# CeRULEo: Comprehensive utilitiEs for Remaining Useful Life Estimation methOds

Luciano Lorenti [1][¶] **and Gian Antonio Susto** [1]

1 Department of Engineering, University of Padova. Italy ¶ Corresponding author

Summary

CeRULEo, which stands for Comprehensive utilitiEs for Remaining Useful Life Estimation methOds, is a Python package designed to train and evaluate regression models for predicting remaining useful life (RUL) of equipment. RUL estimation is a process that uses prediction methods to forecast the future performance of machinery and obtain the time left before machinery loses its operation ability. The RUL estimation has been considered as a central technology of Predictive Maintenance (PdM) (Heimes, 2008; X. Li et al., 2018). PdM techniques can statistically evaluate a piece of equipment's health status, enabling early identification of impending failures and prompt pre-failure interventions, thanks to prediction tools based on historical data (Susto et al., 2014). CeRULEo offers a comprehensive suite of tools to help with the analysis and pre-processing of preventive maintenance data. These tools also enable the training and evaluation of RUL models that are tailored to the specific needs of the problem at hand.

Statement of need

Effective maintenance management helps reduce costs related to defective products and equipment downtime. A well-planned maintenance strategy improves reliability, prevents unexpected outages, and lowers operating costs.

In Industry 5.0, the industrial machines produce a large amount of data which can be used to predict an asset's life (Khan et al., 2023). RUL estimation uses prediction techniques to forecast a machine's future performance based on historical data, enabling early identification of potential failures and prompt pre-failure interventions.

Within the PdM and RUL regression ecosystem, finding a library that effectively combines modelling, feature extraction capabilities, and tools for model comparison poses a significant challenge. While numerous repositories and libraries exist for models and feature extraction in time series data (Christ et al., 2018; Tavenard et al., 2020), few offer a comprehensive solution that integrates both aspects effectively. The prog_models and prog_als libraries from NASA (Teubert et al., 2022) come closest to fulfilling this requirement. However, they have a strong focus on simulation and lack extensive mechanisms for feature extraction from time series data.

On the other hand, CeRULEo provides a comprehensive set of utilities designed to train and evaluate regression models for predicting RUL of equipment. CeRULEo emphasizes a data-driven approach using industrial data, particularly when a simulation model is unavailable or costly to develop, prioritizing model library-agnosticism for easy deployment in any production environment.

In order to achieve good performance, RUL regression requires data preparation and feature engineering. Typically, machinery data is provided as time series data from various sensors

Lorenti, & Susto. (2023). CeRULEo: Comprehensive utilitiEs for Remaining Useful Life Estimation methOds. *Journal of Open Source Software*, *8*(88), 5294. https://doi.org/10.21105/joss.05294

during operation. The first step in data preparation is often to create a dataset based on run-to-failure cycles. This involves dividing the time series into segments where the equipment starts in a healthy state and ends in a failure state, or is close to failure. The second step of data preparation is preprocessing. While PdM models can be used in a variety of contexts with different data sources and errors, there are some general techniques that can be applied (Serradilla et al., 2022), such as time-series validation, imputing missing values, handling homogeneous or non-homogeneous sampling rates, addressing values, range and behaviour differences across different machines and the creation of run-to-failure-cycle-based data.

CeRULEo addresses these issues by providing a comprehensive toolkit for preprocessing time series data for use in PdM models, with a focus on run-to-failure cycles. The preprocessing includes sensor data validation methods, for studying not only missing and corrupted values but also distribution drift among different pieces of equipment.

In addition to preprocessing, it enables the iteration of machine data for use in both mini-batch and full-batch regression models, and is compatible with popular machine learning frameworks such as scikit-learn (Pedregosa et al., 2011) and tensorflow (Abadi et al., 2015). The library also includes a catalog of successful deep learning models (Chen et al., 2022; Jayasinghe et al., 2019; H. Li et al., 2020) from the literature and a collection of commonly used RUL datasets for quick model evaluation.

The acceptance of PdM technologies is pivotal in Industry 5.0 for successful implementation, but hesitations or reluctance by decision-makers can still pose significant barriers (Oudenhoven et al., 2022). One effective approach to foster acceptance and understanding is through explainability, which plays a crucial role in PdM. As such, CeRULEo incorporates explainable models capable of providing additional information about the predictions, enhancing comprehension: one that can select the most relevant features for the model (Lemhadri et al., 2021), and a convolutional model (Fauvel et al., 2021) that provides post-hoc explanations of the predictions to understand the reasoning behind the predicted RUL.

Moreover, CeRULEo provides tools for evaluating and comparing PdM models based on not only traditional regression metrics, but also on their ability to prevent errors and reduce costs. In many cases, the costs of not accurately detecting or anticipating faults can be much higher than the cost of inspections or maintenance due to reduced efficiency, unplanned downtime, and corrective maintenance expenses. In PdM, it is particularly important to be accurate about the RUL of equipment near the end of its lifespan, as an overestimation of RUL can have serious consequences when immediate action is required. CeRULEo addresses this issue by providing mechanisms for weighting samples according to their importance and asymmetric losses for training models, as well as visualization tools for understanding model performance in relation to true RUL.

Financial Acknowledgement

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. https://doi.org/10.48550/arXiv.1603.04467

Chen, Y., Zhang, D., & Zhang, W. (2022). MSWR-LRCN: A new deep learning approach to remaining useful life estimation of bearings. *Control Engineering Practice*, *118*, 104969. https://doi.org/10.1016/j.conengprac.2021.104969

Christ, M., Braun, N., Neuffer, J., & Kempa-Liehr, A. W. (2018). Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package). *Neurocomputing*, *307*, 72–77. https://doi.org/10.1016/j.neucom.2018.03.067

Fauvel, K., Lin, T., Masson, V., Fromont, É., & Termier, A. (2021). Xcm: An explainable convolutional neural network for multivariate time series classification. *Mathematics*, *9*(23), 3137. https://doi.org/10.3390/math9233137

Heimes, F. O. (2008). Recurrent neural networks for remaining useful life estimation. *2008 International Conference on Prognostics and Health Management*, 1–6. https://doi.org/10.1109/phm.2008.4711422

Jayasinghe, L., Samarasinghe, T., Yuenv, C., Low, J. C. N., & Ge, S. S. (2019). Temporal convolutional memory networks for remaining useful life estimation of industrial machinery. *2019 IEEE International Conference on Industrial Technology (ICIT)*, 915–920. https://doi.org/10.1109/icit.2019.8754956

Khan, M., Haleem, A., & Javaid, M. (2023). Changes and improvements in industry 5.0: A strategic approach to overcome the challenges of industry 4.0. *Green Technologies and Sustainability*, *1*(2), 100020. https://doi.org/10.1016/j.grets.2023.100020

Lemhadri, I., Ruan, F., & Tibshirani, R. (2021). Lassonet: Neural networks with feature sparsity. *International Conference on Artificial Intelligence and Statistics*, 10–18. https://doi.org/10.48550/arXiv.1907.12207

Li, H., Zhao, W., Zhang, Y., & Zio, E. (2020). Remaining useful life prediction using multi-scale deep convolutional neural network. *Applied Soft Computing*, *89*, 106113. https://doi.org/10.1016/j.asoc.2020.106113

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

Oudenhoven, B. van, Van de Calseyde, P., Basten, R., & Demerouti, E. (2022). Predictive maintenance for industry 5.0: Behavioural inquiries from a work system perspective. *International Journal of Production Research*, 1–20. https://doi.org/10.1080/00207543.2022.2154403

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://doi.org/10.48550/arXiv.1201.0490

Serradilla, O., Zugasti, E., Rodriguez, J., & Zurutuza, U. (2022). Deep learning models for predictive maintenance: A survey, comparison, challenges and prospects. *Applied Intelligence*, 1–31. https://doi.org/10.1007/s10489-021-03004-y

Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2014). Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial*

Informatics, *11*(3), 812–820. https://doi.org/10.1109/tii.2014.2349359

Tavenard, R., Faouzi, J., Vandewiele, G., Divo, F., Androz, G., Holtz, C., Payne, M., Yurchak, R., Rußwurm, M., Kolar, K., & Woods, E. (2020). Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, *21*(118), 1–6. http://jmlr.org/papers/v21/20-091.html

Teubert, C., Corbetta, M., Kulkarni, C., Jarvis, K., & Daigle, M. (2022). *Prognostics models python package* (Version 1.4). https://github.com/nasa/prog_models