scores for both models against true RUL target.

10.2. Quality Comparison

A Mann-Whitney U test was completed between the two sets of predictions, the 9-parameter model and the 14-parameter model. The test statistic was 334,744 and the p-value was 0.985. As the null hypothesis could not be rejected, this must conclude that the two distributions are the same and thus the models are performing comparatively.

Using the RMSE and NASA Scoring functions [equations 1 and 2], the performance of the models can be assessed against both the true RUL and the piecewise RUL which was developed for training. The 9-Parameter model does appear to perform slightly higher on the combined Sc score [50/50 split between RMSE and NASA Scoring function], achieving 12.8 versus 13.6 as can be seen in Table 14.

Table 15: Model scoring results.

	9 Parameter			14 Parameter		
Target	RMSE	NASA	Sc	RMSE	NASA	Sc
True	14.085	5.761	12.804	14.513	6.370	13.626
Piecewise	6.825	1.871	5.284	6.967	1.871	5.354

As an external benchmark, the score for just the test set for FD01 to FD08 in the PHM 2021 competition, 2nd place achieved a score of 3.3 (Nathaniel DeVol, 2021) with 1st place achieving a score of 3.0 (Lövberg, 2021); both significantly lower and for a much larger dataset.

11. SHAP Analysis & Explain-ability

SHapely Additive exPlanations [SHAP] uses cooperative game theory to create Shapely values that can be used to understand which parameters, or players, in a game, or observation, are having the largest impact on the score, or estimation (Lundberg & Lee, 2017). SHAP values effectively show each predictors individual impact, from a starting mean point which is the same for all predictions, to the actual output prediction – which is the net of all parameter SHAP values. SHAP analysis is an incredibly powerful tool for understand the correlation of predictors to predictions and thus explaining what a model is doing, however it should be noted that as with all correlations it does not provide causation (Masis, 2021).

In this instance DeepExplainer will be used, which is an approximate algorithm for computing SHAP values on deep learning models such as the CNNs used in this research. To initialise the *explainer* a background dataset and model is required to be supplied, in these instances either the 14-parameter or 9-parameter models will be used with a random sample of 100 cycles from the training dataset.

With the explainer initialised, SHAP values for predictions of interest can be calculated and presented back to understand why a model makes a specific prediction. For example, the average feature importance can be determined by calculating the mean absolute SHAP values for a given series of predictions, in the cases below [see Figure 23] this is for all units and cycles with the 9-Parameter model.

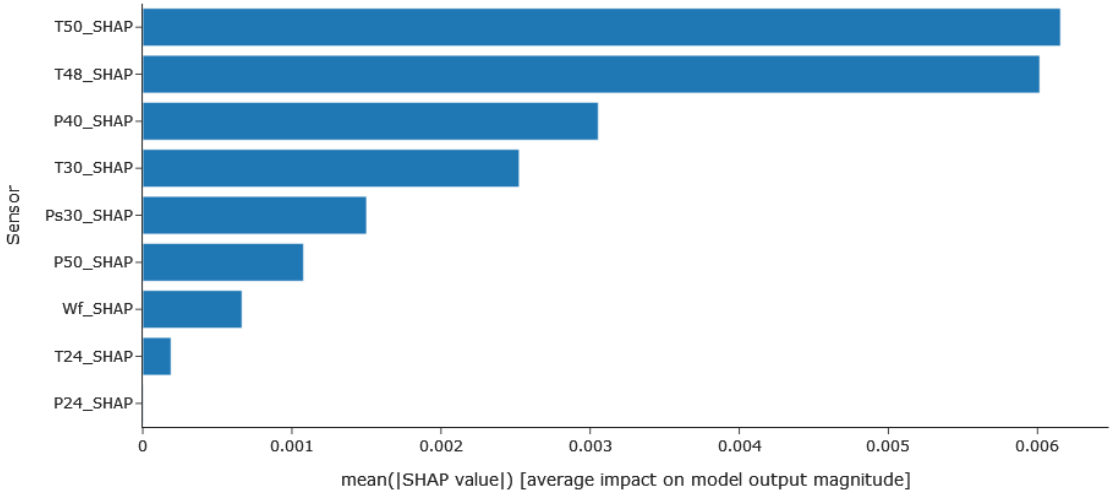


Figure 23: Mean SHAP values for physical sensors for all units and cycles with 9-Parameter model.

The first observation is the T50 and T48 are both ranked very highly, with a mean absolute SHAP value over double the magnitude of the next sensor, which is P40. It was not anticipated that P40 would be ranked 3rd but rather 6th. Another interesting parameter is Wf, which is ranked as 7th but was expected to be ranked 3rd. Finally, Ps30 was ranked 5th when it was anticipated to be 7th; all other rankings are line with expectations within 1 rank. A full break down of ranking can be viewed in Table 16.

Both P40 and Wf were investigated in more detail, to understand *why* the model chose to weight these two parameters the way it did. It is important to note that these are mean *absolute* SHAP values, so the direction of their impact on the prediction is not known through these charts alone.

Table 16: Feature importance ranking from SHAP analysis vs. expectation.

Sensor	Description	Predicted Rank	SHAP Rank
T50	Total temperature at LPT outlet	1	1
T48	Total temperature at HPT outlet	2	2
Wf	Fuel flow	3	7
T30	Total temperature at HPC outlet	4	4
P50	Total pressure at LPT outlet	5	6
P40	Total pressure at burner outlet	6	3
Ps30	Static pressure at HPC outlet	7	5
P24	Total pressure at LPC outlet	8	9
T24	Total temperature at LPC outlet	9	8

Examining the flight descriptors impact on the prediction, it can be observed that TRA has a strong influence with mean SHAP values above 0.006, which is greater than T50 in the sensor chart. This would be expected as the TRA is the main control over the engine with regards to fuel and exhaust gas temperatures and T50. Altitude, Mach number and T2, which is the total inlet temperature for the fan, have a low to moderate impact on the predictions, as can be observed in Figure 24.

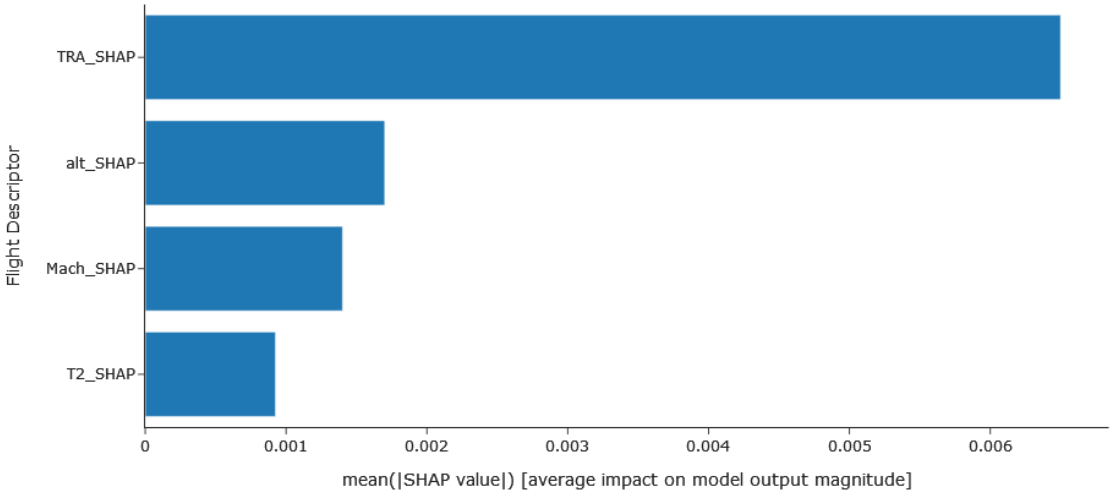


Figure 24: Mean SHAP values for flight descriptors for all units and cycles with 9-Parameter model.

Plotting the SHAP values for T50 and T24 against their measurement values with a colour axis of TRA, for unit 1, cycle 1 in Figure 25. It can clearly be observed that TRA correlates reasonably well with both T24 and T50. A key difference is the magnitude of the SHAP values for each sensor, with T50 having up to +0.4 and -1.0 cycles impact on the prediction during low TRA and mid-temperature, whereas T24 is having very little effect with most observations having no impact on the output prediction and the largest magnitudes being ±0.15 cycles.

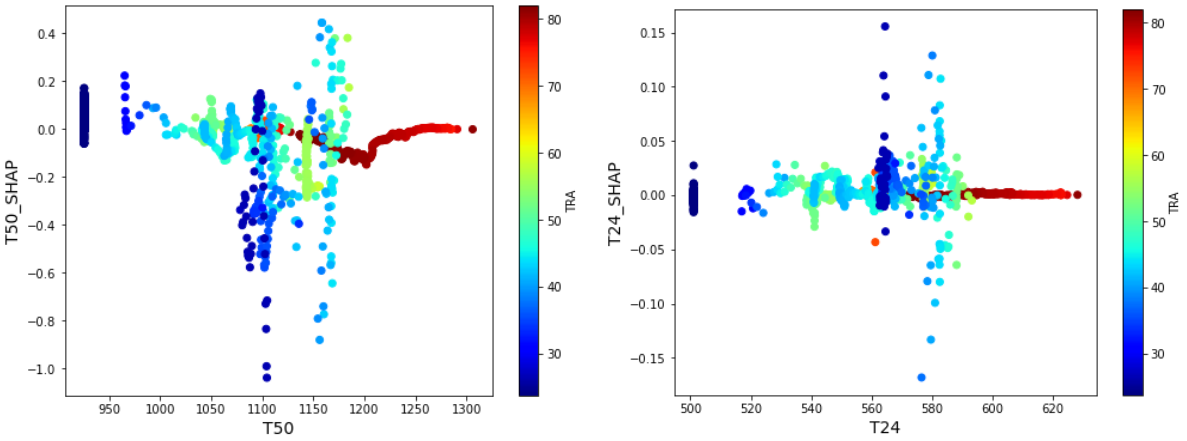


Figure 25: T50 Shap vs. T50 Values [left] and T24 Shap vs. T24 Values [right]. Unit 1, Cycle 1.

Plotting the cycle out against time with altitude overplotted [Figure 26], is it easy to see which portion of the cycle is having the largest influence on the prediction, the end of the cycle during descent. Note, the flat portion at 3000 ft minimum values is driven by the zero padding and inverse scaling. An interesting observation here is that the model still uses these zero values to influence the prediction. It can also be observed that T24 and T50 covary but to a different magnitude – which is why the algorithm has prioritised T50.

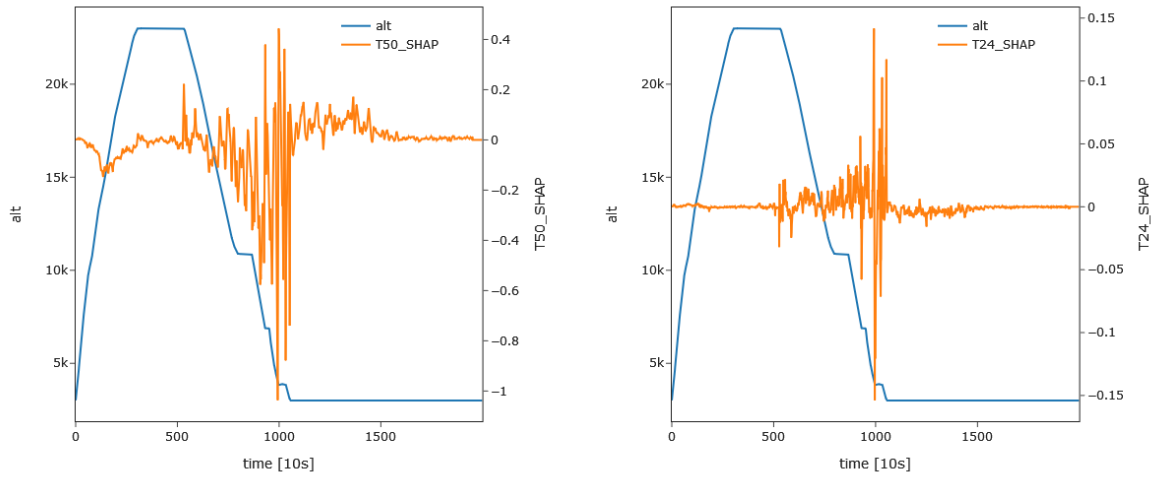


Figure 26: T24 and T24_SHAP [left] and T50 and T50_SHAP versus time. Unit 1, Cycle 1.

A Spearman ranked order correlation gives a ρ coefficient of 0.996 and a p-value of 0.000 between T24 and T50 for unit 1, cycle 1. With the two values being so tightly coupled and T50 having a much higher degree of monotonic degradation – the algorithm has picked up on this and chosen to minimise the impact of T24 on the prediction. The high weighting given to T48 can be explained by T48 also exhibiting a high degree of degradation and being slightly decoupled from T50 with a ρ coefficient of 0.973 and a p-value of 0.000, which drops to 0.947 at cycle 80 when the RUL is zero. Examining scatter plots of T24 and T48 against T50, Figure 27, it can be clearly seen the difference in the correlation between the pairings, all the higher SHAP values for T50 on the T48 versus T50 plot are clustered away from the underlying trend – whereas they're grouped together on the T24 versus T50 plot.

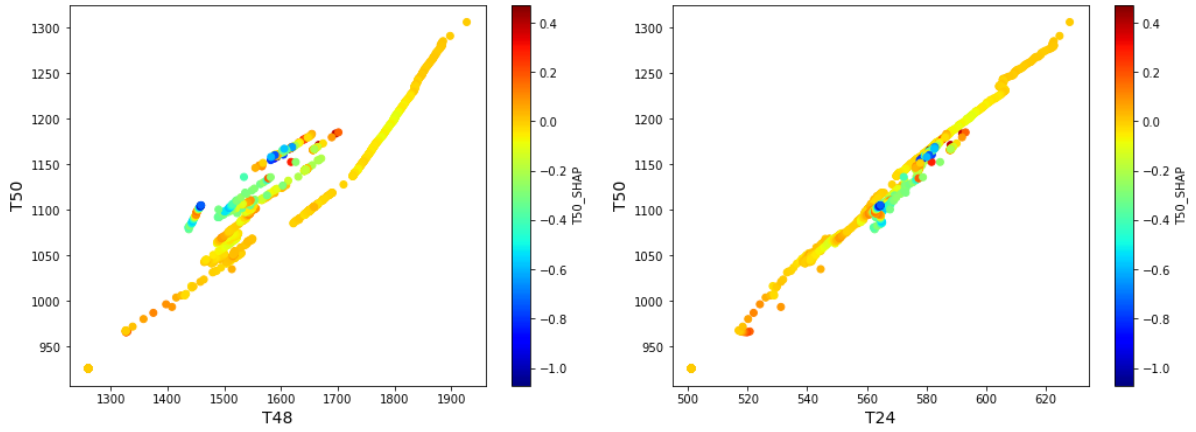


Figure 27: T48 [left] and T24 [right] versus T50 with colours of T50 SHAP values.

This same logic can be applied to the two sensors which stood out as not matching up with the predicted order of importance, Wf, which was expected to rank highly but in fact has very low SHAP values has a high correlation with T50 with a ρ coefficient of 0.998 and a p-value of 0.000 on unit 1, cycle 1. However, P40 which was ranked higher than anticipated provides new information, despite still being highly correlated with these stand out points having the greatest impact on the estimations made by the model, see Figure 28.

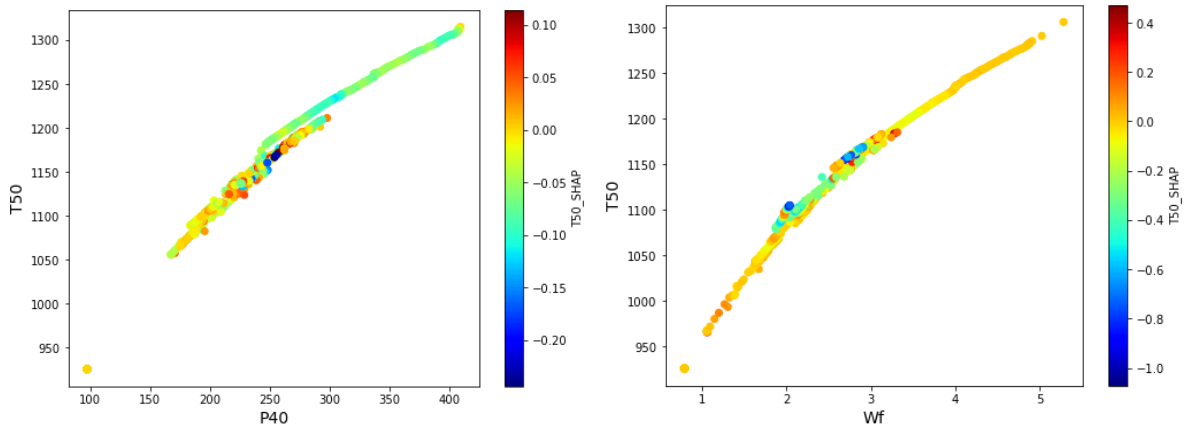


Figure 28: P40 [left] and Wf [right] versus T50 with colours of T50 SHAP values.

Selecting a cycle with a high prediction accuracy, i.e. during Hs = 0, and plotting the sum of the SHAP values by predictor on a waterfall plot provides a visual representation of the adders and subtractors to the base value in order to reach a prediction. An interesting observation here is that nearly all physical sensor predictors have a net negative effect from the base value of 32.34 down to the prediction of -0.28, except for P24. However, the impact P24 has is very small. Another interesting observation in this instance is that P50, T48 and T30 all have a greater impact on the prediction than T50 and that all scenario descriptors, except T2, have a net positive impact on the prediction.

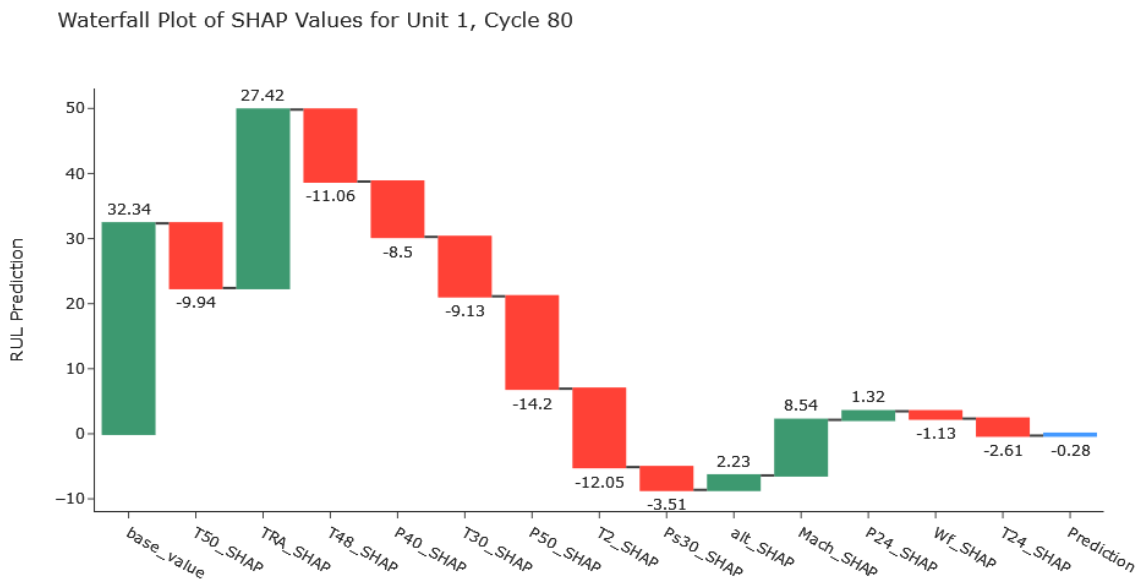


Figure 29: Waterfall plot of SHAP values for Unit 1, Cycle 80.

With the introduction of the 14-Parameter model, additional sensors that were not included in the 9-Paramter model feature highly on the mean absolute SHAP values, see Figure 30. There are significant similarities for rankings 1 – 6, but some slight changes with P2 and Nf being in 7th and 8th position. Both P2 and T50 were investigated in more detail to understand why they were ranked so highly.

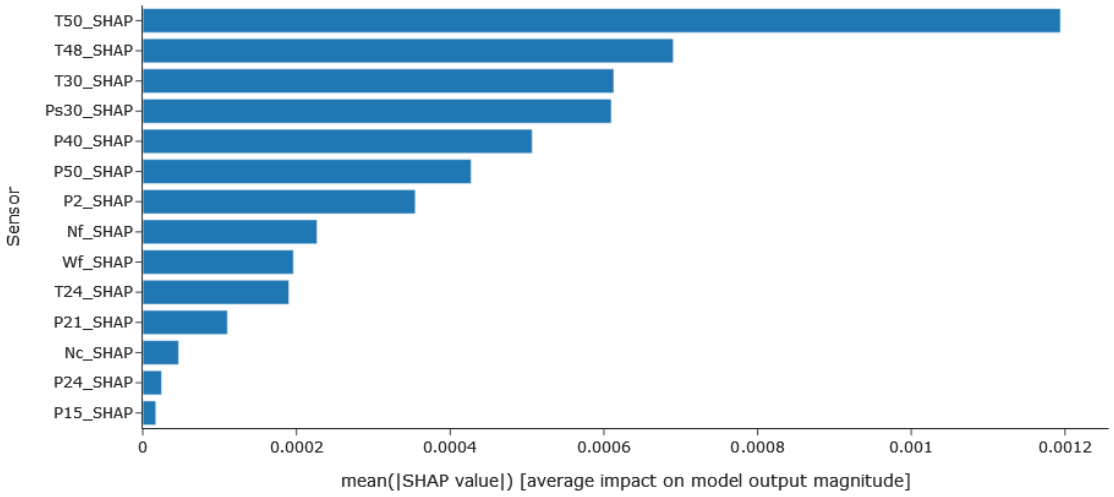


Figure 30: Mean SHAP values for physical sensors for all units and cycles for 14-Parameter model.

It can be seen this again has been driven by new information, not previously available to the model being captured within the sensor data. Examining the Nf and P2 against T50 and T50 SHAP values, it can be observed that there is significant disconnect between the two predictor pairs, with the greatest SHAP values generally sitting in a specific region – see Figure 31. Spearman correlation between Nf and T50 provided a ρ coefficient of 0.953 and p-value of 0.000 for unit 1, cycle 1 then

dropping to 0.931 and 0.000 respectively for cycle 80. Similarly, correlation for P2 and T50 provided ρ coefficients and p-values of 0.921 and 0.000, with 0.869 and 0.000 for unit 1, cycle 1 and cycle 80 respectively.

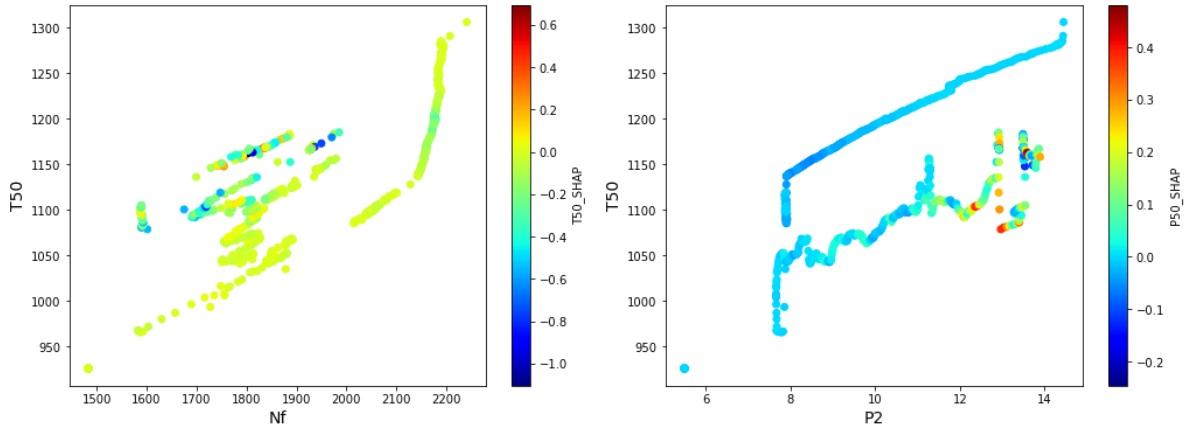


Figure 31: Nf [left] and P2 [right] versus T50 with colours of T50 SHAP.

Examining time plots for Nf overplotted with altitude, it can be observed that Nf has an impact all throughout the cycle, with peaks during ascent and decent with bulk of the negative impact during descent. However, on cycle 80 it can also be observed that the 3000 ft altitude flat line at the end of the cycle is having a significant effect on the SHAP values.

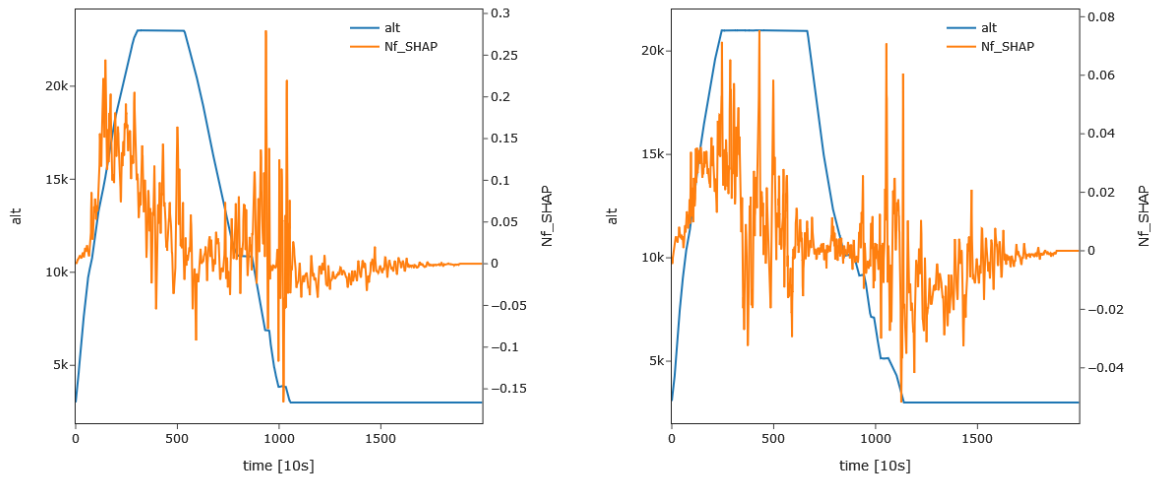


Figure 32: Time plot for Nf SHAP values for unit 1, cycle 1 [left] and cycle 80 [right].

The models are working by selecting the highest magnitude degradation sensors initially, such as T50, but then focussing on sensors which provide additional information not available through sensors which are closely correlated or covarying. Sensors that to have very high correlation have a much lower impact on the magnitude of the model's prediction.

12. Evaluation

The project has been successful in delivering a prognostic that is explainable. This was in part achieved through dimensionality reduction of the predictors and while the method provided clear insight into which predictors contained monotonic degradation, the method did miss the collinearity which only became apparent in the SHAP analysis. This is the biggest miss of the project but can be used going forward in future development.

While the prognostic and the all-parameter benchmark model had very similar scores, hence a similar level of quality – the actual algorithm was relatively straight forward in comparison to (Lövberg, 2021) or (Nathaniel DeVol, 2021), so part of the reason they perform similarly could be down to them both not being high enough quality to detect the lowest levels of patterns between predictors and target.

On reflection of the model evaluation, the piecewise RUL target appears to have hindered the final scores with a significant increase in RMSE during the healthy phase of the unit's life when scored against the true RUL.

The training, validation, and test splits for both the 9-parameter and 14-parameter models both indicated that the models were slightly over-fitted to the validation set but generalised well to the test set. While the results indicate a positive conclusion and agreement with the hypothesis, the models developed were all optimised by creating hyper-models and iterating over hyper-parameters to find a well generalised model; this was very computationally expensive.

The zero-padding combined with scaling has had an influence on the RUL estimations. It can be clearly seen in Figure 32 that, because of the zero padding and scaling, the period when the engine is not in operation is still impacting the RUL predictions; a better method would have been to scale from zero for each predictor and have a maximum value each.

13. Conclusions

1. It was shown in the background research that there has been limited focus on the interpretability and explain-ability of complex black-box algorithms such as CNNs, when applied to aircraft engine RUL predictions.
2. A physical sensor predictive model, using scenario descriptors as predictors, was trained based on the healthy condition of the units with a predictive error on test data below 0.35% MAPE.
3. The physical sensor models were able to clearly distinguish and illustrate sensors showing a high level of monotonic degradation. Allowing for a reduction in predictors with statistical and domain knowledge justification.
4. The 9-Parameter model performed on par with the 14-Paramter model, indicating that all crucial degradation information had been captured with the reduced predictor set.
5. Both the 9-Parameter model and the 14-Paramter model did not perform on par with the state-of-the-art from the background literature review.
6. SHAP analysis has shown there was further opportunity for parameter trimming, making the model less complex and improving the explain-ability further, by closely studying the collinearity of several parameters that did not provide the model new information and that other sensors not originally prioritised, such as Nf and P2, could potentially improve the accuracy of the model.

14. Further Work

An obvious follow on would be to include additional failure modes and perhaps expand to much larger datasets, rather than just DS05 that was worked on in this research – expand to include DS01 to DS08 which would provide 100 units for training. It would be interesting to understand if the predictor list could be reduced again. Alternatively, if all physical sensors were included, could different failure modes be identified using the SHAP analysis.

A more aggressive approach to parameter pruning through collinearity could have reduced the complexity and improved the explain-ability. Looking at collinearity between predictors, it may be possible to reduce to only a few sensor predictions.

Application of more advanced algorithms such as the inception algorithm applied by (Nathaniel DeVol, 2021), the dilated CNN applied by (Lövberg, 2021), or the hybrid CNN LSTM applied by (Kong et al., 2019), could provide more accurate result and better benchmark a reduced parameter versus full parameter model.

Increasing the data volume, through frequency, from 0.1Hz to 1Hz to quantify if the intra-cycle data holds additional value to complement the inter-cycle data and answer the question of quantifying the deficit in performance due to the sampling method employed.

15. Appendices [source code]

All code, models, and scalers available on GitHub:

https://github.com/CarlosTheJackal84/CMI7420-2022_U0370630/invitations

16. References

Chollet, F., & others. (2015, 2015). Keras. https://github.com/fchollet/keras

E. E. Halila, D. T. L., T. T. Thomas. (1982). *Energy efficient engine high pressure turbine test hardware detailed design report*.

El-Sayed, A. F. (2017). *Aircraft propulsion and gas turbine engines: Second edition*. https://doi.org/10.1201/9781315156743

Feldman, K., Jazouli, T., & Sandborn, P. A. (2009). A Methodology for Determining the Return on Investment Associated With Prognostics and Health Management. *IEEE transactions on reliability, 58*(2), 305-316. https://doi.org/10.1109/TR.2009.2020133

IEEE. (2017). IEEE Standard Framework for Prognostics and Health Management of Electronic Systems. *IEEE Std 1856-2017*, 1-31. https://doi.org/10.1109/IEEESTD.2017.8227036

Jansohn, P. (2013). *Modern gas turbine systems: high efficiency, low emission, fuel flexible power generation* (Vol. number 20;no. 20.;). Woodhead Pub. https://doi.org/10.1533/9780857096067

Khumprom, P., Grewell, D., & Yodo, N. (2020). Deep Neural Network Feature Selection Approaches for Data-Driven Prognostic Model of Aircraft Engines. *Aerospace, 7*(9), 132. https://doi.org/10.3390/aerospace7090132

Kong, Z., Cui, Y., Xia, Z., & Lv, H. (2019). Convolution and Long Short-Term Memory Hybrid Deep Neural Networks for Remaining Useful Life Prognostics. *Applied Sciences, 9*(19), 4156. https://www.mdpi.com/2076-3417/9/19/4156

Lattime, S., & Steinetz, B. (2003, 10/01). Turbine Engine Clearance Control Systems: Current Practices and Future Directions. https://doi.org/10.2514/6.2002-3790

Lei, Y., Li, N., Guo, L., Li, N., Yan, T., & Lin, J. (2018, 2018/05/01/). Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing, 104*, 799-834. https://doi.org/https://doi.org/10.1016/j.ymssp.2017.11.016

Li, X., Ding, Q., & Sun, J.-Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability engineering & system safety, 172*, 1-11. https://doi.org/10.1016/j.ress.2017.11.021

Liu, L., Wang, S., Liu, D., Zhang, Y., & Peng, Y. (2015). Entropy-based sensor selection for condition monitoring and prognostics of aircraft engine. *Microelectronics and reliability, 55*(9-10), 2092-2096. https://doi.org/10.1016/j.microrel.2015.06.076

Lövberg, A. (2021). *Remaining Useful Life Prediction of Aircraft Engines with Variable Length Input Sequences* Annual Conference of the PHM Society, 13,

Luna, J. J. (2021). *Metrics, Models, and Scenarios for Evaluating PHM Effects on Logistics Support* Annual Conference of the PHM Society, 1, https://papers.phmsociety.org/index.php/phmconf/article/view/1595

Lundberg, S., & Lee, S.-I. (2017). *A Unified Approach to Interpreting Model Predictions*.

M. Chao, C. K., K. Goebel and O. Fink. (2021). *Aircraft Engine Run-to-Failure Dataset under real flight condition*. https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#turbofan-2

Masis, S. (2021). *Interpretable Machine Learning with Python: Learn to build interpretable high-performance models with hands-on real-world examples*. Packt Publishing.

Mofokeng, T., Mativenga, P. T., & Marnewick, A. (2020, 2020/01/01/). Analysis of aircraft maintenance processes and cost. *Procedia CIRP, 90*, 467-472. https://doi.org/https://doi.org/10.1016/j.procir.2020.01.115

Nathaniel DeVol, C. S., Katherine Fu. (2021). Inception Based Deep Convolutional Neural Network for Remaining Useful Life Estimation of Turbofan Engines. Annual Conference of the PHM Society, 13,

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., & others. (2019). *KerasTuner*. https://github.com/keras-team/keras-tuner. https://github.com/keras-team/keras-tuner

PHM, S. (2021). *2021 PHM Conference Data Challenge*. https://data.phmsociety.org/2021-phm-conference-data-challenge/

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008, 6-9 Oct. 2008). Damage propagation modeling for aircraft engine run-to-failure simulation. 2008 International Conference on Prognostics and Health Management,

Schmidhuber, J. (2015, 2015/01/01/). Deep learning in neural networks: An overview. *Neural Networks, 61*, 85-117. https://doi.org/https://doi.org/10.1016/j.neunet.2014.09.003

Shih, T. I. P., & Yang, V. (2014). *Turbine aerodynamics, heat transfer, materials, and mechanics* (Vol. 243;243.;). American Institute of Aeronautics and Astronautics, Inc. https://doi.org/10.2514/4.102639

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016, 27-30 June 2016). Rethinking the Inception Architecture for Computer Vision. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),

Thoppil, N. M., Vasu, V., & Rao, C. S. P. (2021). Deep Learning Algorithms for Machinery Health Prognostics Using Time-Series Data: A Review. *Journal of Vibration Engineering & Technologies, 9*(6), 1123-1145. https://doi.org/10.1007/s42417-021-00286-x

Xu, J., Wang, Y., & Xu, L. (2014). PHM-Oriented Integrated Fusion Prognostics for Aircraft Engines Based on Sensor Data. *IEEE sensors journal, 14*(4), 1124-1132. https://doi.org/10.1109/JSEN.2013.2293517

Yuan, M., Wu, Y., & Lin, L. (2016, 10-12 Oct. 2016). Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network. 2016 IEEE International Conference on Aircraft Utility Systems (AUS),

Zhang, B., Zheng, K., Huang, Q., Feng, S., Zhou, S., & Zhang, Y. (2020). Aircraft Engine Prognostics Based on Informative Sensor Selection and Adaptive Degradation Modeling with Functional Principal Component Analysis. *Sensors (Basel, Switzerland), 20*(3), 920. https://doi.org/10.3390/s20030920