Delft University of Technology

Air Transport and Operations
AE4465

Assignment:
eXplainable Artificial Intelligence (XAI) for the Maintenance of a Turbofan Engine

Lecturer:
Marcia L. Baptista
Delivery Date:
30th of June 2024

June 3, 2024

Motivation

The student is requested to search and implement a methodology for post-processing the decisions/estimations of machine learning models used in the prognostics of a turbofan engine. Examples of methods are t-SNE, SHAP, or LIME. At least one prognostics models should be implemented in the CMAPSS case study [Saxena et al., 2008] to generate the estimations for the XAI methods. A final report is expected. The student should also deliver the Jupyter notebook in addition to the report. The absence of a Jupyter notebook results in less 1 point out of 10. You can deliver the notebook after the deadline (30th of June) with a penalty of 0.5 for each delay day.

Background

Deep learning applications have gathered much attention recently as they have shown superior performance in diverse fields from image and speech recognition to medical systems. However, these applications lack explainability and reliability. The term eXplainable Artificial Intelligence (XAI) was first mentioned by D. Gunning in [Gunning et al., 2019] as follows:

> "XAI will create a suite of machine learning techniques that enables human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners."

The recent increase in publications in the field of XAI [Speith, 2022] reveals the growing interest in this area. There is an expanding need for techniques that can be used to explore and reveal the inner workings of classical machine learning (ML) methods, such as deep neural networks (DNNs) and random forests. This is especially true for aviation and maintenance, where certification depends on understanding and trusting the model forecasts and predictions [Solís-Martín et al., 2023]. In Figure 2, we see a schema of the problem. The machine learning model produces obscure prognostic predictions, not being possible to interpret the model due to its black-box nature. XAI aims to address this kind of problem. We hereafter describe the three categories of XAI interpretability: pre-model, in-model and pos-model.

Some authors, such as Carvalho et al. [2019], consider the existence of pre-model, or data, interpretability. This kind of interpretability involves applying independent techniques to understand the data used to build the model. These approaches depend solely on the data being therefore model-agnostic. Principal Component Analysis (PCA), Distributed Stochastic Neighbor Embedding (t-SNE), and clustering methods are examples of exploratory data analysis methods [Tukey et al., 1977] that can be put under pre-model interpretability. Often, these techniques do not have a high interpretability power but are still considered [Carvalho et al., 2019, Arrieta et al., 2020, Tjoa and Guan, 2020] to be part of the XAI field. This follows from these techniques promoting a better understanding of the model and aiding experts to understand and gain insights into the prognostics process. They can also work with more advanced techniques to provide a more holistic overview of the model.

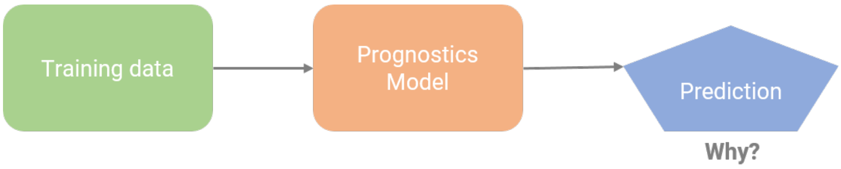

Figure 1: Problematic of most machine learning models performing as black boxes.

In-model interpretability [Carvalho et al., 2019] focuses on intrinsically interpretable models. These "transparent" models, naturally and by design, provide some degree of interpretability. In their work, Fellous et al. [2019] identifies four classes of approaches to achieve in-model interpretability: 1) hybrid models, 2) architecturally explainable models, 3) explainable convolutional networks, and 4) models with regularization.

In addition to pre-model and in-model interpretability, there is post-model interpretability [Carvalho et al., 2019]. Post-model techniques analyze the model after its creation (post-hoc); they are devised as independent methods that can interpret the final decisions. There approaches can be model-specific or model-agnostic. Post-hoc model-specific interpretability consists of methods specifically designed for a given machine learning algorithm. In contrast, post-hoc model-agnostic interpretability is agnostic to the analyzed machine learning model.

Examples of models that follow the model-agnostic approach include Local Interpretable Model-Agnostic Explanations (LIME) [Ribeiro et al., 2016] and its variants. The variants of LIME attempt to address its limitations. For example, NormLime proposed by Ahern et al. [2019] tackles the issue of deriving global interpretability from local explanations. Another popular post-hoc interpretability model is SHAP (SHapley Additive exPlanations) [Lundberg and Lee, 2017]. SHAP works by assigning a SHAP value to each predictor to indicate its contribution to the final outcome. SHAP is one of the most consistent approaches to post-hoc interpretability. Other approaches are described in [Carvalho et al., 2019, Arrieta et al., 2020].

Assignment Description

Importantly, to fulfill this assignment, a working prognostics model is needed (in addition to the XAI model). To show your model's prognostics performance, you can train and test your methodologies in the "train_FD001.txt" dataset. This is a dataset of engines that exhibit only one fault mode and one operational condition. The other datasets, which you can also use, are more complex. The "test_FD001.txt" file and the other test files have smaller trajectories and should **not** be preferably used. The prognostics models can be selected from the existing machine learning alternatives (SVR, RF, MLP, LSTM, CNN, LSTM-CNN, Transformers, etc.).

A Mean Absolute Error (MAE) value of at most 40 cycles ($\leq$ 40 cycles) is expected for the "train_FD001.txt" data. The applicant is evaluated not only by the accuracy of its model but also by the novelty and soundness of their holistic approach. The clarity and quality of the code are also criteria.

A method of XAI, on top of the prognostics model, which makes sense in the context of the case study, should be selected and implemented. You can of course implement more than 1 method. These methods aim to explain/interpret the RUL predictions generated by prognostics models. The student should be able to implement one method to perform prognostics on the case study of CMAPSS.

It is desirable but not necessary to implement preprocessing techniques. The focus is on the XAI model and your interpretation of the results.

Data

You can find the assignment data at: https://www.nasa.gov/content/prognostics-center-of-excellence-data-set-repository (Turbofan Engine Degradation Simulation - 6)

Again, you can train and test your methodologies in the "train_FD001.txt" dataset. This is a dataset of engines that exhibit only one fault mode and one operational condition. The other datasets, which you can also use, are more complex. The "test_FD001.txt" file and the other test files have smaller trajectories and should **not** preferably be used.

Available Code

Several GitHub repositories, such as in the link below, present solutions to this problem. You can base your ideas on the work of others, but it is expected that you credit and **extend** their solutions:

https://github.com/jiaxiang-cheng/PyTorch-Transformer-for-RUL-Prediction

Requirements

- Prognostics models (min. 1) selected and implemented (30%)
- One method for XAI selected and implemented (30%)
- Report writing and future recommendations (30%)
- Novelty and Effort (10%)

Deliverables

- Report of max 8 pages
- Jupyter notebook(s) (seperated notebooks can be delivered in a zip or Github project)

Links

- Report of max 8 pages
- Jupyter notebook(s) (seperated notebooks can be delivered in a zip or Github project)

References

Abhinav Saxena, Kai Goebel, Don Simon, and Neil Eklund. Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 international conference on prognostics and health management*, pages 1–9. IEEE, 2008.

David Gunning, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. Xai—explainable artificial intelligence. *Science robotics*, 4(37):eaay7120, 2019.

Timo Speith. A review of taxonomies of explainable artificial intelligence (xai) methods. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 2239–2250, 2022.

David Solís-Martín, Juan Galán-Páez, and Joaquín Borrego-Díaz. On the soundness of xai in prognostics and health management (phm). *Information*, 14(5):256, 2023.

Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8):832, 2019.

John W Tukey et al. *Exploratory data analysis*, volume 2. Reading, MA, 1977.

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE transactions on neural networks and learning systems*, 32(11):4793–4813, 2020.

Jean-Marc Fellous, Guillermo Sapiro, Andrew Rossi, Helen Mayberg, and Michele Ferrante. Explainable artificial intelligence for neuroscience: behavioral neurostimulation. *Frontiers in neuroscience*, 13:1346, 2019.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

Isaac Ahern, Adam Noack, Luis Guzman-Nateras, Dejing Dou, Boyang Li, and Jun Huan. Normlime: A new feature importance metric for explaining deep neural networks. *arXiv preprint arXiv:1909.04200*, 2019.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

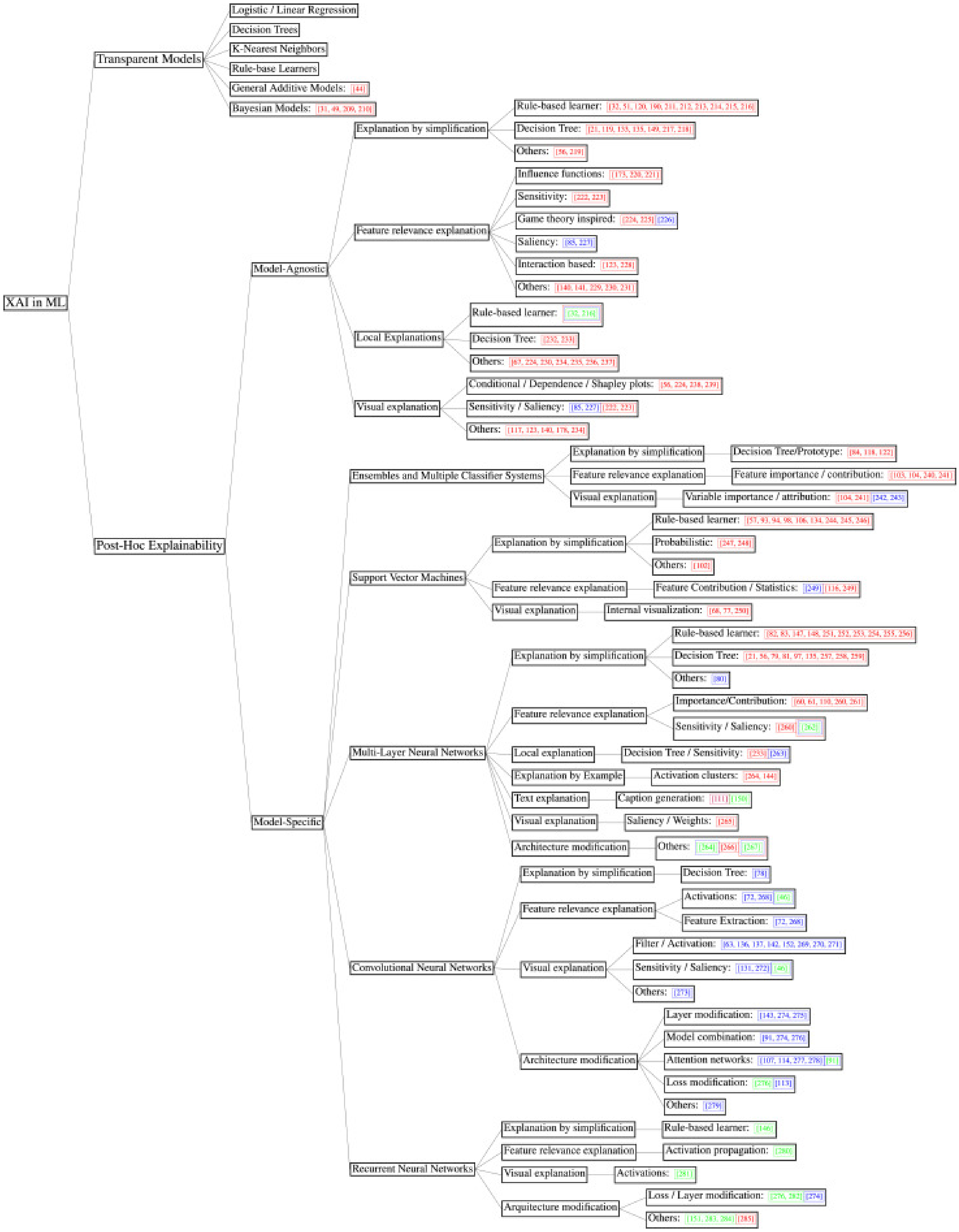
XAI in ML
Transparent Models
Logistic / Linear Regression
Decision Trees
K-Nearest Neighbors
Rule-base Learners
General Additive Models: [44]
Bayesian Models: [31, 49, 209, 210]
Post-Hoc Explainability
Model-Agnostic
Explanation by simplification
Rule-based learner: [32, 51, 120, 190, 211, 212, 213, 214, 215, 216]
Decision Tree: [21, 119, 133, 135, 149, 217, 218]
Others: [56, 219]
Feature relevance explanation
Influence functions: [173, 220, 221]
Sensitivity: [222, 223]
Game theory inspired: [224, 225] [226]
Saliency: [85, 227]
Interaction based: [123, 228]
Others: [140, 141, 229, 230, 231]
Local Explanations
Rule-based learner: [32, 216]
Decision Tree: [232, 233]
Others: [67, 224, 230, 234, 235, 236, 237]
Visual explanation
Conditional / Dependence / Shapley plots: [56, 224, 238, 239]
Sensitivity / Saliency: [85, 227] [222, 223]
Others: [117, 123, 140, 178, 234]
Model-Specific
Ensembles and Multiple Classifier Systems
Explanation by simplification
Decision Tree/Prototype: [84, 118, 122]
Feature relevance explanation
Feature importance / contribution: [103, 104, 240, 241]
Visual explanation
Variable importance / attribution: [104, 241] [242, 243]
Support Vector Machines
Explanation by simplification
Rule-based learner: [57, 93, 94, 98, 106, 134, 244, 245, 246]
Probabilistic: [247, 248]
Others: [102]
Feature relevance explanation
Feature Contribution / Statistics: [249] [116, 249]
Visual explanation
Internal visualization: [68, 77, 250]
Multi-Layer Neural Networks
Explanation by simplification
Rule-based learner: [82, 83, 147, 148, 251, 252, 253, 254, 255, 256]
Decision Tree: [21, 56, 79, 81, 97, 135, 257, 258, 259]
Others: [80]
Feature relevance explanation
Importance/Contribution: [60, 61, 110, 260, 261]
Sensitivity / Saliency: [260] [262]
Local explanation
Decision Tree / Sensitivity: [233] [263]
Explanation by Example
Activation clusters: [264, 144]
Text explanation
Caption generation: [111] [150]
Visual explanation
Saliency / Weights: [265]
Architecture modification
Others: [264] [266] [267]
Convolutional Neural Networks
Explanation by simplification
Decision Tree: [78]
Feature relevance explanation
Activations: [72, 268] [46]
Feature Extraction: [72, 268]
Visual explanation
Filter / Activation: [63, 136, 137, 142, 152, 269, 270, 271]
Sensitivity / Saliency: [131, 272] [46]
Others: [273]
Architecture modification
Layer modification: [143, 274, 275]
Model combination: [91, 274, 276]
Attention networks: [107, 114, 277, 278] [91]
Loss modification: [276] [113]
Others: [279]
Recurrent Neural Networks
Explanation by simplification
Rule-based learner: [146]
Feature relevance explanation
Activation propagation: [280]
Visual explanation
Activations: [281]
Arquitecture modification
Loss / Layer modification: [276, 282] [274]
Others: [151, 283, 284] [285]

Figure 2: Taxonomy of XAI methods (taken from [Arrieta et al., 2020]).