Ref. Ares(2023)3080310 - 02/05/2023

AI·PROFICIENT

Artificial intelligence
for improved production efficiency,
quality and maintenance

Deliverable 2.3

D2.3: Predictive AI analytics for component self-diagnostics

WP2: Smart components and local AI at system edge

T2.3: Field-level automation and control from system edge

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: D2.5: Local automated control for quality assurance

Lead Beneficiary:	TEK
Due Date:	30/04/2023
Submission Date	02/05/2023
Status	Final ~~Preliminary~~ ~~Draft~~
Description	Introduction of AI edge technologies used on the edge and evidence of the demonstration scenarios used to validate such technologies within AI-PROFICIENT.
Authors	Kerman Lopez de Calle (TEK), Alejandro Muro (TEK), Alexandre Voisin (UL)
Type	Others - Report
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	KL	31/03/2023
0.2	Revision and suggestions	AV	10/04/2023
0.3	Reviewed version	KL	18/04/2023
1.0	Final version	KL	26/04/2023

Executive Summary

Deliverable D2.3 - Predictive AI analytics for component self-diagnostics 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 diagnosticate the assets in which they are embedded or run.

Diagnosis 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; 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.

This deliverable aims to disseminate the different diagnosis strategies that are used for asset monitoring and how the AI could be used to improve those strategies. In addition, different use cases were IA based diagnostic technologies have been validated are presented, together with the corresponding technical dissemination of these advances.

1 Introduction

This deliverable disseminates the progress that have taken place during the course of Task 2.3 - Field-level automation and control from system edge. This task has focused on the development of diagnosis system that could be run on the edge, implying the monitored assets themselves could be updated so that they could produce their own diagnostic information. This information could be later used by either other services or by the humans supervising the correct functioning of the assets.

This deliverable introduces the different self-diagnosis approaches, and then, the different ways the AI technologies could contribute to those self-diagnostic systems is explained. Finally, the applied diagnostic technology contributions developed in this project are disseminated, linking them to the Use Cases presented in this project when possible.

At the stage of identifying the required services to solve the requirements and demands of each UC (during the elaboration of D1.3 in WP1), it was identified that 5 UCs would require the contribution of self-diagnostic systems, as displayed in the following Table 1.

Table 1: 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								
2.3 – Self-Diagnostics	TEK	UL	TEK/UL/IBE	INOS	TEK/IBE			

Since the definition of those links the project has matured and the content of D1.3 cannot be longer considered updated. In that sense, according to the contributions that the different partners have made to this deliverable, the updated view of the previous table would be the one presented in the following Table 2.

Table 2: 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								
2.3 – Self-Diagnostics			TEK					

The technical developments carried out in this task were envisioned to contribute to the *Diagnostic and anomaly detection* functionality has reported in D1.4 deliverable (see Table 3).

Table 3: 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

As per the diagnostic service, the following Table 4 contains an updated description of how the diagnostic service is.

Table 4: Updated S_DIA service description.

Service ID	S_DIA
Service input and dependency on other services:	This service is developed taking as a basis the IIoT sensor installation and the acquisition & pre-processing steps carried out by the acquisition and pre-processing service. It contributes to the UCs in which a diagnosis and anomaly identification of the assets is required on the edge, such as UC CONTI5. Besides the application of the service in tyre industries and chemical industries, it has been validated in other scenarios, such as, for example a fuse manufacturing industry. In each of the UCs slight variations of the diagnostics systems have taken place according to the complexity and need of the UCs. In that sense, anomaly detection, fault diagnosis and health index type systems have been developed.
Service output:	Depending on the diagnosis system: - Anomaly or OK conditions - Specific fault name or OK condition - Health Index representing current wear status
High level service description:	The aim of this service is to check the assets are operating under normal conditions. When anomalies are detected, the service will raise an alarm. The alarms raised by the service will be then managed by other systems or operators so that catastrophic failures are avoided.

Regarding the rest of the content of this deliverable, section 2 will present the various approaches used for self-diagnosis condition monitoring systems, providing an overview of the different techniques and technologies used. Section 3 will describe how these technologies have been applied and validated in different use cases, offering real-world examples of their practical use. Finally, section 4 will summarize the results of this task and deliverable. The comprehensive overview presented in this deliverable on self-diagnosis condition monitoring systems and their applications is expected to provide valuable insights to the readers.

2 Self-diagnostic systems: Definition and AI based improvements.

2.1 Existing approaches to Self-Diagnosis

Self-diagnosis in condition monitoring refers to the ability of a system to monitor its own variables and detect deviations that could indicate a developing fault. Depending on the degree of accuracy of the output of the self-diagnosis system, three different types of self-diagnoses can be distinguished:

- **Anomaly detection** refers to the process of identifying deviations from normal system behavior, which can indicate the presence of a fault or failure. This approach often involves the use of statistical analysis, machine learning algorithms, or other similar techniques to identify abnormal patterns in system data.

- **Health-index** based approaches, on the other hand, involve the use of a calculated health index that provides a quantitative measure of the system's health status. This index is typically based on a combination of sensor data, performance metrics, and other relevant information that can provide an overall picture of the system's condition.

- **Diagnostic** based approaches involve the use of explicit diagnostic models that are designed to identify the root cause of a fault or failure. These models are typically based on a detailed understanding of the system's components and how they interact with one another, and may involve the use of expert knowledge, fault trees, or other similar techniques.

Each of these approaches has its own strengths and weaknesses, and the choice of approach will often depend on the specific requirements of the application, the available data, and the desired level of diagnostic accuracy. In addition, it is also possible to combine various of the previous to develop more holistic solutions. This would be known as a hybrid approach.

2.2 The role of AI on self-diagnosis

It is noteworthy to mention that, when speaking about AI based self-diagnostics this deliverable only considers the so-called data-base models, those models that are created by means of modelling data coming from the system/asset. There are alternative diagnostics strategies such as physics-based modelling that are not considered in this deliverable.

Regarding these data-driven or data-based models, there mainly two branches of machine learning that contribute to the core of self-diagnostic models:

- **Unsupervised models** are used when there is no data that reflects the bad condition of the system (no labelled data exists). In such a scenario, some other assumptions need to be made to model the existing data. Traditionally this is the case of the statistical process control (SPC) that assumes the data is normally distributed and detects deviations from that distribution. However, the limitations of the traditional univariate models have been overcome by more recent one-class support vector machines, isolation forests and other clustering algorithms. Unsupervised models are mostly used for anomaly detection as, typically, data related to faults is scarce.

- **Supervised models** are used when the data that is going to be modelled has a target variable or label. At the same time, this label can be of a different nature. If it is a continues variable, we will talk about a **regression problem** (for example, mm of wear); otherwise, if the target variable is categorical, it will be a **classification problem** (for instance, the type of fault). There is an extensive number of algorithms that serve for regression of classification problems. Typical classification algorithms include logistic regression, decision trees, random forest, and support vector machines. More recently, neural networks, Deep Learning models and boosted trees are gaining more attention on the literature. Most of the previous algorithms (but the logistic regression) have their counterpart version that are adapted to work on regression scenarios. Both Health Index based, and diagnostic approaches rely on supervised models, as they require signal data together with the labels (wear percentage or type of fault).

The most important factor in order to choose among the potential algorithmic approach is indeed the type of data that will be available for the development of the diagnosis system. When only good condition data will be available, unsupervised models will have to be chosen. On the contrary, if labelled data (faulty data, degradation evolution data, etc.) will be acquired, it will be better to opt for a supervised approach, as it will provide a higher degree of detail in the diagnosis.

3 AI based predictive AI analytics for component self-diagnostics: Demonstration scenarios.

The different technologies explained in the previous chapter have been instantiated in different UCs. The following sections describe the different use cases and the approaches followed in order to equip the systems with self-diagnostic capabilities.

3.1 CONTI-UC5 - Cutting Blade wear diagnostics.

3.1.1 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. 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.

In sight of that, CONTI wants to develop an intelligent system that would let the user know the approximate wear status of the blade. This way the operators will be able to replace the blade before it starts to produce low quality cuts, or it stops the production line. At the same time, it is desired that the diagnostics system could, somehow, support the creation of scheduled replacements that would reduce the amount of curative mode changes of the blade.

3.1.2 Proposed solution

The first challenge of this use case is the low availability of the data. In principle, there is no direct way of knowing the exact amount of cuts a blade has carried out before it wore out. In that sense, one of the major challenges of this UC relies on the "data-fusion", that is, retrieving data from different databases and combining it to produce meaningful dataset that could be used by the algorithms.

First stages of algorithm development

Given that the data-fusion process was challenging, the development of a potential algorithm was started with simulated data.

For that purpose, a literature review was carried out identifying works that had measured the wear development on similar blades. According to the review [1], most of the works showed constant wearing speeds (see following Figure 1). However, this was not always the case, less frequently, some works showed varying wear speeds, as in Figure 2.

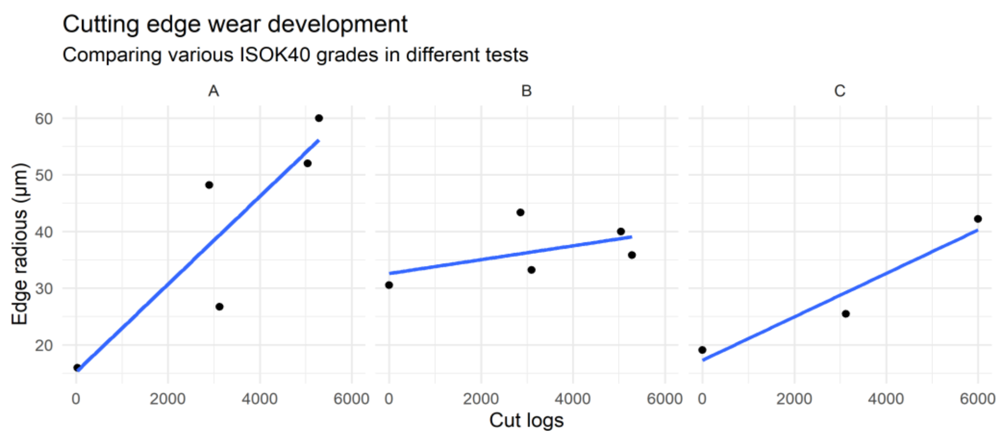


Figure 1: Example of cutting-edge wear development. Adapted from: Ekevad, M., Cristóvão, L., Marklund, B., 2012. Wear of teeth of circular saw blades. Wood Material Science & Engineering 7, 150–153

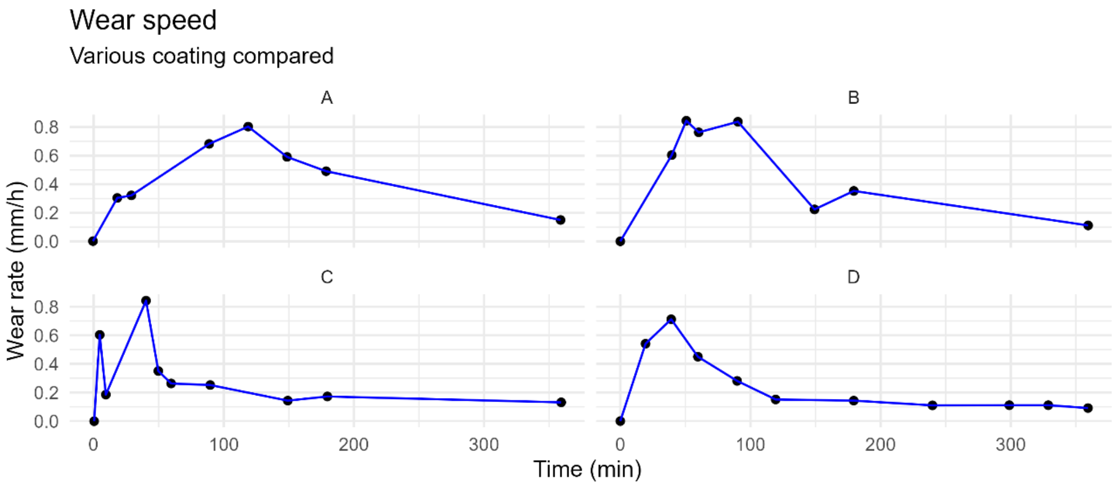


Figure 2: Another example of wear speed in blades with different coatings. Lau, K.H., Mei, D., Yeung, C.F., Man, H.C., 2000. Wear characteristics and mechanisms of a thin edge cutting blade. Journal of Materials Processing Technology 102, 203–207

Consequently, it was decided to design a model that would try to identify the wear-speed based on the data from the database. Different potential wear-speed patterns were tested and the one that best fits the database would be chosen. The following Figure 3 displays the different patterns that were tested on the synthetic data.

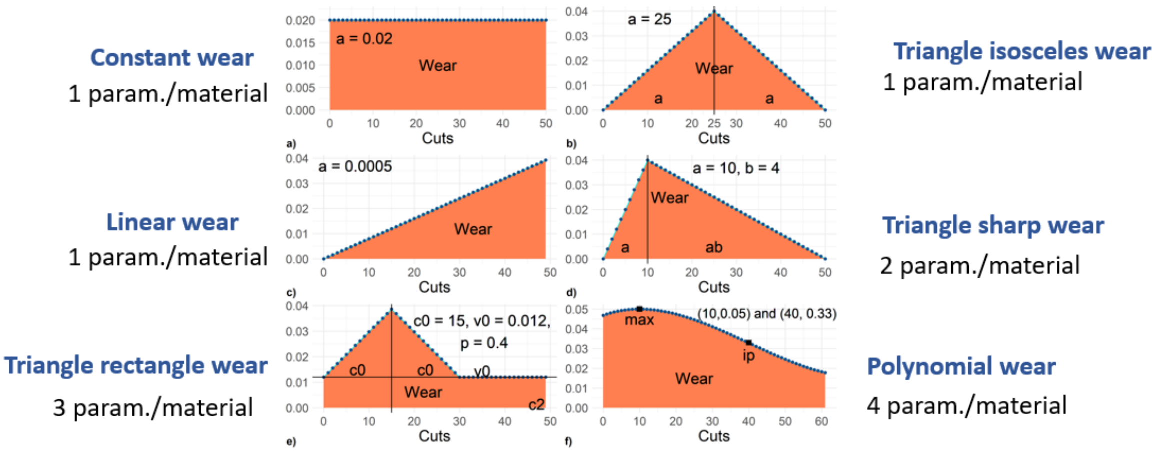


Figure 3: Different wear speed patterns tested on the synthetic data.

These wear-speed patterns would represent the health-index of the blade (inverse of the wear-index), that could let the operators know the approximate wear status of the blade. At the same time, each recipe (each different material to be cut) would have its own parameters, meaning the mechanical differences of the cut materials would be considered (which was one of the major sources of variability according to the literature survey).

The approach was validated by creating some synthetic data with noise from different distributions. Using some wearing patterns and the noise, a database was created (similarly to the one expected to have at CONTI-UC5), this database consisted of 200 blade lives, as the one presented Table 5.

Table 5: Example of a synthetic blade live.

Chronology	Recipe	Cuts
1	4	41
2	2	12
3	4	5
4	2	18
5	1	169
6	3	21
7	4	1

Once the database was ready, an optimization was launched to identify the different potential parameters that might have created the database, without knowing the exact wear profile used to create the database. According to the results, it would be possible to create a quite robust model that would represent the database data. However, the amount of noise on the original wear distribution would greatly affect the accuracy of that model. Further details on the approximation can be found in the work [1] presented by Tekniker in the 14th IFAC Workshop on Intelligent Manufacturing Systems.

Real data retrieval

In parallel to the theoretical validation of the proposed model, by cross-referencing data from different databases it was possible to build a dataset similar to the one required to test the approach. For that purpose, maintenance records (daily created free texts containing the maintenance actions carried out) were parsed so that the blade replacements were identified. At the same time, the signals representing the amount of good treads, the bad treads and the recipes were combined so that the total cuts per recipe and blade life could be computed. Nevertheless, it was detected that given the maintenance logs had low precision on the dates recorded (only the day and the shift could be known) the data had some noise.

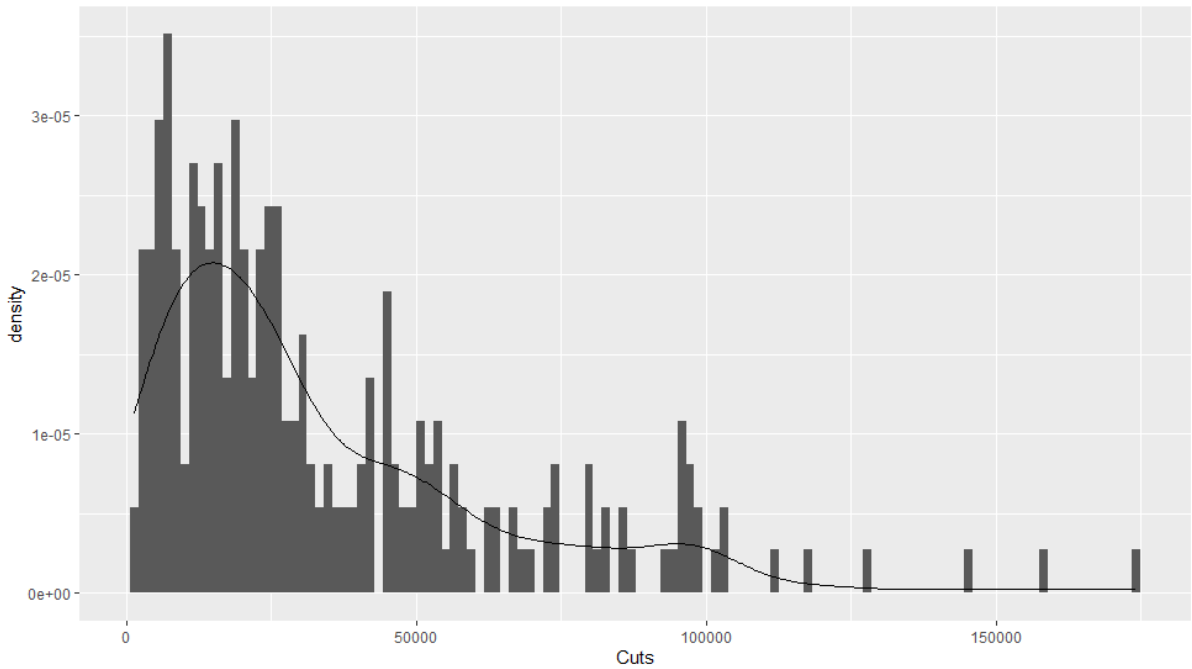


Figure 4: Histogram of final number of cuts of the blade life database.

As the previous Figure 4 shows, the blades recorded in the database follow a Weibull-like distribution. A total of 255 blades are available and there are some potential outliers that have “too many cuts” which could be caused by the non-robust data fusion system. At the same time, the existence of too many recipes is identified which, as it is greater than the number of blades, will turn the modelling through the proposed approach challenging (due to the number of parameters that need to be estimated).

Algorithms comparison

To implement the best possible solution, the suggested approach was compared with more traditional approaches, such as a Weibull based reliability model. The problem of this kind of approaches is that their aim is different. Instead of modelling the wear that occurs on the blade, they try to approximate the survival function of the whole fleet, that is, the probability of an asset of being alive after certain number of time/cuts. One of the shortcomings of this kind of models is that they cannot be improved by additional information unless it is included through adding binary factors (not with many different values) or single continuous variables.

At the same time, it is well known that the cutting process is influenced by many external factors [2], which, in an industrial scenario such as the one in UC5 would be interesting to consider. On that regard, it could be considered that the major source of variability is the working-piece, that is, the mechanical properties of the material being cut. This kind of information could be transferred to a survival model by creating binary variables for the different materials (recipes) cut by a blade. But, given the vast amount of recipes and combinations, the number of samples per recipe is quite small to enable modelling such a large number of curves. Consequently, the fitting would be poor or even not possible. Alternatively, the usage-based model requires a smaller number of parameters and can be fitted with less effort and data requirements.

In order to compare the usage-based approach with the survival like modelling, defining out of the box metrics has been needed. With these metrics, the potential performance improvement of the usage-based models has been tested.

The underlying assumption of the tests is simple: “If cuts are not equivalent in reality (because the materials have different hardness), is it possible to provide an alternative measuring dimension (Equivalent Cuts) that considers the materials that were cut and improves the survival fitting?”. To test such assumption, a regular fitting to a Weibull Distribution is compared to a fitting of the same model after the Cuts have been recomputed with the usage model. To validate the approach in a statistically meaningful way, the data is first split in train/test split.

The results of the fitting are compared using three different metrics:

- **Kolmogorov-Smirnov Hypothesis Test**, which gives a comparison of cumulative distribution functions, and the test statistic is the maximum difference D. The underlying assumption is that the data follows a certain distribution (in our case the Weibull distribution) – Null hypothesis. Thus, we compute:
    - **p-value** associated with the null hypothesis that the data follows a Weibull distribution. It reflects to which extent the cuts follow a Weibull distribution. Values greater than 0.05 imply that the null hypothesis holds with a 95 % confidence interval.
    - **Kolmogorov-Smirnov statistic** for the cumulative function given by the Weibull distribution: $\mathrm{D} = \max_{u}\left(F_x(u) - F_y(u)\right)$. It reflects the maximum difference between both cumulative functions. The greater its value the worse the fitting.

$$S(\boldsymbol{Cuts}) = \exp(-\boldsymbol{Cuts}/\lambda)^{k},$$
$$\log(-\log(S(\boldsymbol{Cuts}))) \propto \log(\boldsymbol{Cuts})$$

Where y = log(-log(S(Cuts))).

- **Coefficient of determination** (R^2), which is another measure of goodness of fitting. As it is assumed that the cuts follow a Weibull distribution, the R^2 is computed by linearising the survival function of the fitted Weibull distribution. This metric ranges from 0 to 1 where values close to 1 reflect a good fitting whereas values close to 0 reflect a poor fitting.
  Given the transformation of survival function is linearly dependent on log(Cuts).

  Coefficient of determination can be computed:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

Finally, three scenarios are compared:

1. Directly modelling through fitting a Weibull distribution on Cuts
   a. KS-Test
      i. p-value = 0.08256
      ii. D = 0.084735
   b. $R^2 = 0.9683114$
2. Recoding Cuts with Triangle-Rectangle wear-speed model and Weibull fit on the Equivalent Cuts
   a. KS-Test
      i. p-value = $8.042 \cdot 10^{-7}$
      ii. D = 0.18214
   b. $R^2 = 0.8609486$
3. Recoding Cuts with Constant wear-speed model and Weibull fit on the Equivalent Cuts
   a. KS-Test:
      i. p-value = 0.2468
      ii. D = 0.068624
   b. $R^2 = 0.9742978$

According to the results, slightly better results than direct usage of Weibull distribution are obtained when Equivalent Cuts are computed using a Constant wear-speed model. As the p-value is greater meaning we can be more confident on the null hypothesis, i.e., the data follows a Weibull distribution. At the same time, the D value, is smaller, indicating both cumulative functions are closer, as well as the coefficient of determination value, that is also greater, validating the fact that the fitting improves. However, computation time and complexity of this model may result in complication in the deployment from a practical point of view. Thus, more simple models, which still offer good fitting results, provide an adequate initial solution to this particular problem.

3.1.3 Deployment in AI-PROFICIENT platform

Given the vast number of recipes that CONTINENTAL uses during production, managing all that variety is needed to enable the usage model. However, at some point the potential improvement that could be achieved by implementing the usage model is eclipsed by the effort needed to manage the variability of the recipes correctly. As the clustering of the recipes requires additional pipelines in order to reduce the number of parameters that the usage model needs to fit.

For that reason, it has been decided to finally implement the survival like model on the AI-PROFICIENT platform. This model will be simpler (and hence more robust) and will equally benefit from the existence of the human reinforcement system so that it can improve over time.

3.2 PHME-2021 - Fuse production line fault diagnosis (PHME-2021)

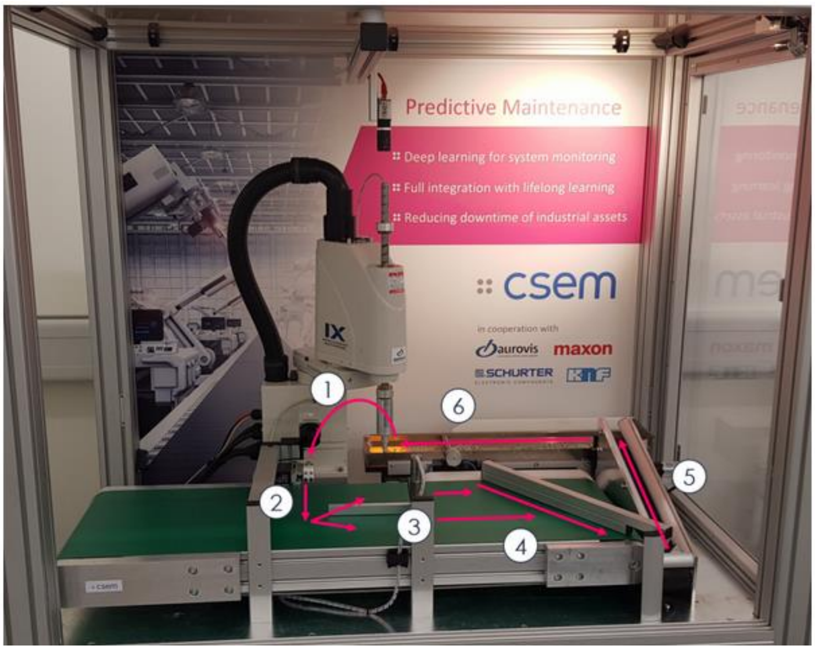
3.2.1 UC description

This use case is presented by the Prognostics and Health Management Society in Europe. This society wants to promote the development, growth, and recognition of prognostics and health management (PHM) as an engineering discipline and has worldwide recognition in that field.

In the year 2021, they organized a data-challenge, an engineering problem that consisted of building a diagnostics system for a fuse production line in which some faults occurred. This resembled a typical component of a large-scale quality-control pipeline of a production line. The experimental bed, courtesy of the Swiss Centre for Electronics and Microtechnology (CSEM), was used to generate data similar to a real-world industrial manufacturing line.

Besides the normal operating condition (class 0) 8 additional faulty states were recorded in the data. Participants were not given any clue regarding the type of fault that each different class could have, in fact, part of the challenge was identifying the signals related to each fault.

The following **Erreur ! Source du renvoi introuvable.** depicts the test rig used for the data generation.



Exp.	Nº
0	70
2	4
3	4
4	3
5	4
7	4
9	4
11	3
12	3

Figure 5: Left) Test rig used for data generation. Right) Number of tests provided for each faulty state.

The challenge had 3 different goals:

- Creating a diagnosis model that would correctly classify experiments.
- Provide diagnosis in the shortest time.
- Identify the root causes of the faults.

For simplicity's sake, only the diagnosis and the time consideration are explained in this deliverable. Nevertheless, the reader can refer to the following publication for more detail [3].

3.2.2 Proposed solution

During the first exploratory analysis, typical issues occurring in industrial scenarios were identified. Including:

- **Missing values**: The sensor records had data that had to be dealt with.
- **Class imbalance**: As it occurs in most of the manufacturing/industrial scenarios, the target classes are imbalanced in favor of the nominal class, which is usually over-represented on the data.
- **Multicolinearty**: Given the high number of signals and indicators, it is quite common that they are linearly correlated, which is a problem for certain algorithms and means some information is redundant.
- **High dimensionality**: As previously mentioned, the high amount of sensor and indicators leaded to a high dimensionality scenario, which can be challenging.
- **Experimental uncertainty**: Even if the test rig was the same, the sole fact of re-starting it leaded to different patterns on the signals. This variability was also something to take into consideration as some of the classes had only a couple of experiments.
- **Chronology in diagnosis**: Given that the goal was to provide a diagnosis in the shortest possible time, that meant to take into account the sequences of observations which is not very common in the machine learning field.

In sight of the previous issues, different strategies were tested. Finally, it was decided to use Last Observation Carried Forward (LOCF) technique to deal with missing values. Class imbalance was tackled by using SMOTE-Tomek to resample the observations and balance the classification problem and tree models were chosen for the diagnostic system as they usually work better for scenarios with correlated variables. Overfitting over the experiments was fixed by using leave one group out validations scheme, where, each time, only data from the same experiments were taken for training and testing each time.

Different machine learning algorithms were tested and, in general, the performance did not change much from one to another. Nevertheless, some of the classes were difficult to identify regardless of the employed algorithm. To solve that matter feature engineering was used. With the inclusion of new features, created by means of an exploratory analysis that served as basis to engineer the new features, it was possible to detect those difficult classes with a rather simple decision tree (DT). The final structure of the diagnosis pipeline is depicted in the following Figure 6.

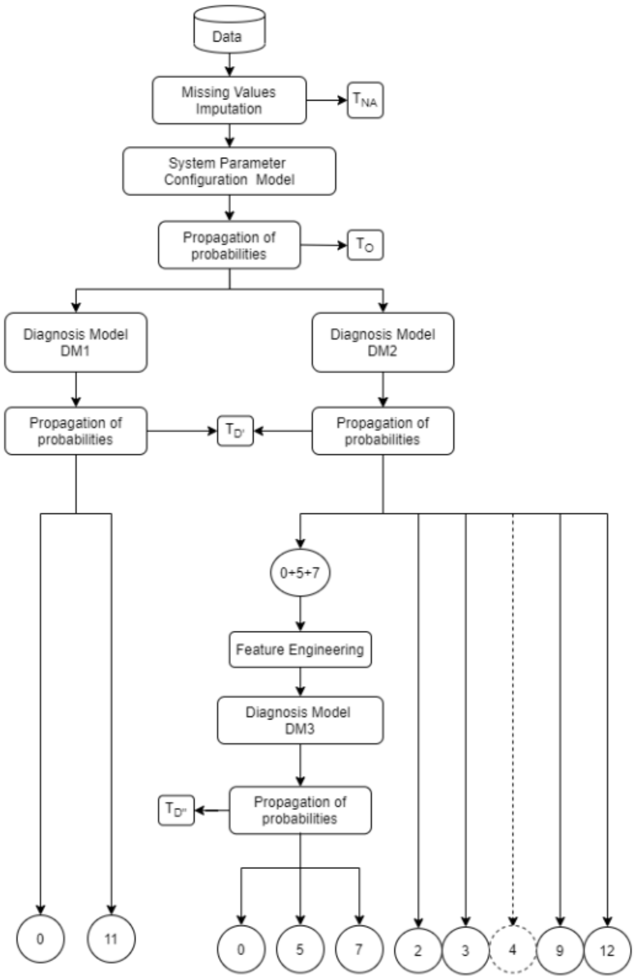


Figure 6: Diagnostics pipeline structure.

Regarding the shortening of the diagnostic time, it must be considered that the diagnosis algorithms was executed for each new observation, but the small changes in the signals could lead to completely different diagnosis values (see Figure 7: Left).

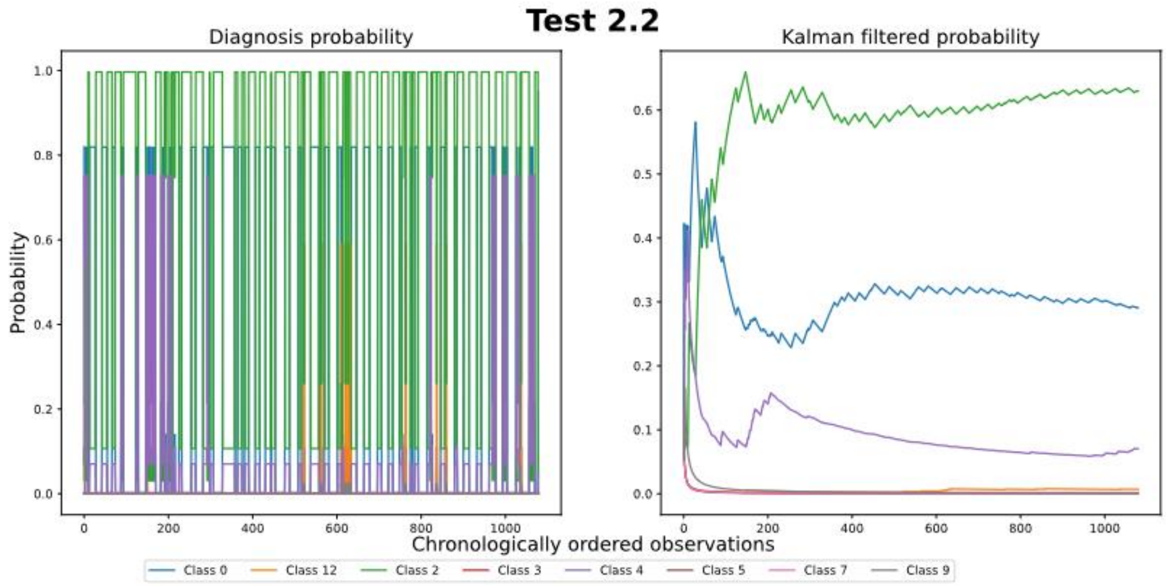


Figure 7: Left) Evolution of diagnostic values over time with no filtering. Right) Diagnosis filtered with a Kalman filter.

To avoid the oscillating behavior of the diagnosis algorithm a Kalman filter was fed with the probabilistic output of the algorithm. This filter would be able to identify the main trend of the diagnosis so that the prevailing diagnosis class could be detected, as Figure 7 Right) shows.

The solution here presented was the one that won the PHME-Data Challenge in 2021, proving the validity of the approach. More details can be found in [3], the journal paper were the solution was disseminated.

This solution belongs to a diagnostic type of problem were supervised algorithms (Decision Trees) have been used to build the diagnostic model that is able to distinguish among different faults. This is possible because the set of data contained fault related data. In such conditions, using supervised approximations was possible and preferred over unsupervised approaches that would be vaguer.

3.2.3 Deployment in AI-PROFICIENT platform

This UC is not related to AI-PROFICIENT platform, and hence, has not been deployed on AI-PROFICIENT platform.

4 Conclusions

Providing assets with diagnostic capabilities is an interesting improvement from the asset lifecycle point of view. As this ability leads to a reduction of both downtime and repairment times, as the root causes of the faults can easily be known and addressed.

In that sense, this deliverable identifies the different diagnosis approaches that exist, and the AI based technologies that can be used in different scenarios. On top of that, it provides examples of how to instantiate such technologies in applied scenarios, some related to UCs of the AI-PROFICIENT projects as well as other industrial scenarios.

The following table summarizes the UCs where edge diagnostic capabilities have been tested, the technological approach that was followed and the responsible partner.

Table 6: AI enhanced edge diagnosis UCs.

UC	Type of AI enhancement technique	Responsible
CONTI-UC5	Supervised usage-based Wear Index estimation	TEK
PHME-2021	Decision Tree based supervised modelling for diagnosis	TEK

Furthermore, the achievements in this task have been disseminated in different scientific publications and conferences, which serves as a further validation of the results obtained in this task. The following papers have been written in relation to T2.3: Field-level automation and control from system edge:

- K. López de Calle - Etxabe, E. Garate - Perez, y A. Arnaiz, «Towards a Circular Rotating Blade Wear Assessment Digital Twin for Manufacturing Lines», *IFAC-PapersOnLine*, vol. 55, n.º 2, p. 566, 2022, doi: https://doi.org/10.1016/j.ifacol.2022.04.253.
- K. López de Calle -Etxabe, M. Gómez - Omella, y E. Gárate - Perez, «Divide, Propagate and Conquer: Splitting a Complex Diagnosis Problem for Early Detection of Faults in a Manufacturing Production Line», *PHM Society European Conference*, vol. 6, n.º 1, Art. n.º 1, jun. 2021, doi: 10.36001/phme.2021.v6i1.3039.

In addition, it is expected that the final diagnosis solution for UC5 will be disseminated in another high impact journal.

Considering these demonstrations and findings, it is clear that edge level AI will play a key role on the industry as enabling technology for improved monitoring systems.

5 References

[1] K. López de Calle - Etxabe, E. Garate - Perez, y A. Arnaiz, «Towards a Circular Rotating Blade Wear Assessment Digital Twin for Manufacturing Lines», *IFAC-PapersOnLine*, vol. 55, n.º 2, p. 566, 2022, doi: https://doi.org/10.1016/j.ifacol.2022.04.253.

[2] V. Nasir y J. Cool, «A review on wood machining: characterization, optimization, and monitoring of the sawing process», *Wood Material Science & Engineering*, vol. 15, n.º 1, pp. 1-16, ene. 2020, doi: 10.1080/17480272.2018.1465465.

[3] K. López de Calle -Etxabe, M. Gómez - Omella, y E. Gárate - Perez, «Divide, Propagate and Conquer: Splitting a Complex Diagnosis Problem for Early Detection of Faults in a Manufacturing Production Line», *PHM Society European Conference*, vol. 6, n.º 1, Art. n.º 1, jun. 2021, doi: 10.36001/phme.2021.v6i1.3039.

Acknowledgements

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