An Intelligent Toolkit for Benchmarking Data-Driven Aerospace Prognostics

Conference Paper · July 2019

DOI: 10.1109/ITSC.2019.8917115

4 authors:

Divish Rengasamy
University of Nottingham
6 PUBLICATIONS **23** CITATIONS
SEE PROFILE

Jimiama Mafeni Mase
University of Nottingham
9 PUBLICATIONS **19** CITATIONS
SEE PROFILE

Benjmain Rothwell
University of Nottingham
8 PUBLICATIONS **10** CITATIONS
SEE PROFILE

Grazziela P. Figueredo
University of Nottingham
90 PUBLICATIONS **262** CITATIONS
SEE PROFILE

An Intelligent Toolkit for Benchmarking Data-Driven Aerospace Prognostics

Divish Rengasamy[1], Jimiama M Mase[2], Benjamin Rothwell[3], Grazziela P. Figueredo[2,4]
[1]Institute for Aerospace Technology, The University of Nottingham, UK
[2]School of Computer Science, The University of Nottingham, UK
[3]Gas Turbine and Transmissions Research Centre, The University of Nottingham, UK
[4]The Advanced Data Analysis Centre, The University of Nottingham, UK
Email: {Divish.Rengasamy, Jimiama.MafeniMase1, Benjamin.Rothwell, Grazziela.Figueredo}@nottingham.ac.uk

***Abstract*— Machine Learning (ML) has been largely employed to sensor data for predicting the Remaining Useful Life (RUL) of aircraft components with promising results. A review of the literature, however, has revealed a lack of consensus regarding evaluation metrics adopted, the state-of-the-art methods employed for performance comparison, the approaches to address data overfitting, and statistical tests to assess results' significance. These weaknesses in methodological approaches to experimental design, results evaluation, comparison and reporting of findings can result in misleading outcomes and ultimately produce less effective predictors. Arbitrary choices of approaches for novel method's evaluation, the potential bias that can be introduced, and the lack of systematic replication and comparison of outcomes might affect the findings reported and misguide future research. For further advances in this area, there is therefore an urgent need for appropriate benchmarking methodologies to assist evaluating novel methods and to produce fair performance rankings. In this paper we introduce an open-source, extensible benchmarking library to address this gap in aerospace prognosis. The library will assist researchers to conduct a proper and fair evaluation of their novel ML RUL predictive models. In addition, it will assist stimulating better practices and a more rigorous experimental design approach across the field. Our library contains 13 state-of-the-art ML methods, 12 metrics for algorithm performance evaluation and tests for statistical significance. To demonstrate the library's functionalities, we apply it to gas turbine engine prognostic datasets.**

I. Introduction

Prognostics and Health Management (PHM) systems have become increasingly important in aviation. Aircraft are now fully equipped with sensors that constantly gather information regarding their status, and possible faults. The ability to utilise these sensor data to accurately predict problems in aircraft parts, facilitates their intelligent health and maintenance management. In addition, the widespread adoption of data collection in aircraft has allowed for the transition from Time-Based Maintenance (TBM) activities, where maintenance is scheduled under fixed intervals, to Condition-Based Maintenance (CBM), where decisions are based on information collected via sensor monitoring [1], [2]. CBM has enabled the rapid development of data-driven methods for aerospace maintenance and stimulated research of predicting when aircraft components will break.

There are several studies employing Machine Learning (ML) techniques to perform Remaining Useful Life (RUL) prediction for aircraft components using publicly available datasets [3]–[8]. Among most research reviewed, there is a lack of methodological agreement with regards to: (1) the evaluation metrics employed to assess their results, (2) the choice of the state-of-the-art methods for performance comparison, (3) the strategies chosen to address data overfitting, and (4) the adequate statistical methods for assessing results' significance. These existing weaknesses in methodological approaches can result in misleading outcomes and ultimately misguide future studies. Additionally, random choices of evaluation methods for novel approaches can introduce reporting bias in performance evaluation.

In order to address these methodological gaps, we conduct an in depth review of the evaluation methodologies used in data-driven aerospace prognosis and similar machine learning tasks, such as regression, and select state-of-the-art methods for evaluating and comparing performance. Subsequently, we develop a library in *Python* using open-source *Keras* and *Scikit-learn* libraries, which allows researchers to evaluate their novel methods using a systematic, robust, fair, and reproducible methodology. This library aims to achieve two main objectives: (1) to introduce an extensible, open-source data-driven toolkit for researchers, to encourage more systematic replication of data-driven prognosis models; and (2) to provide a robust methodology for evaluation and comparison of novel methods. In order to achieve the first objective, we implement 13 existing state-of-the-art data-driven prognosis algorithms and optimise their hyperparameters using random search and cross validation. For the second objective, we employ 12 evaluation metrics and statistical assessment of the outcomes.

This paper is organised as follows. Section II provides a review of the methodologies for evaluating and benchmarking ML prognosis algorithms and introduces the datasets used for validating our library. Section III provides an overview of the library, the methods implemented, and the results after applying the library on the datasets. Section IV concludes the paper and introduces opportunities for future research.

II. Literature Review

In order to better understand the rationale behind our toolkit and methodology we introduce an overview of the current efforts towards benchmarking RUL predictions and

the existing gaps in the literature. Subsequently, we identify the state-of-the-art ML prognosis methods and commonly used evaluation metrics. We also contrast the different performance evaluation approaches by different groups of authors. The objectives are (1) to draw attention to the lack of consensus regarding methods adopted; and (2) to establish a common set of well-known methods to be used by researchers for a more rigorous approach across the field in the future.

A. Remaining Useful Life Prognostics Benchmarking

Prognosis benchmarking has been an under-explored area for aerospace RUL prediction models. To the best of our knowledge, the main investigation towards advances in the area is introduced by Ramasso and Saxena [8]. The authors review and analyse an extensive list of studies employing intelligent prognosis methods to a well-known set of benchmark sensor data, i.e. Commercial Modular Aero-Propulsion System Simulation Datasets (CMAPSS). The authors also list the existing methods employed for the models' performance evaluation. The main objective of their study is to provide a clear guideline for using the dataset to ensure consistent comparison between different techniques. Their findings reveal inconsistencies in selecting performance evaluation metrics for results comparison among different authors. There is little literature, however, regarding methodologies for benchmarking novel prognosis models.

B. Benchmarking Datasets: Commercial Modular Aero-Propulsion System Simulation

The most widely used, publicly available data set for evaluating aerospace prognostic algorithms is CMAPSS [9]. It consists of four sub-datasets (Table I), established from a high fidelity simulation of a complex non-linear system that closely models a real aerospace engine. Each sub-dataset contains one training set and one test set with different operating conditions and fault patterns. The training set is the complete engine life cycle data, i.e. run to failure, but testing set cycles do not reach failure. The datasets consist of the engine unit number, the operating cycle number of each unit, the operating settings and the raw sensor measurements.

TABLE I: Description of the CMAPSS datasets

Datasets	# Engines in Training set	# Engines in Test set	# Training Samples	# Test Samples	Operating Conditions	Fault Modes
FD001	100	100	20,631	100	1	1
FD002	260	259	53,759	259	6	1
FD003	100	100	24,720	100	1	2
FD004	248	248	61,249	248	6	2

C. Current Methodologies in Aerospace Prognostic Algorithms Evaluation

Evaluation of prognostic algorithm involves: (1) selecting the different algorithms for performance comparison, and (2) selecting the metrics for result evaluation. Across three surveys of data-driven approaches for prognostics [8] [10] and [11], it can be seen that there is inconsistency in the selection of models for comparison and metrics for evaluation. In order to validate these findings, we review a broad list of studies employing data-driven prognostic methods and their preferred algorithms for model comparison and evaluation (Table II and III).

In Table II, the most commonly used state-of-the-art machine learning algorithms are Support Vector Regressor (SVR) [12], [13] , Random Forest (RF) [14], Deep Convolutional Neural Network (DCNN) [15], Long Short-Term Memory (LSTM) [6], [16], [17] and Neural Network (NN) [18]. We observe inconsistency in model comparison, similarly to the findings in [8] [10] and [11]. For instance, Yuan *et al.* [6] compare their proposed LSTM architecture for prognosis to only three Recurent Neural Network (RNN)-based algorithms while Hinchi *et al.* [17] compare their LSTM method to only Survival Analysis (SA). Similarly, Li *et al.* [15] compare DCNN for RUL prediction to four different NN architectures while Zhao *et al*. [18] compare their proposed NN architecture to only Discriminating Shapelet Extraction (DSE). Furthermore, Zaidan *et al*. [19] and Gao *et al.* [12] utilise Bayesian Hierarchical and SVR Models respectively for gas turbine engine prognostics but did not compare their methods to other models.

Furthermore, Table III presents the metrics used in the literature for model evaluation. The most commonly used metrics are Absolute Error (AE) [12], Relative Error (RE) [20], Root Mean Squared Error (RMSE) [21], Mean Error (ME) [13], Mean Squared Error (MSE) [21], timeliness [20], False Positives (FP) [22], Median Absolute Error (MdAE) [23], Mean Absolute Deviation (MAD) [23], symmetric Mean Absolute Percentage Error (sMAPE) [23], False Negatives (FN) [22], training time [23], and test time [23]. These metrics evaluate different aspects of performance and together they enable an in depth understanding of the model. The metrics are classified into three major categories: (1) algorithmic performance evaluation metrics, (2) computational performance metrics, and (3) cost-benefit performance metrics [23]. Algorithmic performance metrics evaluate the accuracy of the model in predicting RULs. Computational performance metrics evaluate the amount of time needed for the model to run, which is imperative for real-time monitoring and safety critical prognosis. Cost-benefit metrics are employed to evaluate the economic value of the model. We also observe disagreement in selecting these metrics. For instance, Yuan *et al.* [6], and Hinchi *et al.* [17] use LSTM for prognostics and employ RE and timeliness as evaluation metrics while Wang *et al.* [16] use Bidirectional LSTM but employ RMSE and timeliness as their choice of performance metrics. In addition, Gao *et al.* [12] use SVR with AE as the only evaluation metrics while Baptista *et al.* [13] also use SVR with ME, RMSE, MdAE and training time as their choice of evaluation metrics.

Current evaluation methodologies clearly show a lack of consensus regarding the evaluation metrics adopted and the state-of-the-art methods employed for performance comparison and evaluation (see Table II and Table III). In

TABLE II: Research employing data-driven algorithms for remaining useful life prediction and the algorithms which the studies compare their novel methods with. The ticks represent the algorithms which the studies compared their methods with. Zaidan *et al.* [19] and Gao *et al* [12] having no ticks (indicated with grey bar) illustrate that they did not compare their methods with any algorithm.

Research	**Method**	**Prognostic Algorithms Employed for Comparison**												
		GLM	KNN	XGB	RF	SVR	NN	RNN	LSTM	GRU	CNN2D	DNN	SA	DSE
Li *et al.* [15]	CNN						✓	✓			✓	✓		
Yuan *et al.* [6]	LSTM							✓	✓	✓				
Wang *et al.* [16]	BiLSTM					✓			✓		✓	✓		
Hu *et al.* [24]	Ensemble Learning					✓		✓						
Gao *et al.* [12]	SVR													
Baptista *et al.* [13]	SVR		✓	✓			✓							
Baptista *et al.* [14]	Kalman filter + (KNN, GLM, RF, NN, SVR)	✓			✓	✓	✓							
Zaidan *et al.* [19]	Bayesian Hierachical Model													
Zhao *et al.* [18]	NN													✓
Hinchi *et al.* [17]	LSTM												✓	

TABLE III: Research employing data-driven algorithms for remaining useful life prediction and the metrics used for evaluating the performance of their novel methods.

Research	**Metrics used for evaluating performance in the studies**								
	Timeliness	RE	ME	MAD	AE	MAE	MdAE	RMSE	Training Time
Li *et al.* [15]	✓							✓	
Yuan *et al.* [6]	✓	✓							
Wang *et al.* [16]	✓							✓	
Hu *et al.* [24]	✓								
Gao *et al.* [12]					✓				
Baptista *et al.* [13]			✓			✓		✓	✓
Baptista *et al.* [14]			✓	✓			✓		
Zaidan *et al.* [19]		✓							
Zhao *et al.* [18]	✓								
Hinchi *et al.* [17]	✓	✓							

addition, we observe the absence of strategies to address data overfitting in the literature and statistical tests on the RUL predictions for evaluating significant improvement of results. We therefore introduce an intelligent toolkit that allows users to compare different machine learning algorithms using a multitude of evaluation metrics and a significance test to reduce reporting bias.

III. The Intelligent Toolkit

In this section, we provide an overview of the toolkit's components and the performance results obtained after applying the toolkit on the CMAPSS datasets.

A. Overview Of Toolkit's Evaluation Methodology

Our toolkit is implemented in Python programming language using open-source *Keras* with *Tensorflow* backend and *Scikit-learn* libraries, which consists of high-level efficient ML and neural network functions for fast implementation of ML models. The toolkit has the *GNU General Public License v3.0* in Github [1], which enables researchers to freely contribute and use the toolkit. Figure 1 illustrates a flowchart of how the toolkit can be used to evaluate existing and novel prognosis models. The toolkit first automatically checks if a new model and dataset is defined by user. If they are defined, the toolkit compares the new model with the existing built-in algorithms (defined in Section III-B) on the new dataset and evaluates its performance (using the metrics define in Section III-C). Subsequently, Mann-Whitney-Wilcoxon non-parametric test evaluates the statistical significance of the

[1]Our library is available at https://github.com/divishrengasamy/intelligent-toolkit-prognostic

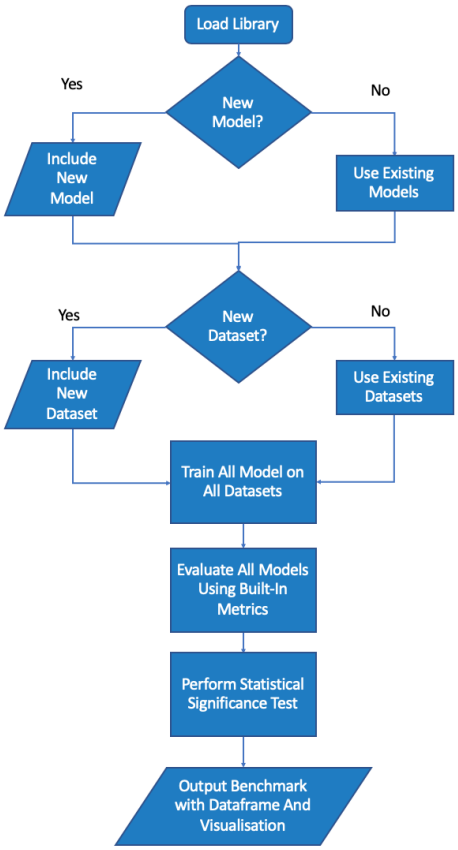

Fig. 1: Flowchart to benchmark the performance of prognostic algorithms

results by calculating the pairwise p-values of the algorithms. Finally, results and p-values are visualised using tables along with graphs such as box-plots and heatmaps.

B. Machine Learning Algorithms

The library consists of 13 machine learning algorithms, along with their respective hyperparameters for all CMAPSS datasets. The values of optimized hyperparameters can be found on our Github [2]. These data-driven algorithms consist of linear (SGD), kernel (SVR), tree (ET, RF, Boosting, GBR, Adaboost) and deep neural network (DNN, CNN, LSTM, GRU) models. We choose these algorithms as they are among the most widely used machine learning methods in the intelligent prognostic community [7], [25], [8]. To reduce overfitting of these algorithms, we optimise their hyperparameters using random search and 10-folds cross validation. Researchers using our toolkit are required to optimise the hyperparameters of their models to reduce overfitting and evaluate their algorithms with the optimised models in the toolkit.

Furthermore, to illustrate the effectiveness of random search optimisation, we examine the validation loss of CNN with and without optimisation (i.e. Figure 2). With no optimisation (purple line), we can observe an upward trend in validation loss for CNN after approximately 20 epochs that continues throughout the training process which indicates overfitting. Subsequently, after using optimisation (cyan line), there is a gradual decrease in validation loss across the epochs showing no sign of overfitting.

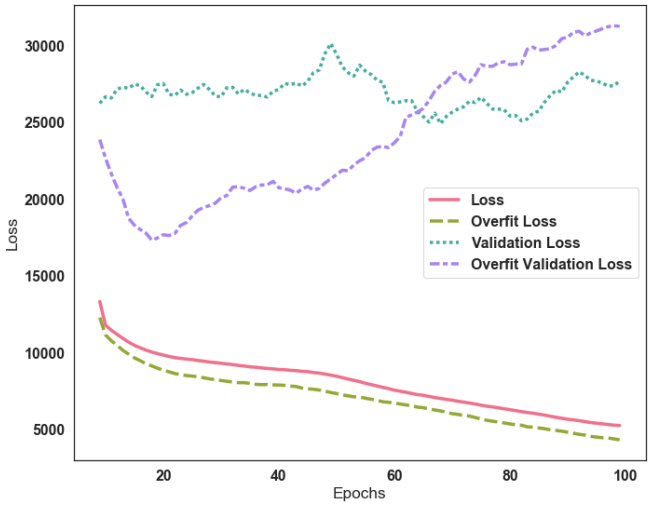

Fig. 2: Reduction of overfitting in CNN after optimisation shown using validation loss. The purple line is the validation loss before optimisation while the cyan line is the validation loss after optimisation is applied. The red and green lines represent the training loss before and after optimisation respectively.

C. Performance Evaluation Metrics

For performance evaluation, we implement coefficient of determination (R^2), Absolute Error (AE), Relative Error (RE), Root Mean Squared Error (RMSE), Mean Error (ME), Mean Squared Error (MSE), Timeliness, Median Absolute Error (MdAE), Mean Absolute Deviation (MAD), symmetric Mean Absolute Percentage Error (sMAPE), Training Time, and Test Time. These metrics are among the most commonly used algorithmic and computational performance evaluation metrics for regression problems. They are therefore useful for the prediction of RUL [7], [25], [8].

D. Results And Outputs From Intelligent Toolkit

We apply the 13 optimised ML algorithms from Section III-B to all CMAPSS datasets. Table IV shows the evaluation of models' performance on CMAPSS FD001 dataset with the emboldened values representing the best performing algorithm for each metric (the performance evaluation results of the toolkit on CMAPSS FD002, FD003 and FD004 datasets are found in our Github[3] due to page limitation). We observe some disagreement as to which algorithms perform the best. For instance, CNN1D performs the best on timeliness, MAE and R^2 metrics, while GRU in RE, MAD, AE, MdAE and RMSE. In addition, CNN2D

[2]Hyperparameter is available at https://github.com/divishrengasamy/intelligent-toolkit-prognostic

[3]Supplementary results: https://github.com/divishrengasamy/intelligent-toolkit-prognostic

TABLE IV: Results of the library using CMAPSS Dataset 1

Algorithms	**Metrics used for evaluating performance in the studies**											
	RE	ME	MAD	AE	MdAE	Timeliness	MAE	RMSE	R^2	sMAPE (%)	Training Time (s)	Testing Time (s)
SGD	61.8	25.2	20.5	2520.2	22.2	2477.9	25.2	30.4	0.464	40.650	**0.02**	**0.009**
Extra Trees	31.8	19.2	17.6	1924.4	11.7	1540.4	19.3	25.9	0.621	25.16	13.3	0.678
AdaBoost	41.9	21.7	19.0	2166.4	16.6	2050.9	21.8	28.4	0.530	31.185	32.7	0.023
Bagging	48.3	21.3	17.6	2214.2	18.1	1433.5	21.7	26.8	0.559	34.516	1.2	0.030
RF	31.8	19.0	18.0	1918.4	12.1	1672.5	18.9	25.7	0.609	24.795	37.0	0.413
SVR	36.5	19.7	17.5	1970.6	13.3	1877.6	19.7	25.5	0.622	29.472	20.0	0.678
GBR	31.9	19.4	19.0	1944.8	13.9	1912.7	19.4	26.7	0.586	26.009	9.2	0.011
KNN	33.1	20.7	19.1	2073.1	14.5	2030.9	20.7	27.7	0.553	26.511	0.2	0.08
DNN	30.7	**5.2**	17.4	1762.3	13.2	1221.4	14.7	21.6	0.72	31.2	1051	0.048
GRU	**17.3**	8.2	**13.3**	**1268.5**	**8.5**	999.5	11.8	**17.7**	0.61	27.7	2625	0.071
CNN2D	23.9	7.5	15.2	1550.3	8.6	957.3	14.2	21.2	0.67	**19.8**	1725	0.238
CNN1D	23.9	7.8	17.1	1616.6	9.9	**890.2**	**11.4**	18.0	**0.78**	22.6	262	0.231
LSTM	32.6	21.9	19.9	2228.9	15.2	1147.4	14.6	22.2	0.34	22.6	402	0.084

achieves the best result with sMAPE metric while DNN leads in ME. This disagreement is due to the fact that the metrics evaluate different facets of performance such as accuracy, precision, robustness and timeliness [23], and therefore, are all required for a better understanding of model performance.

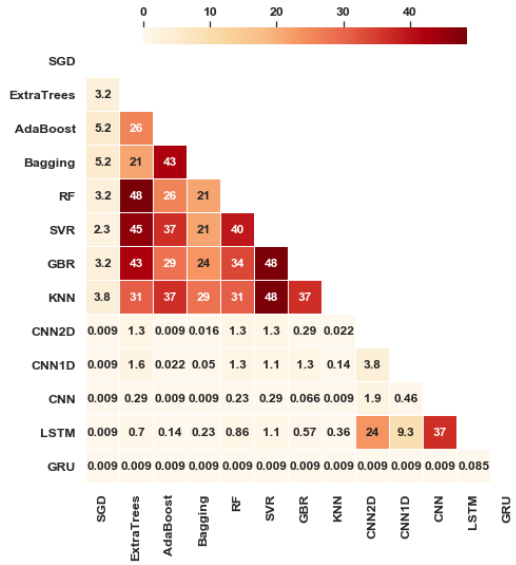

Fig. 3: A heat map of p-value (%) for pairwise comparison of MAE results of the 13 algorithms using Mann-Whitney-Wilcoxon non-parametric t-test.

Generally, evaluation metrics do not tell us if improvement in results is significant or not relative to other methods due to uncertainties from sensor readings and the combination of different sensors [26]. Therefore, using the Mann-Whitney-Wilcoxon non-parametric test [27] at a 5% significance level we can determine the statistical significance of the result of one algorithm compared to the others. We choose Mann-

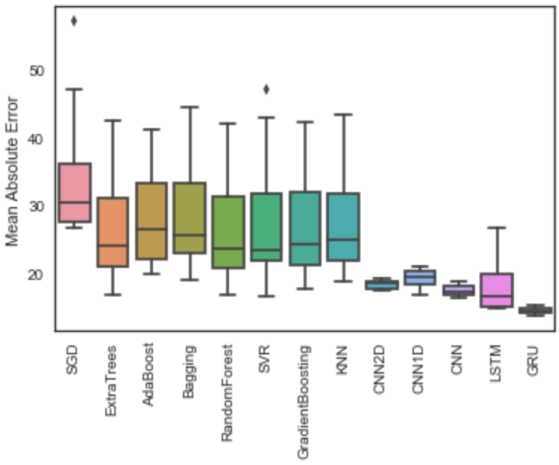

Fig. 4: Box-plots showing variability of MAE for each algorithm in toolkit after 10-fold cross validation.

Whitney-Wilcoxon non-parametric test because it does not assume normality of results' distributions. In Figure 3, we present a heat map of p-values (%) for pairwise comparison of the MAE results of the 13 algorithms applied on CMAPSS FD001 dataset. The heatmap clearly shows that there is no statistical improvement of performance among Extra Trees, Adaboost Regressor, Bagging Regressor, RF, SVR, GBR and KNN. Similarly, the performance of LSTM is not statistically different from CNN using the MAE metrics. In addition, Figure 4 displays the box-plots of the algorithms' MAE results for 10-fold cross validation to show the variability of model performance. We observe high degree of variability for all models except for CNN and GRU, and outliers are present in both SGD and SVR. This high variability in validation score for SGD, SVR, ET, RF, Boosting, GBR, Adaboost and LSTM indicates that the models poorly capture

the degradation process from the sensor measurements.

This toolkit enables researchers to benchmark their novel method to other optimised machine learning algorithms by producing a table with results using a wide variety of evaluation metrics and a heatmap illustrating statistical significance of results. These outputs (i.e. table and heatmap) provide a better understanding of the performance of researchers' novel methods in comparison to existing state-of-the-art data-driven methods to further research in the area. However, there are some limitations to this toolkit. The toolkit does not support automatic preprocessing and hyperparameters optimisation i.e., the toolkit uses default hyperparameters for new models.

IV. Conclusion and Future Work

In this paper we have developed an intelligent toolkit that allows researchers to evaluate their novel methods using a systematic, robust, fair, reproducible methodology. The toolkit is aimed at achieving two main objectives: (1) introducing an extensible, open-source data-driven toolkit for researchers, to encourage more systematic replication of data-driven prognostic models; and (2) provide a robust methodology for evaluation and comparison of novel methods. We implemented 13 existing state-of-the-art machine learning models for prognosis, 12 evaluation metrics and statistical assessment for a more robust evaluation and comparison of the models. Subsequently, we validated our toolkit by applying it to the four CMAPSS datasets. The results show an advantage in utilising diverse evaluation metrics as there is variability in the performance of algorithms across different metrics. Thus, the wide variety of evaluation metrics and data-driven prognostic algorithms in our toolkit provide a deeper understanding of the performance of novel models in predicting the remaining useful life of a component. For future work, we consider extending the library to include fault and anomaly detection benchmarking to provide a better overview machine health monitoring. Finally, we intend to apply our toolkit on additional prognostic datasets from other domains to test its robustness.

Acknowledgement

This work is funded by the INNOVATIVE doctoral programme. The INNOVATIVE programme is partially funded by the Marie Curie Initial Training Networks (ITN) action (project number 665468) and partially by the Institute for Aerospace Technology (IAT) at the University of Nottingham.

References

[1] R. Ahmad and S. Kamaruddin, "An overview of time-based and condition-based maintenance in industrial application," *Computers & Industrial Engineering*, vol. 63, no. 1, pp. 135–149, 2012.

[2] J. Kim, Y. Ahn, and H. Yeo, "A comparative study of time-based maintenance and condition-based maintenance for optimal choice of maintenance policy," *Structure and Infrastructure Engineering*, vol. 12, no. 12, pp. 1525–1536, 2016.

[3] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in *2008 international conference on prognostics and health management*. IEEE, 2008, pp. 1–6.

[4] T. Wang, "Trajectory similarity based prediction for remaining useful life estimation," Ph.D. dissertation, University of Cincinnati, 2010.

[5] P. Yu, X. Yong, L. Datong, and P. Xiyuan, "Sensor selection with grey correlation analysis for remaining useful life evaluation," 2012.

[6] M. Yuan *et al.*, "Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network," *IEEE Int Conf on Aircraft Utility Systems*, pp. 135–140, 10 2016.

[7] D. Rengasamy, H. P. Morvan, and G. P. Figueredo, "Deep learning approaches to aircraft maintenance, repair and overhaul: a review," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 150–156.

[8] E. Ramasso and A. Saxena, "Performance benchmarking and analysis of prognostic methods for cmapss datasets." *International Journal of Prognostics and Health Management*, vol. 5, no. 2, pp. 1–15, 2014.

[9] A. Saxena and K. Goebel, "C-mapss data set," *NASA Ames Prognostics Data Repository*, 2008.

[10] M. Schwabacher, "A survey of data-driven prognostics," in *Infotech@ Aerospace*, 2005, p. 7002.

[11] M. Schwabacher and K. Goebel, "A survey of artificial intelligence for prognostics," in *Aaai fall symposium*, 2007, pp. 107–114.

[12] Z. Gao, C. Ma, and Y. Luo, "Rul prediction for ima based on deep regression method," in *2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA)*, Nov 2017, pp. 25–31.

[13] M. Baptista, I. P. de Medeiros, J. P. Malere, H. Prendinger, C. L. Nascimento Jr, and E. Henriques, "A comparison of data-driven techniques for engine bleed valve prognostics using aircraft-derived fault messages," in *Third European Conference of the Prognostics and Health Management Society*, 2016.

[14] M. Baptista, E. M. Henriques, I. P. de Medeiros, J. P. Malere, C. L. Nascimento Jr, and H. Prendinger, "Remaining useful life estimation in aeronautics: Combining data-driven and kalman filtering," *Reliability Engineering & System Safety*, vol. 184, pp. 228–239, 2019.

[15] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[16] J. Wang, G. Wen, S. Yang, and Y. Liu, "Remaining useful life estimation in prognostics using deep bidirectional lstm neural network," in *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*. IEEE, 2018, pp. 1037–1042.

[17] A. Z. Hinchi and M. Tkiouat, "Rolling element bearing remaining useful life estimation based on a convolutional long-short-term memory network," *Procedia Computer Science*, vol. 127, pp. 123 – 132, 2018, proceedings Of The First International Conference On Intelligent Computing In Data Sciences, ICDS2017.

[18] Z. Zhao, B. Liang, X. Wang, and W. Lu, "Remaining useful life prediction of aircraft engine based on degradation pattern learning," *Reliability Engineering & System Safety*, vol. 164, pp. 74 – 83, 2017.

[19] M. A. Zaidan, A. R. Mills, R. F. Harrison, and P. J. Fleming, "Gas turbine engine prognostics using bayesian hierarchical models: A variational approach," *Mechanical Systems and Signal Processing*, vol. 70-71, pp. 120 – 140, 2016.

[20] G. J. Vachtsevanos, F. Lewis, A. Hess, and B. Wu, *Intelligent fault diagnosis and prognosis for engineering systems*. Wiley Hoboken, 2006, vol. 456.

[21] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[22] K. Goebel and P. Bonissone, "Prognostic information fusion for constant load systems," in *2005 7th International Conference on Information Fusion*, vol. 2. IEEE, 2005, pp. 9–pp.

[23] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," in *2008 International Conference on Prognostics and Health Management*. IEEE, 2008, pp. 1–17.

[24] C. Hu, B. D. Youn, P. Wang, and J. T. Yoon, "Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life," *Reliability Engineering & System Safety*, vol. 103, pp. 120 – 135, 2012.

[25] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing*, vol. 107, pp. 241 – 265, 2018.

[26] R. Cheng and S. Prabhakar, "Managing uncertainty in sensor database," *ACM SIGMOD Record*, vol. 32, no. 4, pp. 41–46, 2003.

[27] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947.