AI·PROFICIENT

Artificial intelligence
for improved production efficiency,
quality and maintenance

Deliverable 2.4

D2.4: Local AI for proactive maintenance support

WP2: Smart components and local AI at system edge

T2.4: Self-prognostics and component operating condition estimation

Version: 1.0

Dissemination Level: PU

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

Table of Contents

List of Figures

List of Tables

Disclaimer

This document contains a description of the AI-PROFICIENT project work and findings.

The authors of this document have taken any available measure for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated in the creation and publication of this document hold any responsibility for actions that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of the AI-PROFICIENT consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 28 Member States of the Union. It is based on the European Communities and the Member States cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors (http://europa.eu/).

AI-PROFICIENT has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 957391.

Title: D2.4: Local AI for proactive maintenance support

Lead Beneficiary:	UL
Due Date:	30/04/203
Submission Date	02/05/2023
Status	Final ~~Preliminary~~ ~~Draft~~
Description	Algorithms for prognostics applied to manufacturing assets and components.
Authors	Kerman Lopez de Calle (TEK), Regis Benzmuller (CONTI), Vasillis Spais (INOS), Marc Anderson (UL), Alaaeddine Chaoub (UL), Alexandre Voisin (UL)
Type	Other
Review Status	~~Draft~~ ~~WP Leader accepted~~ PC + TL accepted
Action Requested	~~Contribution from partners requested~~ ~~To be revised by partners~~ ~~For approval by the WP leader~~ ~~For approval by the Project Coordinator & Technical Leaders~~ For acknowledgement by partners

VERSION	ACTION	OWNER	DATE
0.1	First version released	UL	01/04/2023
0.2	Partners input	UL, TEK	15/04/2023
0.3	Final version	UL, TEK	21/04/2023
1.0	Final version reviewed	UL, ATC	01/05/2023

Executive Summary

Deliverable D2.4, Local AI for proactive maintenance support, presents the advances made in the context of Work Package 2 (WP2) Smart components and local AI at system edge that are related to the development of edge systems used for the prognostics of the assets health in which they are embedded or run.

Prognostic services are one of the cornerstones of Industry 4.0. These services can be used by higher level systems (such as the ones developed in WP3 of AI-PROFICIENT project) to optimize asset operation in coordination with other assets to optimize the maintenance scheduling for instance; or, by other edge systems (such as the ones developed in T2.5) that could modify their controls adapting them to the current condition of the controlled asset in order to optimize their usage timespan.

This deliverable aims to disseminate the prognostics models and algorithms that have been developed within Task 2.4, Self-prognostics and component operating condition estimation. As such, some results on public datasets will be presented as well as some results on APROFICIENT use cases.

1 Introduction

The goal of this deliverable is to gather the contribution that has been provided in task 2.4 in relation with Self-prognostics and component operating condition estimation using AI technique in the context of AI-PROFICIENT project. The objective of the task is to provide prognostics capabilities at component level and derive the corresponding service. Hence, the task contributes to the development of S_PRO service described in D1.5 and recap in Table 1.

Table 1: Excerpt of S_PRO service description from D1.5

Service ID	S_PRO
High level service description:	Degradation based prognostics is based on a degradation model of the degradation modes of the equipment. The degradation models are mainly built based on historical data and may consider age, usage and measurement of the equipment. Such a model is then updated depending on available current measurements. The degradation model makes projections over the future in order to predict the remaining useful life of the item in consideration. Such a model includes AI-based techniques but also more conventional approaches such as stochastic processes, trend, and time series models. They may deliver not only the RUL but also the degradation trajectory. RUL prediction prognostics provides only the RUL of the component. Such a model has already been investigated in the project on the public C-MAPSS dataset. The proposed deep neural networks used for this purpose exploit automatic representation learning to discover weak and complex correlations between sensors that may not be easily captured by domain experts and thus potentially increase portability of the prediction model to other configurations and environments.

This service aims at covering the _PRO requirements identified and detailed in the deliverable D1.4 (see Table 2).

Table 2: Functionalities to be provided by the AI-PROFICIENT project (from D1.4).

AI-PROFICIENT Functionalities	***ID***
Monitoring	_MON
Diagnostic and anomaly detection	_DIA
Health state evaluation	_HEA
Component prognostics	_PRO
Hybrid models of production processes and digital twins	_HYB
Predictive Production quality assurance	_PRE
Root-cause identification	_ROO
Early anomaly detection	_EAR
Opportunistic maintenance decision-making	_OPP
Generative holistic optimization	_GEN
Future scenario based Lifelong self-learning system	_LSL
Human feedback	_HUM
Explainable and transparent decision making	_ETD

In the context of AI-PROFICIENT, 4 use cases have been selected, in WP1, to design, develop, and demonstrate the services provided by the project. During the elaboration of D1.3 (Pilot-specific demonstration scenarios) some use cases include task 2.4 as potential contributor for component prognostics (see Table 3).

Table 3: Original excerpt of expected partners involvement in T2.3 for each use case (from D1.3).

WP/Task	CONTI-2	CONTI-3	CONTI-5	CONTI-7	CONTI-10	INEOS-1	INEOS-2	INEOS-3
WP2– Smart components and local AI at system edge								
T2.4	TEK	UL	TEK UL					UL

Nevertheless, when exploring more in detail the use cases data availability and partners' intention to support task 2.4, some use cases have been discarded. After data analysis, no prognostics will be performed in CONTI-2. For INEOS-3 use case, INEOS decided to stop the use case after data analysis. Table 4 reflects the actual situation during the writing of this deliverable.

Table 4: Updated partner and UC contribution matrix.

WP/Task	CONTI-2	CONTI-3	CONTI-5	CONTI-7	CONTI-10	INEOS-1	INEOS-2	INEOS-3
WP2– Smart components and local AI at system edge								
T2.4		UL	TEK					

The deliverable is structured as follows. The second section presents the main approaches for prognostics including AI techniques. The third section provides the scientific contribution provided by the task on prognostics. The fourth section describes the prognostics services developed on the use case.

2 Prognostics: Definition and AI based improvements

Prognostics involve predicting the future health state of a system to anticipate potential failures before they occur (Jardine at al., 2006). This prediction can take the form of a forecast of the remaining useful lifetime or an estimation of the system's health for future operations. Prognostics rely on the features generated by the condition assessment step and the output of the diagnostics step.

As noted in the literature (Peng et al., 2010), prognostics can be broadly categorized into three main types: physics-based models, data-driven models, and hybrid models. Physics-based models make predictions based on physical laws and principles governing the system. Data-driven models rely on historical data to learn the system's behavior and predict its remaining useful life (RUL). Hybrid models combine both approaches. AI-PROFICIENT aims at leveraging AI techniques and as such the proposed approach relies on data driven techniques.

Machine learning has already been extensively used in system health prognostics, with promising results. In (Leukel et al., 2021) a systematic review has been conducted on prognostics of industrial systems using machine learning, which involves using data-driven methods. One example of an advanced learning model that has been applied to various applications is the long-short term memory neural network (LTSM). This type of model has been considered in AI-PROFICIENT and will be developed for both as scientific advance, since in (Chaoub et al., 2021) such a model has been used to predict the remaining useful life (RUL) of turbofan engines and for use case developments.

3 Scientific contribution of T2.4

We report here 2 scientific contributions developed within T2.4 that have been published. The first deals with a Deep Learning model for prognostics and the second with some development of these models improving its interpretability thanks to mixture of expert approach.

3.1 Learning Representations with End-to-End Models for Improved Remaining Useful Life Prognostic

This work has been published as:

Alaaeddine Chaoub, Alexandre Voisin, Christophe Cerisara, Benoît Iung. Learning representations with end-to-end models for improved remaining useful life prognostic, European Conference of the Prognostics and Health Management Society, Jun 2021, Virtual event, Italy

We proposed an MLP-LSTM-MLP architecture that is trained end-to-end to predict RUL. This specific architecture has been proposed in a prior study (An et al., 2020) and has exhibited encouraging outcomes for diagnostic purposes. Such architecture, should be able to overcome two main drawbacks:

- Introducing an initial feature selection phase can potentially hinder the modeling process by removing important information and subtle signals that experts may have missed or overlooked.
- Well-designed and uncomplicated neural networks often perform just as well as more intricate deep learning models. The latter requires extensive and energy-intensive experimentation to fine-tune their hyperparameters, which poses a technical obstacle for industrial applications.

The proposed architecture is presented Figure 1.

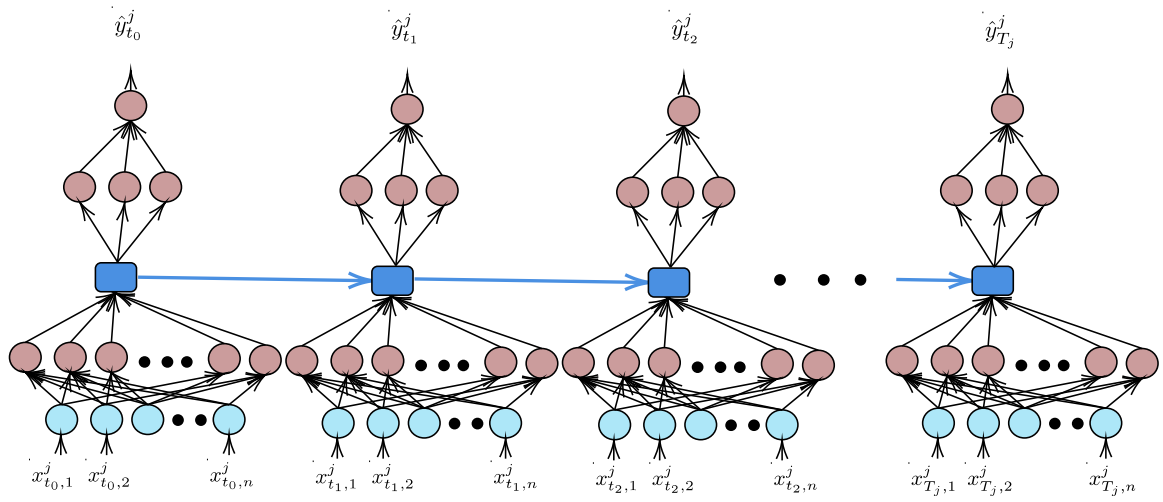

Figure 1: Architecture of the proposed model. To simplify the diagram, only one layer has been drawn for the MLPs.

The model has been trained on the well-known C-MAPSS dataset; this dataset is probably the most used for prognostics purpose. The C-MAPSS dataset has been generated using the simulation program by monitoring the degradation of multiple Turbofan engines called commercial modular aero-propulsion system simulation. The description of the dataset can be found in (Saxena et al., 2008).

Our model utilizes full time series data of each turbofan engine from the first cycle to failure, and does not rely on fixed sequence length, truncation, or padding. Therefore, the sequence length of each

sample may vary. We used 75% of the turbofan engines' run-to-failure trajectories for training and the remaining 25% for validation.

The validation set was used to manually adjust the hyper-parameters through a few trials and errors. These hyper-parameters include the learning rate, the number of layers in the input and output MLPs, the number of LSTM layers, the number of neurons or cells in each layer, the activation functions, the dropout percentages, and the optimizer presents the optimal hyper-parameters discovered for the proposed model.

Hyper-parameter	Value
Learning Rate	0.0001
Number of MLP layers before LSTM	3
Number of neurons in MLP layers	100/50/50
Number of LSTM layers	1
Number of LSTM cells	60
Number of MLP layers after LSTM	3
Number of neurons in MLP layers	60/30/1
Activation function for MLP layers	Tanh()
Batch size	5
Dropout percentage	0%

Figure 2: Hyper-parameters of the proposed mode

Due to the random initialization of the model parameters, the optimized values may differ between training runs. To account for this variability, we conducted 10 training runs and report the mean values and standard deviations of the model's performance on the four data sets in Figure 3[1]. The comparison with state-of-the-art models in the literature is presented Figure 4.

Further insight can be found in the afore-mentioned paper.

DATASET	FD001	FD002	FD003	FD004
RMSE	13.26 ± 0.57	12.49 ± 0.28	13.11 ± 1.28	13.97 ± 0.48
SCORE	284.88 ± 42.32	571.4 ± 37.45	352.39 ± 179.96	1252.32 ± 104.97

Figure 3: Results of the proposed model on the 4 subsets of CMAPSS.

[1] In this table the RMSE is the Root Mean Square Error and the Score is a penalty function that disadvantage late prognostic rather than early. Both are defined in (Saxena et al., 2008)

Models	DATA SETS								Average	
	FD001		FD002		FD003		FD004			
	RMSE	Score	RMSE	Score	RMSE	Score	RMSE	Score	RMSE	Score
DA-CNN (Song et al., 2020)	11.78	229.48	16.95	1842.38	11.56	257.11	18.23	2317.32	14.63	1161.57
DCGAN (Hou et al., 2020)	**10.71**	**174**	19.49	2982	**11.48**	273	19.71	3874	15.34	1825.75
MS-DCNN (Li et al., 2020)	11.44	196.22	19.35	3747	11.67	**241.89**	22.22	4844	16.17	2257.27
HDNN (Al-Dulaimi et al., 2019)	13.017	245	15.24	1282.42	12.22	287.72	18.15	1527.42	14.65	835.64
LSTM (Pasa et al., 2019)	16.5	444	18.1	942	15.9	718	17.2	1487	16.92	897.75
Proposed LSTM without the first MLP	14.31	337.86	17.44	1716.11	15.53	1356.36	18.86	2111.05	16.53	1380.34
Proposed LSTM with the first MLP	13.26	284.88	**12.49**	**571.4**	13.11	352.39	**13.97**	**1252.32**	**13.20**	**615.24**

Figure 4: comparison of the proposed approach with state-of-the-art results.

3.2 Towards interpreting deep learning models for industry 4.0 with gated mixture of experts

This work has been published as:

Alaaeddine Chaoub, Christophe Cerisara, Alexandre Voisin, Benoît Iung. Towards interpreting deep learning models for industry 4.0 with gated mixture of experts. 30th European Signal Processing Conference, EUSIPCO 2022, Aug 2022, Belgrade, Serbia.

The objective of this work was to apply the Gated Mixture of Experts (GMoE), to the model developed and presented in the previous section, to interpret the deep learning model trained on industrial data. We also proposed to add a regularization term to the loss that includes prior knowledge and enables to boost the performances. Unlike traditional deep learning models, this approach decomposes parts of the model in a way that can be understood by domain experts or users. The study transforms the above presented model that performs well on the C-MAPSS dataset for predicting the RUL. The structure of the model is presented Figure 5. The first MLP layer of the previous model is replaced by a MLP GMoE. The same hyper-parameters as above have been used. Indeed, this architecture has been selected since it is expected that the first part of the model, i.e., GMoE, will be able to retrieve/discover that the turbofan measurement were done under several operating conditions (OC). We know from the dataset description that 6 OC were used to generate the data.

The number of OCs in real-world scenarios may not be known and is often estimated by experts. To test the robustness of the approach, experiments were conducted using 6 and 9 experts. Results may vary across different training runs due to random initialization, so each experiment was run 20 times to calculate the variance. Model parameters were chosen on the validation corpus by manually testing a few reasonable values and early stopping was used during training with a maximum of 2000 epochs. The model with the lowest validation loss was chosen for evaluation on the test corpus.

At each timestep, the GMoE network produces a probability distribution across the experts, from which we can determine the predicted cluster by identifying the argmax. The true value, using the OC, and the predicted clusterings can be evaluated using the normalized mutual information (NMI) (Kvalseth, 1987).

The clustering generated by the simple GMoE-LSTM-MLP is shown Figure 6. Results indicate that in over 87% of runs, only one or two experts are utilized, and in such cases, the corresponding NMI is at most 0.5. When nine experts are available, the GMoE may use up to five experts, but this occurs in only 5% of the runs. These findings suggest that although the gating network fails to recover the expected six clusters, increasing the number of clusters to three results in more correlated clustering with the

operating conditions (NMI reaching 0.5 and 0.7 in one run). However, for larger values of Nc, up to five, the NMI decreases, indicating that the target six clusters are too specific for the simple GMoE.

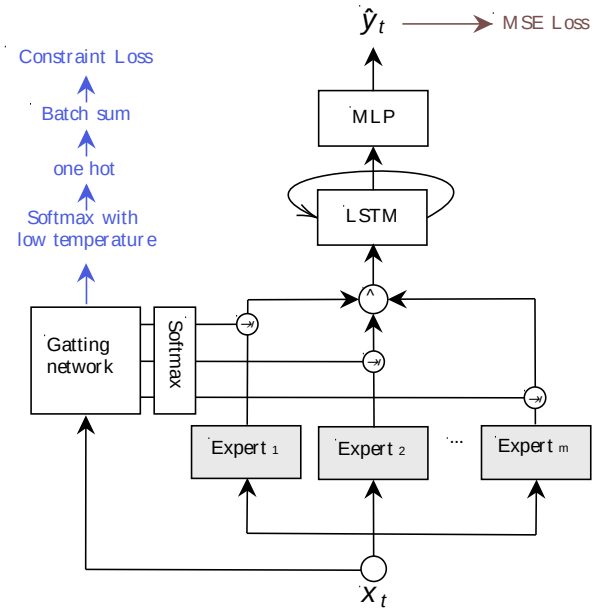

Figure 5: GMoE-LSTM-MLP architecture with m experts.

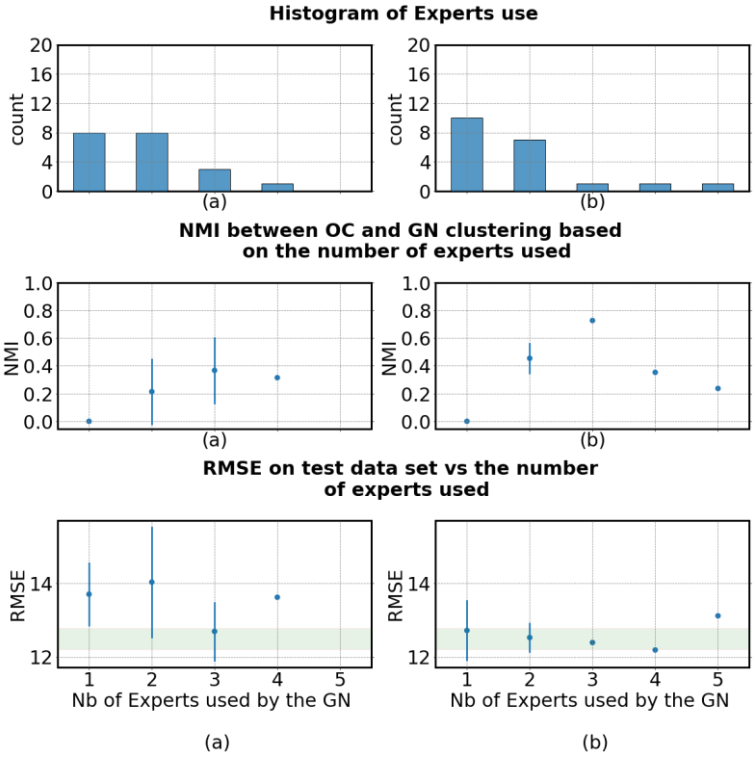

Figure 6: Clustering evaluation when the simple GMoE-LSTM-MLP is trained on all data; left column (a): m = 6; right column (b): m = 9.

To get better results, we introduced a regularization term to the loss. The idea is to add a posterior regularization term to the loss function that encourages the frequency distribution of Experts to match known prior. We assumed a uniform prior frequency distribution over the 6 OCS. The regularization term can be balanced using a weight factor (Lambda). Figure 7 shows the results with the regularization term and several weight (λ).In this case, regardless of the number of experts used, mostly 3 clusters are predicted that match the OCs with an NMI of 0.7. Increasing the number of experts increases the NMI logarithmically, which encourages the model to decompose the data in an interpretable way. The number of predicted clusters is not larger than the number of OCs even when using all 9 experts. The model's predictive performances do not vary much across conditions and have RMSE values close to the state of the art. Changing the strength of the constraint does not result in significant changes in model interpretability or performance.

Further insight can be found in the afore mentioned paper.

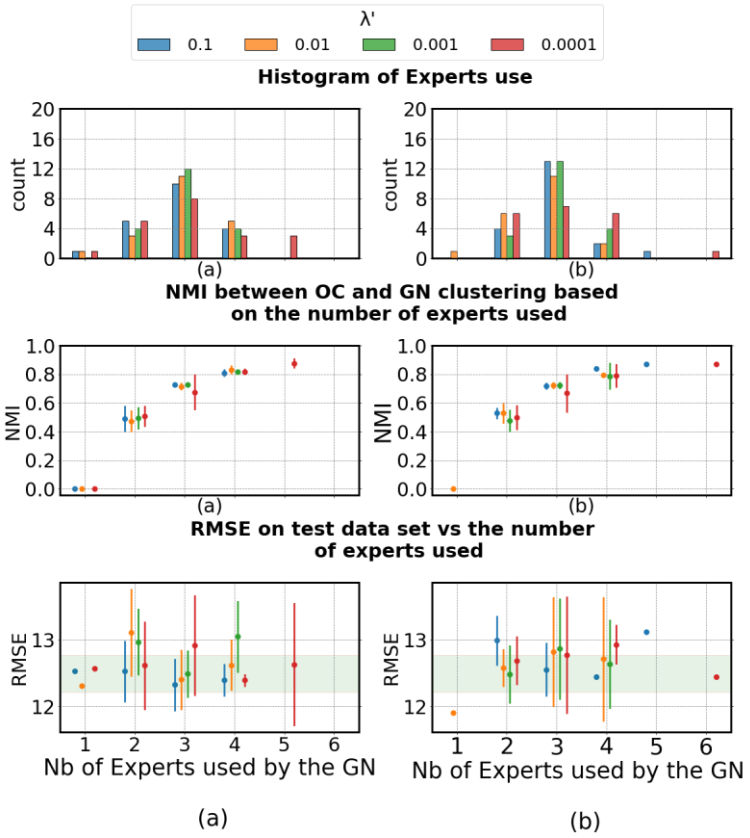

Figure 7: GMoE-LSTM-MLP with knowledge-based loss constraint results, the constraint value is not part of the validation loss; left column (a): m = 6; right column (b): m = 9; different weight factors (λ) are considered.

4 AI-PROFICIENT prognostics use cases

Different technologies have been developed in the 2 UCs of task 2.4. The following sections provide a short recap of the use case and the approaches followed in order to develop the prognostics capabilities at the edge level.

4.1 CONTI3 – Released extrusion optimization.

4.1.1 UC description

Relaxed extrusion is a concept to improve the quality of the semi products produced on the Combiline. When extruding the objective is to have the minimum tension inside the product so that shrinkage effects after cutting are minimized to avoid length issues and bad weight repartition on the surface of

the tire (RFPP deviations). There are 3 factors to consider minimizing tension in the product among which the easiest controllable during the production is the conveying, the 2 others being the flow balancing in the die and the visco-elastic phenomenon.

4.1.2 Proposed solution

In this use case, the service must predict the drift of the extruder that can lead to not relaxed product. When a not relaxed product is about to occur, some changes in the setting point of V1 need to be done. Such changes can occur manually or thanks to a control loop at the line level. Hence the goal of the model will be to prognosticate the remaining time before the V1 setpoint change.

Some data are available on PETA[2] repository. Thank to discussion with Continental expert, we have defined some rules to segregate the data into relevant segments. Indeed, this use case is focused on in-production drift. Hence, all set-up time, stoppage, sidewall production, trials… have to be removed from the data. Only sequences of thread production have been kept.

The proposed solution is based on the MLP-LSTM-MLP model presented on section 3.1. Furthermore, thanks to the analysis of the data and some initial trials, the use case has been reformulated as classification task for which the time frame remaining to a V1 setting point change has to be predicted. Indeed, no other measurement of the relaxation of the product is available.

4.1.3 First developments

In this early development the classes used represent the number of half minutes remaining before change, this makes evaluating the performance of the model easier to understand and analyze. As shown in Figure 8, when the time left to change is more than 5 minutes (before the green vertical line), the frames are considered to belong to a class that intuitively represents a relaxed product on the line.

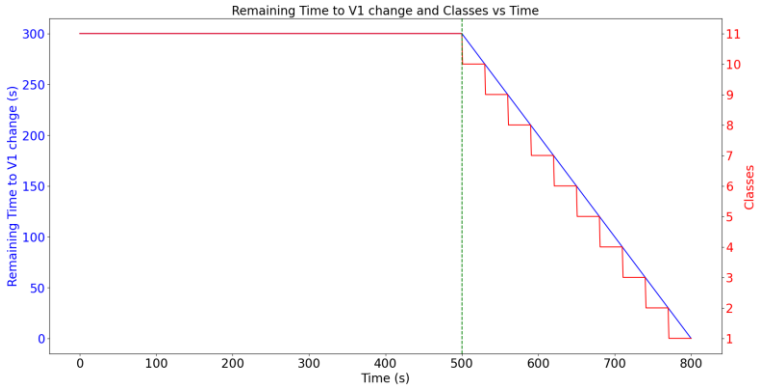

Figure 8: The remaining time to change with the true classes used for training the model.

In this development stage, we used only 1 month of data and did not fully optimize the model. The aim is more to agree on the objective of the model and the metric to evaluate the performances.

Some results have been obtained and are shown in Figure 9 and Figure 10.

As a conclusion, the proposed model can predict V1 setpoint change in about 55% of the trajectory, in the remaining 45%, the model is not able to predict the V1 setpoint change (predicted class 11 vs actual

[2] PETA is a cloud repository for research data hosted by UL. The PETA service allows to store data on a centralized platform, hosted on the servers of the UL and managed by the UL digital department. Secured by an authentication, the access to the data is done through a WEB browser or via a software installed on the computer. The service guarantees the partitioning of data by isolating them by structure or by project. This data repository is reliable, secure and resilient.

class from 10 to 1 in Figure 9). When deviations are detected, as shown in Figure 10, most of the points are around the diagonal, which provides proof of concept for the future development of the approach. When setpoint change is not detected, we assume that the change cause are not in the hot part of the Combiline but in its cold part for which we do not have data.

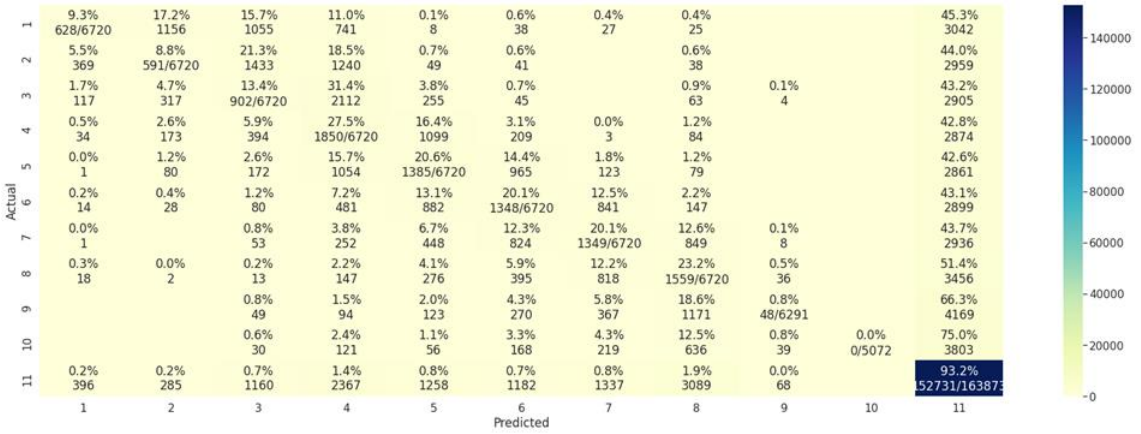

Figure 9: Confusion matrix between actual and predicted classes

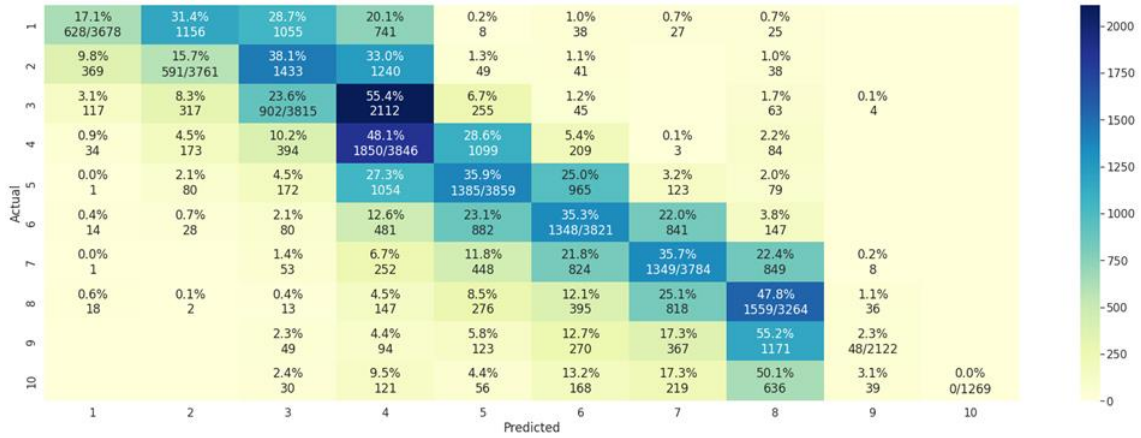

Figure 10:Confusion matrix between actual and predicted classes without the final class that represents the relaxed product.

The next steps for the models are to consider external factors that have some influence as reported by the process engineer and the Combiline driver, such as Air Temperature, type of compound and more. It is also planned to consider the whole dataset (over the 1,5 years) to train the model.

Bearing in mind the solutions will be delivered to the operator, Continental and the ethic teams have been put in the loop. A preliminary ethical issue has been raised regarding how the operator's perception of the accuracy of the AI suggestions will affect the operator's trust and subsequent use of them. The issue can be partially addressed by making the operator's use of AI suggestions facultative, but attention to the format in which the suggestions are presented will also be needed.

4.1.4 Service development

In order to further develop the prognostic model, we utilized data from the year 2021, as train and validation sets, and the first quarter of 2022, as test set, to assess the model's generalization

capabilities. This was done to ensure the model's accuracy and relevance in predicting outcomes based on real-world production data. To simplify predictions and make them more accessible for operators, we reduced the output classes from 11 to 6. Instead of using half-minute increments for measuring the time remaining for set point V1 change, we opted for a one-minute frame. This decision was driven by the need for clear visualization and easier interpretation, as depicted in Figure 11.

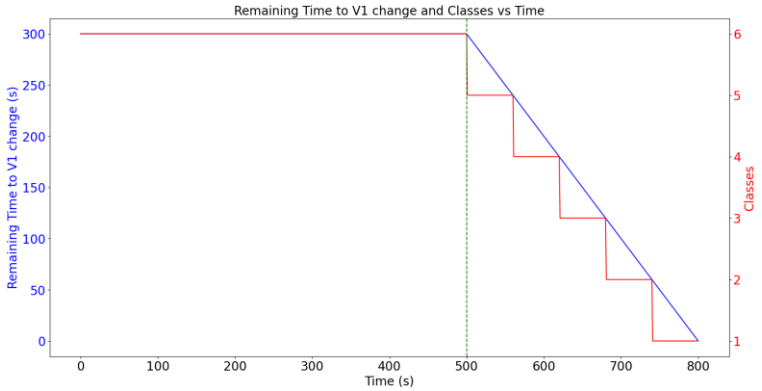

Figure 11: The remaining time to change with the new true classes used for training the model.

For the development of this model, we used data collected during the production process in 2021. The initial step involved clustering production times, which yielded multiple trajectories representing various production phases. In order to build a model capable of predicting changes in the V1 setpoint, we only considered trajectories that exhibited this change. This meant excluding stable productions from our dataset, leaving us with approximately 60% of the total trajectories.

We employed the same base architecture as before, which is based on a Multi-Layer Perceptron (MLP) - Long Short-Term Memory (LSTM) - Multi-Layer Perceptron (MLP) structure. This hybrid architecture combines the strengths of both MLP and LSTM networks, allowing the model to effectively learn and capture complex patterns within the data. MLP layers provide the capacity to learn non-linear relationships, while the LSTM layers enable the model to retain and process information over extended time periods.

The performance of the prognostic model is illustrated through confusion matrix heat maps generated using multiple trajectories from the first quarter of 2022. These heatmaps display the predicted values against the true values (see Figure 12).

While the overall accuracy of the model exceeds 60%, it is important to note that the previously encountered issue of only predicting one class has been resolved, enabling the model to predict deviations more effectively. The relative loss in accuracy can be attributed to some values not being situated directly on the diagonal, indicating that there may be occasional discrepancies between the predicted and true values. However, these discrepancies are generally minor, with predictions often being slightly early or late compared to the true values. The overall results are still considered to be satisfactory, as the model is capable of capturing the essential dynamics and changes in the V1 setpoint.

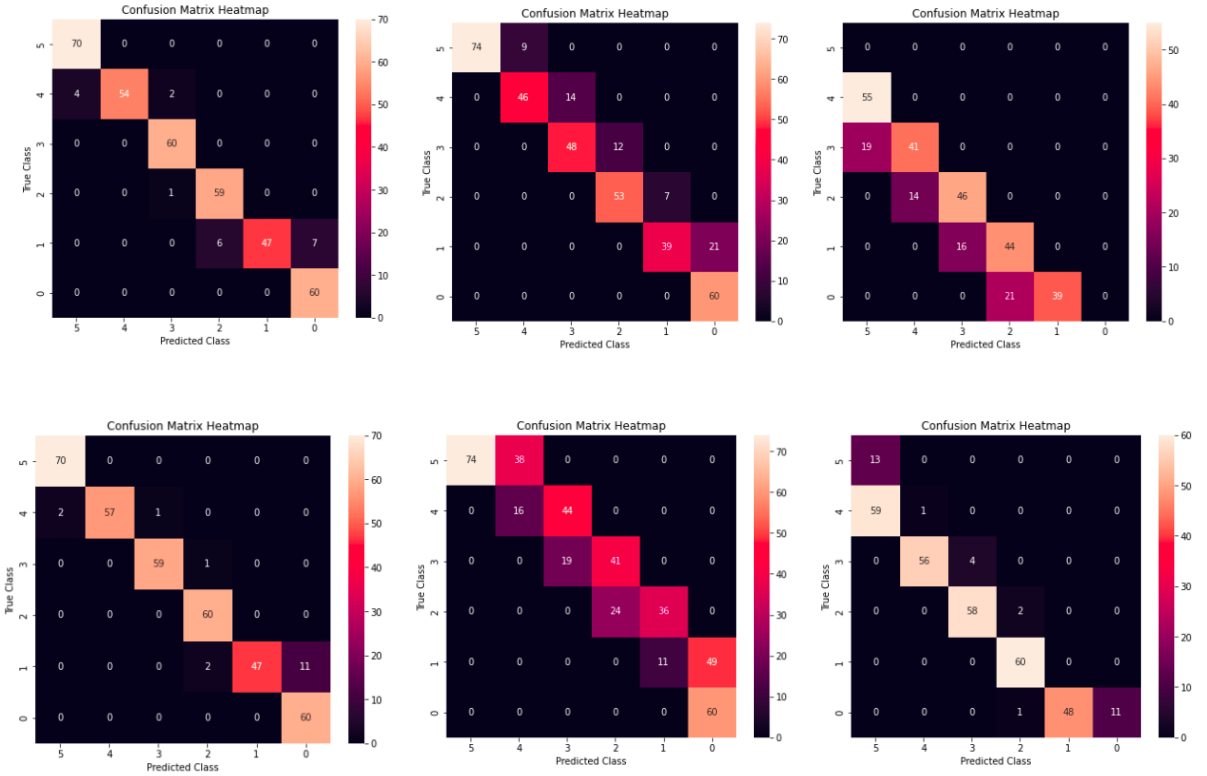

Figure 12: Confusion matrix for the 6 classes (see Figure 11) of the prediction model.

4.1.5 Deployment in AI-PROFICIENT platform

The goal was to develop a model specifically designed for prognostics. This model would be preceded by another model specializing in diagnostics, making the learning process more efficient and resulting in smaller, more manageable models. The implementation of such models on the edge would enable real-time monitoring and analysis, ultimately improving the overall production process. By employing a two-stage approach, with separate models for diagnostics and prognostics, we can effectively identify issues in the production process and predict potential changes in the V1 setpoint. This streamlined system will not only enhance the accuracy of predictions but also facilitate easier implementation and interpretation for operators in a production setting. Figure 13 presents the corresponding architecture.

At some point during the deployment of the prognostic algorithm it was detected that the V1 values were not truly representative of a relaxed product on the production line. This finding led to the reconsideration of the prognostic/diagnostic model's utility, as the V1 values may not be a reliable indicator of the product's actual state or quality during production. Consequently, any model built upon these values would not be able to deliver accurate assessments or predictions regarding product quality or potential issues within the production process. Under such circumstances, and, considering the potential harm that providing misleading predictions could produce to operators, it was decided to cease the deployment of the algorithm in the AI-PROFICIENT platform. In any case, the prognostic algorithm and its value have been validated and would produce satisfactory results under good data quality conditions.

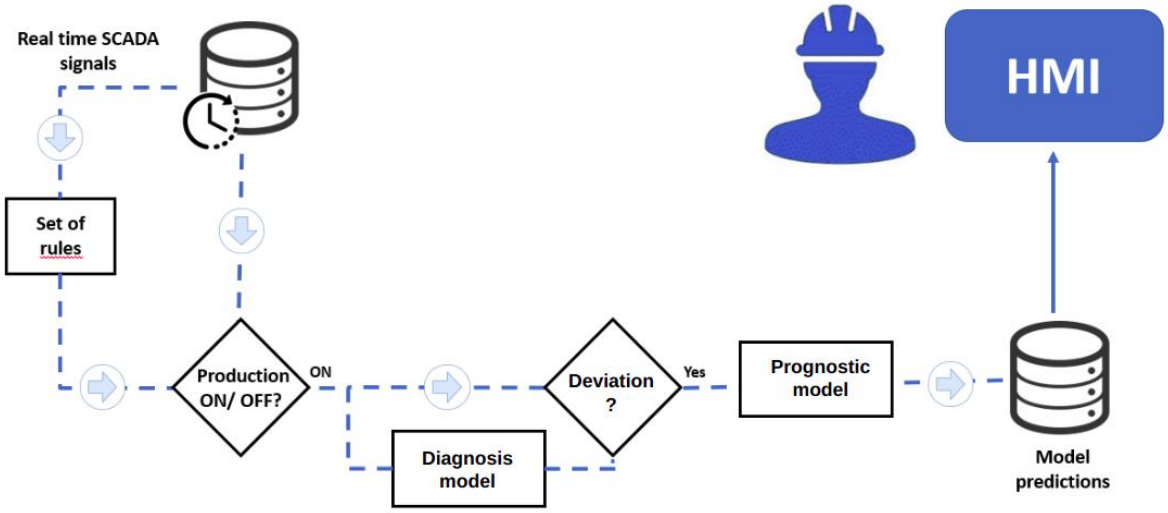

Figure 13: Implementation architecture of the CONTI-3 prognostic service.

4.2 CONTI5 – Cutting Blade wear diagnostics.

4.2.1 UC description

The aim of CONTI-UC5 is to develop a solution that will allow the operators and maintenance managers to know about the current wear state of the blade that is placed on the tread cutting system as well as giving some clues of how this wear will evolve in the future. In a daily basis, this blade keeps wearing until there is a point in which it produces bad quality cuts (hence, having to scrap the tread it cut) or it gets stuck. Consequently, the production line has to be stopped and the blade replaced incurring in the consequent downtime costs.

The first steps towards the development of the algorithms that would enable CONTINENTAL deciding when to change the blade according to wear estimations are detailed on deliverable D2.3 - Predictive AI analytics for component self-diagnostics, for further details, please refer to this deliverable's chapter 3.1.

As stated in D2.3, survival-like models try to model the probabilities of an asset being alive after certain usage/cuts. Yet, from the operator point of view, this information is not easily applicable on the production line and its counterintuitive. The survival value of 0.95 does not mean that the blade that is currently working is at the 95 % of its life, instead, it means that only 5 % of the blades do that many number of cuts. For that reason, a workaround that will allow the users to have a more understandable algorithm as well as to give some prognostic capabilities is needed.

4.2.2 Proposed solution

Considering the previous, it is decided to produce a Health Index that will reflect the health condition of the asset that implicitly considers the probability shown by the survival function. This Health Index (HI) rescales a certain number of cuts to a 0-1 scale. This way, the algorithm can be used not just for diagnosis purposes but for prognosis purposes, as, the estimated remaining HI can be known based on the expected future cuts.

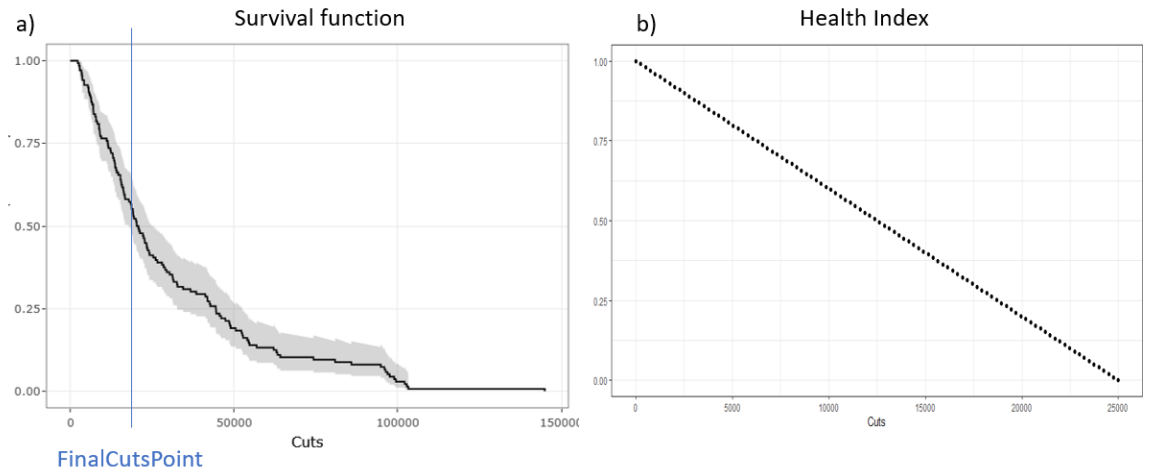

Figure 14: a) Survival function with a potential FinalCutsPoint. b) Example of a Health Index based on that Final Cutting Point.

For this approach to be valid there is a key point that has to be considered: determining the value of the Final Cuts Point. That is, identifying the amount of cuts that does not incur in too high cost caused by unplanned stoppages because of worn blades; or a point in which too many blades are changed early without considering the replacement cost. For the optimization of this point these two factors are considered:

- Cost of unplanned blade change
- Cost of a regular blade change

Given that using exact values to these factors is complex, different scenarios are simulated using the survival function and considering different quantiles of the survival function as Final Cutting Point. In each simulation, certain simulation parameters are considered:

- Monthly Cuts (MC): The average number of cuts carried out during a month, which is fixed value.
- Planned Final Cut Point (PFCP): The value of cuts that will be used by operator to replace the blade, which is varied in each simulation.
- Unplanned Blade change cost (UBCC): The cost associated to an unplanned blade change.
- Planned Blade change cost (PBCC): The cost associated to a planned blade change.

The schema of the simulation is depicted in the following Figure 15:

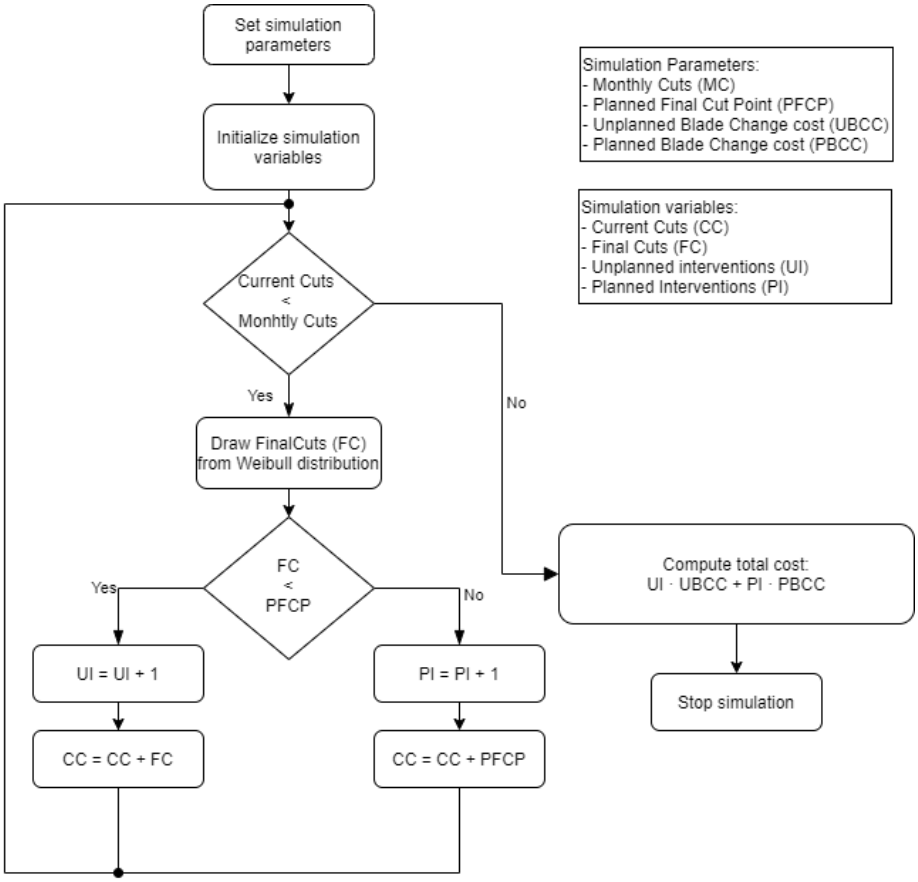

Figure 15: Schema of the simulation.

The simulation is run various times, with different PFCP values and considering different UBCC/PBCC ratios. The following Figure 16 depicts the results of the simulation:

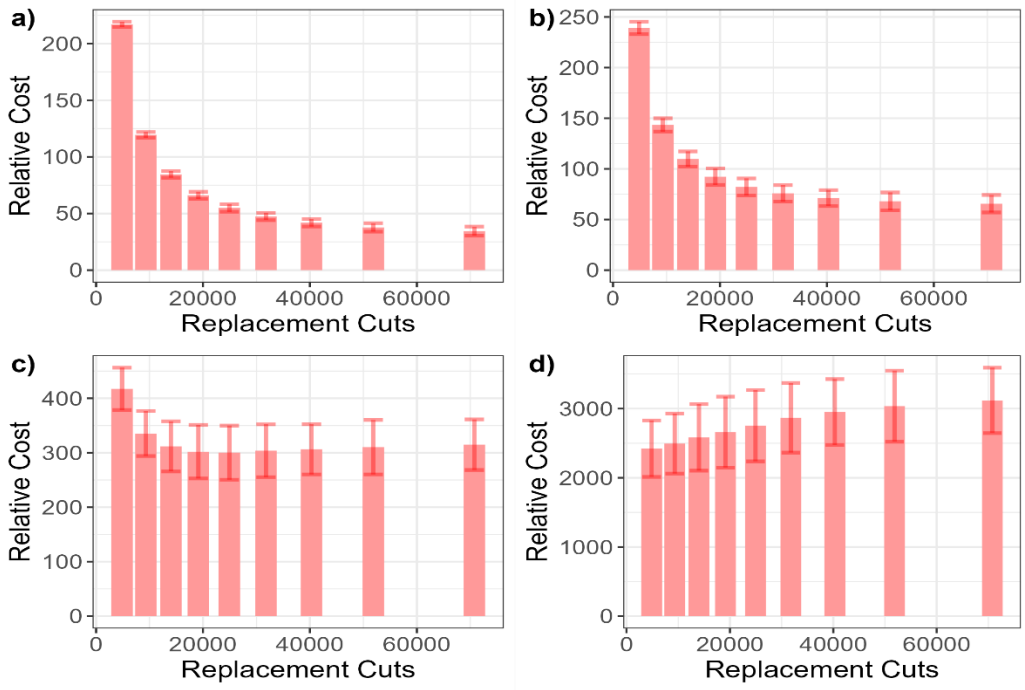

Figure 16: Simulation results under different UBCC/PBCC ratios. Bars represent the quantile values (or PFCP, which is equivalent). a) UBCC/PBCC = 1 b) UBCC/PBCC = 2, c) a) UBCC/PBCC = 10, d) a) UBCC/PBCC = 100.

Given the distribution of the data and, considering the results of the different UBCC/PBCC scenarios, it is clear that, unless the incurred cost when the blade change is unplanned is much greater (more than ten times the cost of a planned one), there is no economic incentive to replace the blade before it breaks. This could be partly caused due to a bad quality of the data (which comes from the scrapping of free text) as it has very long tails on the distribution (see the following Figure 17). However, it is not possible to trace back or to acknowledge if the data records are actually valid.

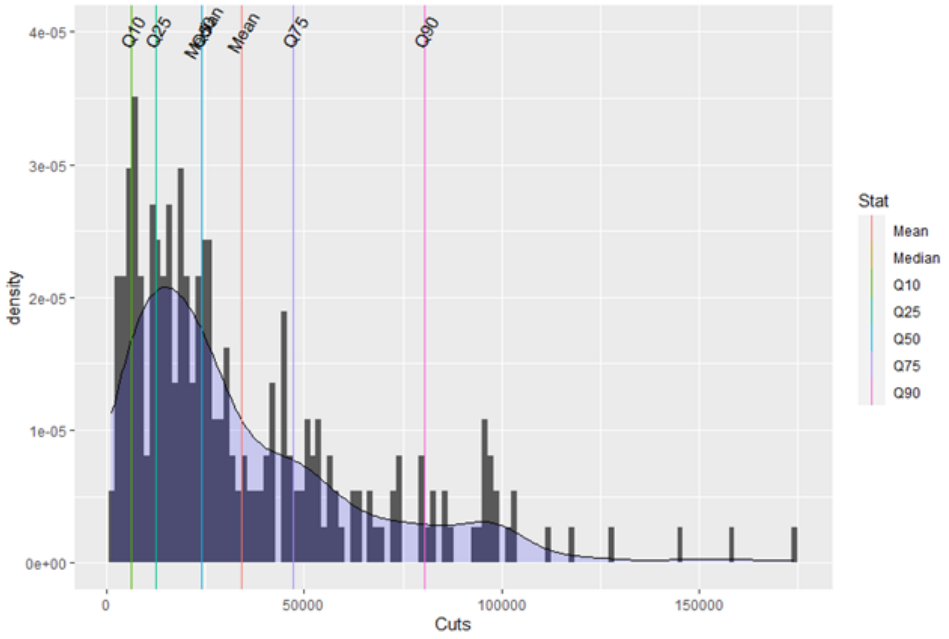

Figure 17: Density plot of the number of cuts carried out by the blades on the dataset.

At the same time, it is difficult to quantify the actual estimations of a planned and an unplanned blade change. In addition, there are other aspects (such as the stress felt by the operator when an unexpected production line stoppage occurs) that could be improved regardless of the economic justification. For that reason, the end-to-end approach is developed.

For that purpose, the Health Index values of the different scenarios are computed. In each of them, the final cutting point is considered the point at which the most economical benefit would be reached according to the simulation. Then, the intersecting point between the FinalCutsPoint and the Survival curve is used, this intersection point is then rescaled to 1-0 scale. This way the cuts consider the probability of being alive and do not just increase gradually till reaching the FinalCutsPoint. The different Health Indexes are depicted in the following Figure 18.

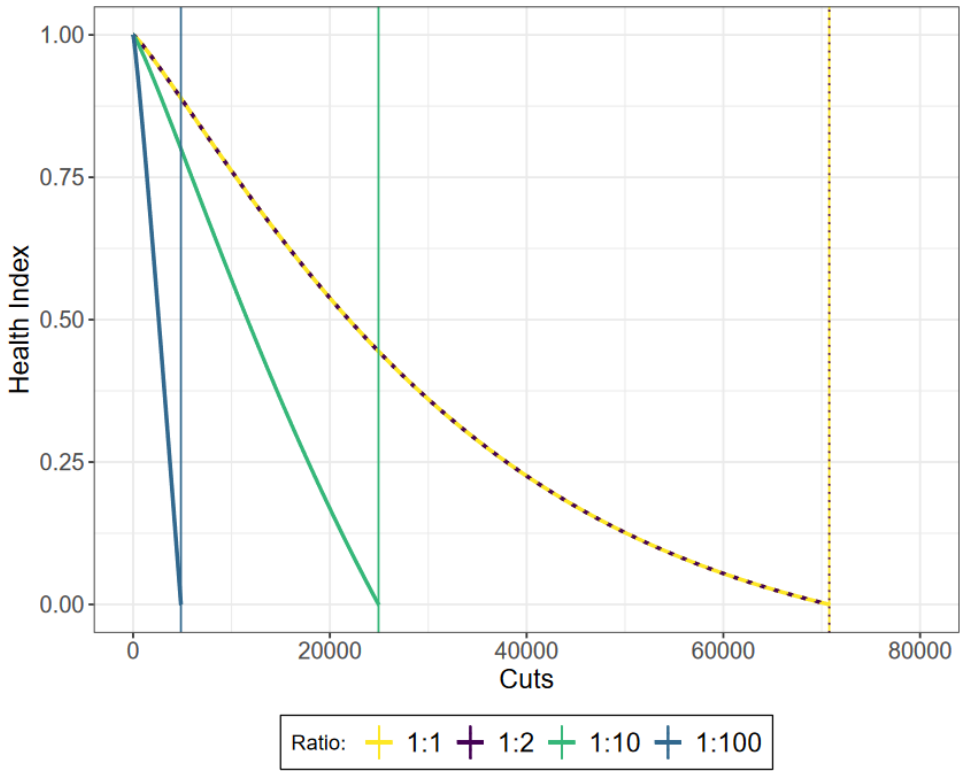

Figure 18: Health Index development for different Final Cutting Points.

It is noteworthy that, in practice, the Health Index is almost equal to the survival curve. If the Final Cutting Point is close to the point at which the survival function reaches 0, then, the Health Index will be exactly equal to the survival function. However, this is not how the Health Index is meant to be used. Ideally, the FinalCuttingPoint is reached way before the survival probability is 0, and, in such circumstances, the curves will differ. In such scenario, the user will have a more intuitive indicator that better reflects the potential condition of the blade, instead of having to interpretate the probability of the asset of being alive.

4.2.3 Deployment in AI-PROFICIENT platform

At the writing of deliverable, some changes have been made on the data pipeline on AI-PROFICIENT platform as described in D3.5. The aim of these changes is to guarantee an improved quality of the data that is stored on the databases. With these changes it is expected that, as the models will be periodically retrained in the future, the outcomes of the models will be more accurate.

For more details of the deployment of this algorithm please refer to D3.5: Future scenario-based decision making.

5 Conclusions

Providing prognostics at the edge for critical assets is a must to be for deploying predictive maintenance. The expected outcome of Task 2.4 is to develop some services enabling prognostics at the edge for AI-PROFICIENT platform as well as some scientific contributions in the field of edge prognostics. This deliverable reports these developments.

Starting with Scientific development for prognostics leveraging deep learning MLP-LSTM-MLP model, the task then develop 2 services for CONTI-3 and CONTI-5 use cases. Table 5 summarizes the UCs AI technologies where edge prognostics have been developed.

Table 5: AI enhanced edge prognostics *UCs.*

UC	Type of AI enhancement technique	Responsible
CONTI3	Deep Learning Model for prognostics	UL
CONTI5	Blade health index model for prognostics	TEK

As already seen through literature review, AI plays a major role in prognostics. AI-PROFICIENT brings some new elements on the table. Indeed, thanks to the work performed in task 2.4 and the provided use case, some relevant advances have been provided including publication and development in use cases.

6 References

An, Z., Li, S., Wang, J., & Jiang, X. (2020). A novel bearing intelligent fault diagnosis framework under time-varying working conditions using recurrent neural network. ISA Transactions, 100, 155-170. doi:10.1016/j.isatra.2019.11.01

Jardine, A. K., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, *20*(7), 1483-1510.

T. O. Kvalseth, Entropy and correlation: Some comments, IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no. 3, 1987.

Leukel, J., González, J., & Riekert, M. (2021). Adoption of machine learning technology for failure prediction in industrial maintenance: A systematic review. *Journal of Manufacturing Systems*, *61*, 87-96.

Peng, Y., Dong, M., & Zuo, M. J. (2010). Current status of machine prognostics in condition-based maintenance: a review. *The International Journal of Advanced Manufacturing Technology*, *50*, 297-313.

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In 2008 international conference on prognostics and health management (pp. 1–9).

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 957391.