Data-Driven Capability-Based Health Monitoring Method for Automotive Manufacturing

Alexandre Gaffet[1,2], Pauline Ribot [2,3], Elodie Chanthery[2,4], Nathalie Barbosa Roa[1] and Christophe Merle[1]

[1] *Vitesco Technologies France SAS 44, Avenue du Général de Croutte, F-31100 Toulouse, France*
alexandre.gaffet@vitesco.com
nathalie.barbosa.roa@vitesco.com
christophe.merle@vitesco.com

[2] *CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France*
elodie.chanthery@laas.fr
pauline.ribot@laas.fr

[3] *Univ. de Toulouse, UPS, LAAS, F-31400 Toulouse, France*

[4] *Univ. de Toulouse, INSA, LAAS, F-31400 Toulouse, France*

ABSTRACT

Testing equipment is a crucial part of production quality control in the automotive industry. For those equipments a data-based Health Monitoring System could be a solution in order to avoid quality issues and false alarms, that reduce production efficiency, potentially leading to huge losses. In manufacturing industries, a widely accepted index for evaluating process performance is the process capability, which assumes data following a normal distribution. In this article we propose a capability-based health monitoring method based on electrical test data. These data might vary according to the testing equipment, but also on manufacturing parameters. Gaussian Mixture Models (GMM) are used to model the data distribution exposed to equipment and parameter variations supposing that the hypothesis of normal distribution of the data holds. Two approaches are discussed for selecting the GMM number of modeled distributions. The first approach is based on the well-known Bayesian Information Criterion (BIC). The second approach uses a new multi-criteria index function. The health monitoring method is evaluated on real data from In-Circuit Testing (ICT) machines for electronic components at a Vitesco factory in France.

1. INTRODUCTION

In the automotive industry, the large amount of available equipment data promotes the development of data-driven methods when designing health monitoring systems. If complete information about the equipment health status and its maintenance operations is available, supervised methods are preferred. Conversely, if the information is uncertain or unavailable, unsupervised methods should be considered. Unsupervised methods are often based on statistical models that learn trends or anomalies in the data.

One major challenge of the monitoring is to find the appropriate health indicator for the problem. An approach could be to use a known health indicator with some interesting properties linked to the problem under study. For instance, (Baraldi, Di Maio, Rigamonti, Zio, & Seraoui, 2015) used spectral residual as input of an unsupervised algorithm. Another approach is to learn it using artificial neural networks (Ren et al., 2019).

In industry, Statistical Process Control (SPC) is a common approach to achieve a good quality of production. Links between the actual performance of production and specification limits are made using capability indexes (Wu, Pearn, & Kotz, 2009). With these indexes the health monitoring is based on the probability of finding out-of-bounds outputs from a process. Nonetheless, process capability indexes are usually used to supervise the production and do not distinguish the source of anomalies. Possible sources are the equipment, the product or its components. The goal of this article is to use

such capability indexes for health monitoring purposes and isolate the cause of an anomaly. To this end, the use of Gaussian Mixture Models (GMM) is proposed. Mixture modeling is a powerful statistical technique for unsupervised density estimation, especially for high-dimensional data (Mehrjou, Hosseini, & Araabi, 2016).

The GMM method requires the selection of a number of mixture components, a fitting algorithm and an initialisation method. To select the right number of mixture components at least two approaches are possible: the Maximum A Posteriori (MAP) approach and the full Bayesian approach. The MAP approach selects the most likely hypothesis according to the data and a prior distribution of the parameters and is often more tractable than full Bayesian learning (Montesinos-López et al., 2020). For the MAP approach, two criteria are presented: the Bayesian Information Criteria (BIC) and a Mixed-criteria method created by machine learning. In this work the Expected Maximisation (EM) algorithm is chosen as a fitting algorithm for GMM. Different initialisation methods are compared on real data in order to select the most appropriate method for the application.

In the considered industrial use case the tested products correspond to Printed Circuit Boards (PCB) where electronic components are placed using Surface Mount Technology (SMT). These components are then tested by In Circuit Testing (ICT) machines. Because of the uncertainty on the completeness of maintenance data, this paper proposes an original unsupervised health monitoring method using capability index as a health indicator. It uses tests values to detect faults linked to the equipment or with the products. The main contribution of the work is to show that these kinds of methods can provide useful knowledge for equipment health monitoring.

This paper is organized as follows. Section 2 is devoted to the presentation of the dataset and our capability-based methodology. Section 3 presents the MAP approach and the two proposed criteria to find the best number of mixture components. Section 4 discusses three different initialisation strategies. Application of the capability based methodology is provided in Section 5. Conclusions and future work are finally presented in Section 6.

2. METHODOLOGY

2.1. Industrial database

Vitesco production is grouped into product families, according to their design and application. Within each of these product families several product references can be found. For the sake of standardisation, the production of electronic cards uses substrates of similar size for each family. This substrate is called a panel. Panels are composed of identical duplicates of the same product. Electrical tests are designed for individual product references and have an associated test version. A PCB contains several components of different forms and sizes. The amount of components varies from 10 to 1000, however, each component needs to be tested individually. For that purpose one or more test steps are required. The usage of test versions allows to track any changes on the test steps and parameters. This product specific approach entails a tremendous amount of combinations to analyse, accordingly, the chosen monitoring methodology must easily adapt to product references and test versions.

The study uses two years of historical data from ICT equipment for several products. As illustrated in Figure 1, on the ICT two elements interact to make the process product specific. The first element is the ICT machine, which analyses the results. All machines can be used for all product families. The second element is the interface composed of a pneumatic bed-of-nails wired to fit the specific design of each product family and to test its electronic components.

Data coming from one interface, one test, one test version and multiple machines can be seen as a time series. These time series will be truncated since the production of one specific product and test version is often discontinuous. Furthermore, in the meanwhile the same machine can be used for testing other products with other interfaces. It is then impossible to extract a continuous time series from test data, which represent one of the main challenges of this dataset.

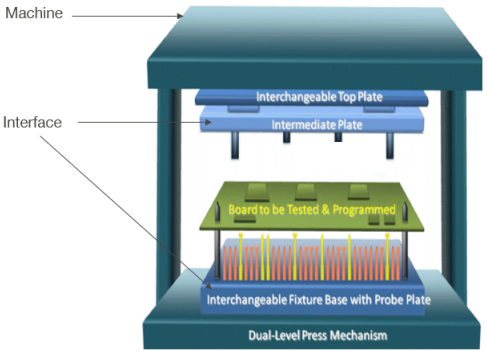

Figure 1. ICT equipment

Another challenge in the dataset is the large number of features. For each reference of product there are around 1000 tests. The developed a solution should be valid for all these tests, therefore, have a large power of generalization. Since ICT tests are supposed to follow a normal distribution, the use of a process capability index, defined in Section 2.2, seems to be adapted to evaluate the test process and the health of the testing equipment. For each test in our dataset the parameters are the ICT machine, the interface and the product position. All of these parameters are equipment related. Nonetheless, there are also product-related parameters. Indeed, for each product, the mounted electronic components, i.e. the raw ma-

terials, might come from different providers or have different values within the expected tolerance. These sources of variability cannot be controlled in the manufacturing process. For example, the temperature of the oven may be different and have an influence on the distribution of the component values. The idea is then to create groups of products that have the same parameters. In theory, splitting the data set by machine, interface and test will result in a normal distribution for each split. However, this neglects the influence of the raw materials and the manufacturing process. Nonetheless, including this information is difficult because the number of parameters to define each group is significantly large. The idea is to group the data samples according to the distribution created by the unconsidered parameters. In fact, multiple sets of parameters could lead to the same distribution. Henceforward, the hypothesis of normality for these distributions is made.

2.2. Capability

We propose a methodology based on process capability which is a well-known index in the industry. This index is calculated over a sample of data given an upper limit and a lower limit. These limits are defined by an expert during the design and pre-industrialisation stages of the product. Under some normality conditions, the index is directly proportional to the probability that a point in the sample is greater than the upper limit or lower than the lower limit. There are different capability indexes depending on the controlled process. We decide to use the CPk process capability index.

The choice of this particular capability index among all possible capability indexes is guided by the application and the dataset. In machine tool capability (CP), the size of the deviations from the average value of the process determines the location of the process within the specified limit. Nevertheless, as in our case the process is not always centred between the specification limits, the CPk index is a better index. Two other capability indexes could also be interesting: CPm, so-called Taguchi index (Hsiang, 1985) and $CPmk$, which combines the CPm and CPk indexes (Boyles, 1991). The Taguchi index measures the ability of a process to return values around a target value. We decide not to use these indexes because they require the target value of the components which in our case does not change in time, contrary to the specification limits. The CPk index is defined as follows:

$$CPk(X) = \min\left(\frac{UL - \mu(X)}{3\sigma(X)}, \frac{\mu(X) - LL}{3\sigma(X)}\right) \quad (1)$$

with X a sample of points, μ, σ, UL, LL the mean, the standard deviation, the upper limit and the lower limit, respectively.

2.3. Overview of the methodology

The proposed methodology is described in Figure 2. The incoming time series is defined by selecting one test version, one test, one period and one interface for that test version. Production periods are defined for one interface as the time between two machine replacements. The method starts with a clustering stage during which Gaussian distributions are identified in the input data. To each identified mixture component corresponds a cluster. Then, for each of these mixture components, a criticality index is computed as well as a cluster confidence index. This index is called the Posterior Uncertainty Index (PUI) in the following sections. Then, a detection stage compares both indexes to thresholds to decide whether the input triggers an alarm or not. When an alarm is triggered, its root cause analyse starts.

To identify Gaussian models, the method starts by fitting a GMM that clusters the input data. A MAP approach is used to determine the best number of components in the mixture. There are no constraints on the variances and means of GMM. A particular initialisation strategy is chosen and argued in Section 4. The clustering stage takes as input a time series defined as one period of one test and returns as output the mixture components identified in the period. The clusters correspond to the different Gaussian distributions identified in the time series.

In order to obtain an index with highly generalizing properties related to the health state of the equipment, the CPk index has been chosen as the criticality index. One of the advantages of GMM is to build cluster membership functions for all points. These functions are used to propose an index of the quality of the mixture. The objective of this second index, the PUI index, is to control the posterior uncertainty of the chosen mixture components of all GMM points.

During the detection stage, the CPk and the PUI indexes are used. The CPk index is compared to a threshold fixed as a model parameter. It determines the criticality of the input data cluster. Indeed, the smaller the CPk, the higher the probability of encountering out-of-bounds tests. The PUI is also compared to a threshold to determine if the built mixture is acceptable. If mixture is not acceptable, all test values in the period are assigned to the same cluster and CPk is recomputed with it. The goal of this health monitoring stage is to detect within the acceptable mixtures those input clusters that have a high probability of containing out-of-bounds test values.

If the input clusters have out-of-bounds CPk index and PUI, then an isolation stage starts. These clusters are then called critical clusters. The idea is to compare the criticality index between ICT machines, interfaces and product positions in order to isolate the detected fault. In our case, the isolation consists in finding the elements responsible for the critical

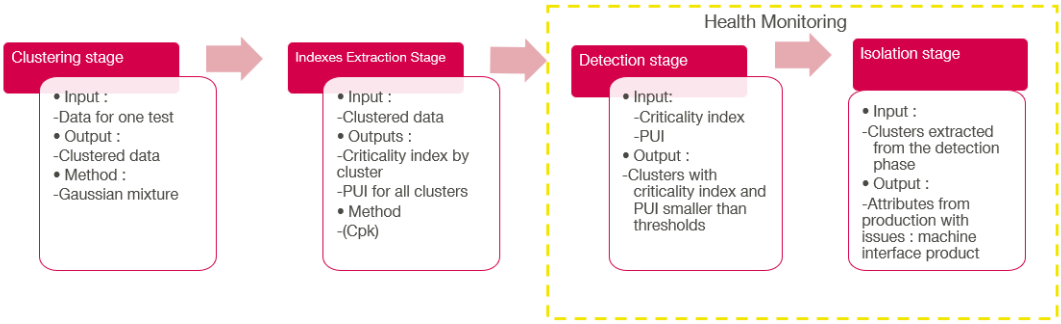

Figure 2. Overview of the proposed method

clusters. Possible candidates are the ICT machine, the interface and the product group.

2.4. Gaussian Mixture Model

Our methodology requires an unsupervised machine learning algorithm to automatically identify the distributions in a time series defined for one test and one period. For our data, the hypothesis of normality is considered, i.e. the distributions to be identified are Gaussian distributions.

Among the iterative clustering algorithms, k-means is one of the most popular. However, in our case, its application is difficult because the variance for each distribution can be different and k-means can be seen as a GMM with spherical Gaussian functions having a common width parameter σ (Bishop et al., 1995).

The Expected Maximisation (EM) algorithm for the GMM method seems to be more appropriate for the data. The EM algorithm iteratively improves an initial clustering solution in order to better fit the data to the chosen mixture (McLachlan & Thriyambakam, 1997). The solution is the locally optimal point with respect to a clustering criterion. The criterion used in the EM algorithm is the log-likelihood. It represents the probabilistic measure of how well the data of the EM algorithm fit the mixture. Another advantage of the EM algorithm compared to k-means concerns the membership function of the identified clusters. Indeed, k-means uses a membership function that assigns a point to a single cluster. Therefore, it is not possible to describe the uncertainty of clustering for each point. On the contrary, for GMM, the membership functions can take any value between 0 and 1 and can be used to build an uncertainty index (Bradley, Fayyad, & Reina, 1999). Let $\mu_k(x)$ denote the membership function of the cluster $k \in K$ for a point x with K the number of mixture components of the GMM. Posterior Uncertainty Index (PUI) is defined by

$$PUI = \sum_{i=1}^{n} \max_{k \in K}(\mu_k(x_i)) \tag{2}$$

with n the number of points of one period.

Nonetheless, like k-means, the EM algorithm gives a local optimum solution. This means that the solution depends on the initialisation of the algorithm. The quality of the solution relies on the chosen initialisation strategy. Initialisation strategies are discussed in Section 4.

3. Model selection strategies

In order to choose the best number of distributions to consider during the GMM stages, two solutions are investigated. The first solution is linked to the popular Bayesian Information Criterion (BIC) (Schwarz et al., 1978) and Akaike Information Criterion (AIC) (Akaike, 1998). To determine the best number of distributions, this solution considers the problem as a component density estimation problem. It searches for a mixture solution that has a Probability Density Function (pdf) as close as possible to the pdf of the data. If the hypothesis of normality of data is respected, the considered approximation is adequate according to the properties of the BIC explained in Section 3.1. The second solution considers a trade-off between the BIC and some properties of the mixture components and the distributions formed by the GMM. These properties are: the skewness and kurtosis of the mixture components and the overlap of the distributions. We propose a Mixed-criteria method to determine the best number of mixture components for the GMM.

3.1. Model order selection : BIC

Determining the correct number of mixture components in an unsupervised learning problem is a difficult task. For GMM the problem can be translated into finding the right number K of normal distributions in the mixture. A finite mixture distribution G is defined as:

$$G = \sum_{k=1}^{K} \eta_k f_k(y|\theta_k) \tag{3}$$

where η_k are the weights of the k^{th} GMM normal distribution, f_k is its density function parametrised by θ_k and $y = (y_1, ..., y_n)$ are the observations, i.e. the data in one period.

Selecting a wrong K may produce a poor density estimation. A common approach to select the best number of features is the MAP approach. This approach creates mixtures with different numbers of distributions. Depending on a criterion, the approach will select the best mixture model. The most popular criteria for model selection are BIC and AIC. These criteria are based on the log likelihood function of a mixture model M_K with K being the number of mixture components. Let n be the number of points of the dataset, the log likelihood is defined as:

$$l(\theta_K; K) = log(L(\theta_K; K)) \tag{4}$$

with

$$L(\theta_K; K) = \prod_{i=1}^{n} [\sum_{k=1}^{K} \eta_k f_k(y_i|\theta_k)]. \tag{5}$$

Both BIC and AIC use the penalty term v_K defined as $v_K = K(1 + r + r(r + 1)/2) - 1$ with r the number of features in the dataset. This penalty term is proportional to the number of free parameters in the mixture model M_k and penalises the complexity of the model. BIC is defined as follows:

$$BIC(K) = -2l(\theta_K; K) + v_K log(n) \tag{6}$$

with n the number of points.

AIC is defined as follows:

$$AIC(K) = -2l(\theta_K; K) + 2v_K \tag{7}$$

BIC approximates the marginal likelihood of a mixture model M_k and AIC is related to the Kullback-Leibler divergence between the mixture model and the real dataset. Both criteria have, under appropriate regular conditions, interesting properties. In particular, they both assume that the true distribution of the data lies within the created mixture models. Under this condition, BIC has been shown to be consistent (Yang, 2005). Indeed, if among the created mixture models, a model has the distribution of the data, it will be selected by BIC. For AIC, it has been shown that it is minimax optimal, i.e. it will select the model that minimises the maximum risk among all the built models (Yang, 2005). However, these regular conditions are often not met in practice. In particular for BIC, Laplace approximations are often invalidated. Nevertheless, in practice the consistent property of BIC seems to still be present if the objective of the mixture is to estimate density (Fraley & Raftery, 2002), (Roeder & Wasserman, 1997). On the contrary, if the objective is to estimate the real number of mixture components, AIC is known to overestimate this value (Celeux & Soromenho, 1996). Therefore, we choose to use BIC as the first method for our application case.

3.2. Model based clustering : Mixed-criteria method

As AIC and BIC have different, interesting properties, some articles try to combine them. (Barron, Yang, & Yu, 1994) proposed the creation of a criterion that combines both of them into a single one, while (Hansen & Yu, 1999) proposed to switch between these criteria using a statistical parameter. More recently, (Peng, Zhang, Kou, & Shi, 2012) proposed a multiple criteria decision-making based approach using validity indexes for crisp clustering. We propose a method that can mix BIC with other criteria, combining their strengths to obtain clusters better suited to our problem than those obtained with a single criterion. To improve the completeness of the method, a dataset with similar properties to the one of this study is created. The learning of the criteria aggregation is done with that dataset. This subsection describes our proposition for the chosen criteria and the aggregation function.

3.2.1. Chosen criteria

Several criteria are chosen as input parameters. In addition to the inference-based criteria BIC and AIC mentioned before, we also considered the Normalized Entropy Criteria (NEC) introduced by (Celeux & Soromenho, 1996) and defined as:

$$NEC(K) = \frac{E(K)}{l(\theta_K; K) - l(\theta_1; 1)} \tag{8}$$

where $E(K)$ is an entropy measure which involves the posterior probabilities of y_i belonging to the k^{th} mixture component. Entropy measure is computed by the following equation:

$$E(K) = -\sum_{k=1}^{K} \sum_{i=1}^{n} t_{ik} ln(t_{ik}) \tag{9}$$

with

$$t_{ik} = \frac{\eta_k f_k(y_i|\theta_k)}{\sum_{j=1}^{K} (\eta_j f_j(y_i|\theta_j)} \tag{10}$$

and $l(\theta_K; K)$, $l(\theta_1; 1)$ the log likelihoods as defined in Equation (4). By definition, $NEC(1) = 1$.

NEC is linked to the overlap between the normal distributions of the studied mixture. AIC, BIC and NEC can take very different values depending on the dataset. That is why it is mandatory to normalise them in order to use them for machine learning. A min-max normalisation is chosen, so that each new criterion will take a value between 0 and 1. The normalisation procedure is described as follows: Let $F_{e,k}$ be the value of the input feature for the experiment e and cluster number k. Let $NormF_{e,k}$ be its normalised value, it is defined as:

$$NormF_{e,k} = \begin{cases} \frac{-\min_{p \in [1:P]}(F_{e,p}) + F_{e,k}}{\theta(e)} & \text{if } \theta(e) \neq 0 \\ 0 & \text{else} \end{cases} \tag{11}$$

with

$$\theta(e) = -\max_{p\in[1:P]}(F_{e,p}) + \min_{p\in[1:P]}(F_{e,p}) \quad (12)$$

Other criteria linked to the normality of the clusters created by K mixture components are also considered:

$$Mk_j = \max_{k\in[1,K]}(|kurtosis_{k,j}|) \quad (13)$$

$$mk_j = \sum_{k=1}^{k=K}(|kurtosis_{k,j}|) \quad (14)$$

$$Ms_j = \max_{k\in[1:K]}(|skewness_{k,j}|) \quad (15)$$

$$ms_j = \sum_{k=1}^{k=K}(|skewness_{k,j}|) \quad (16)$$

$$Mm_j = \max_{k\in[1,K]}\left(|1 - \frac{median_{k,j}}{mean_{k,j}}|\right) \quad (17)$$

$$mm_j = \sum_{k=1}^{k=K}\left(|1 - \frac{median_{k,j}}{mean_{k,j}}|\right) \quad (18)$$

In these equations, the index k, j implies that the selected value is calculated on the k^{th} distribution of the mixture for the period j.

For clarity, the set of normalised BIC, AIC, NEC and normality-linked criteria is henceforth called $X_{j,K}$.

3.2.2. Aggregation function and performance assessment of the criteria

The aggregation function is based on the results of a clustering stage. This article uses the random forest classifier as it outperforms support vector machines and logistic regression for our application. The algorithm classifies the GMMs with the right number of mixture components into the class "recognized" and the others into the class "not recognized" using the chosen criteria $X_{j,K}$ as input. Nevertheless, this clustering has a major drawback: for one period, several GMMs can be classified in the "recognized class". Instead of using these classes directly, the aggregation function uses the posterior probability of the "recognized" class defined as $\alpha(X_{j,K})$. This allows to choose the best model among the models classified in the recognized class. The selected model is the one with the maximum posterior probability.

The pseudo-algorithm for the aggregation function is described in Algorithm 1. First, for each period in $[1 : J]$, GMMs are fitted with a number of distributions ranging from 1 to P. Next, the chosen criteria $X_{j,k}$ required by the random forest classifier are computed for each GMM. Then, the posterior probability $\alpha(X_{j,k})$ given by the random forest classifier is computed and the GMM with the maximum probability for the selected period j and number of distributions k in $[1 : P]$ is chosen as $GMM^*_{j,K}$.

Algorithm 1 Best Model Selection

Input:
Dataset with J periods
P, number of distributions to test
Random forest classifier already fitted
Empty list of GMM
Dataset with J periods

Output:
List of GMM^*

for j in $1, ..., J$ **do**
  $Maxproba = 0$
  for k in $1, ..., P$ **do**
    Fit GMM with period j and k number of distributions $GMM_{j,k}$
    Compute $X_{j,k}$
    Compute $\alpha(X_{j,k})$ using Random forest classifier
    if $\alpha(X_{j,k}) > Maxproba$ **then**
      $GMM^*_{j,K} = GMM_{j,k}$
      $Maxproba = \alpha(X_{j,k})$
  Insert $GMM^*_{j,K}$ in list of GMM^*

In order to assess the performance of the models selected as best for each criteria (GMM^*), an evaluation metric m is defined. This metric is computed for each dataset sample s and each criterion $X_{j,K}$ as follows:

$$m_{s,X_{j,K}} = \begin{cases} 1 & \text{if the criterion } X_{j,K} \text{ finds the right number of} \\ & \text{mixture components for sample } s \\ 0 & \text{else.} \end{cases}$$

3.3. Synthetic database

In order to compare the results of the model selection with the two approaches described in the sections 3.1 and 3.2, some 1D synthetic databases composed of several normal distributions are created. For the database generation, the procedure given by (Qiu & Joe, 2006a) is adapted. The general idea is to create several datasets based on an experimental design. In order to generate a balanced database, four factors are used:

1. The number of clusters for one example,
2. The minimum separation degree,
3. The relative cluster density,
4. The sample size.

For the first factor, values $[1, 2, 3, 4, 5]$ are used to match the tests to the exploratory analysis conducted on the industrial dataset.

Table 1. Factors of the design of experiments

Factor	Values
Number of clusters	$\{1, 2, 3, 4, 5\}$
Minimum separation degree	$\{10^{-5}, 0.01, 0.21, 0.34\}$
Relative cluster density	{all equal, one cluster 10% of data, one cluster 60% of data}
Sample size	{500, 1000, 2000}

The minimum separation degree is defined in (Qiu & Joe, 2006b) as an index extracted from an optimal dimension projection. In our case, the degree is simplified as follows:

$$Z(a) = \frac{L_k(a) - U_{k'}(a)}{L_{k'}(a) - U_k(a)} \tag{19}$$

with L_k, U_k, $L_{k'}$ and $U_{k'}$ the lower and upper $\frac{a}{2}$ percentiles of classes k and k'.

(Qiu & Joe, 2006b) present three separation values for the generation: $Z = 0.01$ indicates a close structure, $Z = 0.21$ indicates a separated structure and $Z = 0.34$ indicates a well-separated cluster structure. We choose to add the value $Z = 10^{-5}$ to create a very closer structure because in our data, we have closer structure than the ones created with $Z = 0.01$.

The relative cluster density factor is based on (Shireman, Steinley, & Brusco, 2017) and is defined as follows:

1. all clusters have the same number of points,
2. one cluster has 10% observations and the other clusters have the same number of observations,
3. one cluster has 60% of the observations and the other clusters have the same number of observations.

Finally, the considered sample size values are $[500, 1000, 2000]$. This corresponds to the period length in the application. Table 1 summarized the factors used in the proposed design of experiment.

3.4. Results

Model selection results are obtained by a cross-validation over the test data. The dataset is divided into four random parts indexed by t in the following. Then, each part of the dataset is chosen as the test dataset, the three other parts are used to learn the random classifier inside the Mixed-criteria method. The results presented in Table 2 correspond to the following evaluation metric D:

$$D = \frac{1}{4 \cdot S} \cdot \sum_{t=1}^{4} \sum_{s_t=1}^{S} m_{s_t,i} \tag{20}$$

with S the number of samples per parts. The closer D is to 1, the better the model selection.

The results in Table 2 show that BIC is very good at determining the right number of mixture components and works much better than AIC. The Mixed-criteria method improves the results of the BIC criterion and almost always finds the right number of mixture components.

Table 2. Performance comparison of criteria

Criterion	Generated Distributions					
	All	1	2	3	4	5
AIC	0,684	0,808	0,752	0,665	0,640	0,561
BIC	0,986	0,989	0.995	0,989	0.993	0,965
NEC	0,926	0,872	0,992	0,936	0,908	0,921
mk	0,524	0,975	0,890	0,363	0,226	0,175
Mk	0,664	0,955	0,884	0,534	0,471	0,478
Ms	0,423	0,994	0,661	0,273	0,135	0,056
ms	0,339	0,994	0,554	0,138	0,006	0,003
mm	0,264	0,978	0,179	0,064	0,074	0,021
Mm	0,227	0,895	0,115	0,050	0,049	0,019
Mixed-criteria	**0,993**	**0,997**	**0,997**	**1,000**	**0,995**	**0,997**

In our case, both BIC and Mixed-criteria method can be used. The main differences in the clusters obtained with both approaches are found for the overlap cases. The Mixed-criteria method is less likely to accept mixtures with overlapping between distributions, while this does not directly affect BIC.

4. INITIALISATION STRATEGIES

Initialisation strategies are very important to limit cost time of an algorithm. As the method has to be run at least 1000 times per product, the cost time is an important parameter. For our application, the choice is made to test three different categories of initialisation strategies.

4.1. Random initialisation

Initialisation strategies based on stochastic methods are quite popular. Among them, two methods are tested. A first basic strategy consists in initialising the EM input parameters randomly for a fixed number of Gaussian distributions. The EM parameters are in that case the variance, the mean, and the weight of each mixture component.

(Biernacki, Celeux, & Govaert, 2003) presents the second strategy called "emEM" (for expected maximisation Expected Maximisation). This method starts with a stage called short EM. In this stage, an EM algorithm is run for several iterations from random starting points. The number of iterations of this stage is set as a parameter. Finally, among the solutions given by the short EM, the strategy with the highest likelihood is chosen. Then the EM algorithm is run, starting with the parameters of the chosen solution. This allows more initialisation solutions to be explored or less time to be spent exploring the same number of initialisation solutions. This approach, which is considered the most efficient by (Biernacki et al., 2003), has some drawbacks. The maximum number of iterations in the short EM stage has to be defined by the user and can have a huge impact on the final result of the algorithm. Moreover, this parameter can change depending on the considered test and period. Another drawback, shared by both

approaches, is that they can lead to locally optimal solutions when too few initialisations are performed. This drawback is aggravated for the emEM because the first stage restricts the search space by construction.

4.2. K-means initialisation

Some methods use the results of a k-means clustering algorithm as input of the EM algorithm. In (Steinley & Brusco, 2011), a theoretical property between the parameters extracted from a solution given by a k-means algorithm and a mixture model is demonstrated. The main drawback of this approach is that, as for random methods, it could lead to locally optimal solutions. Additionally, the number of initialisations of the k-means procedure is also a parameter to settle. Another drawback relates to the solutions generated by this method. When clusters overlap, k-means has difficulties in representing correctly the data. Moreover, when the data have a large difference in variance within their Gaussian distribution, k-means is not the most suitable algorithm (Shireman et al., 2017).

A classical procedure with a fixed number of initialisations of the k-means method has been tested. It gives a set of solutions which are then used as input for the EM algorithm. In addition, an approach with "short EM", similar to the one with the random initialisation method, has also been tested.

4.3. Otsu initialisation

One of our contributions is a new initialisation strategy adapted to small dimensions. It is based on a clustering procedure, named the Otsu method. This method, described in (Otsu, 1979), is widely used for image segmentation. The initial objective of the method is to select thresholds between levels in a greyscale image. It is based on the 1D histogram of grey levels. For a foreground/background problem, the selected threshold is defined as the one that minimises the intra-cluster variance. This problem can be translated as finding different distributions in a greyscale histogram, which has similarities with our clustering problem. For multiple clusters, an implementation of this algorithm is described in (Liao, Chen, Chung, et al., 2001). For our application, the outputs of the multi-cluster Otsu method are used as input by the EM algorithm. One of the main drawbacks of this approach is the restriction of the search space to only one possible result.

4.4. Evaluation of the initialisation strategies

The following procedure is used in order to evaluate the quality of performance of different initialisation methods. The input is a dataset divided into samples. One sample represents a period of production of one test. The performance of each initialisation method is evaluated according to three criteria: the ability to find the best global solution (BIC in our case), the ability to find the closest result to the optimal solution and the computation time of the evaluated strategy.

4.4.1. Best global solution

The best global solution is given by BIC. The lowest BIC is the best local solution that can be given by the EM algorithm in terms of fitting the Gaussian mixture density. Therefore, this criterion is chosen as an indicator of the fitting algorithm. For each input, the l strategy with the lowest BIC receives a score $\zeta_{j,l}$ equal to 1 and the other strategies receive a score of 0. Then, for each initialisation strategy l, a global index called GFI_l (Global Fitting Index) is built over all inputs and is defined as follows:

$$GFI_l = \frac{1}{J}\sum_{j=1}^{J}\zeta_{j,l} \tag{21}$$

with

$$\zeta_{j,l} = \begin{cases} 1 & \text{if } BIC_{j,l} = \max_{i\in[1,L]}(BIC_{j,i}) \\ 0 & \text{else.} \end{cases} \tag{22}$$

where J is the number of inputs, l is the evaluated strategy, L is the total number of strategies evaluated and j indexes over all inputs.

4.4.2. Distance to the optimal solution

One of the main drawbacks of the previous metric is that if an initialisation strategy is always the second best, it will have a GFI of 0, even if it could be an acceptable solution. In order to complete the evaluation, another metric Δ is proposed and defined as follows:

$$\Delta_l = \sum_{j=1}^{J} score_{j,l} \tag{23}$$

with

$$score_{j,l} = \begin{cases} Dist_{opt}(j,l) & \text{if } \beta_j \neq 0 \\ 0 & \text{else.} \end{cases} \tag{24}$$

$$Dist_{opt}(j,l) = \frac{\min_{t\in L}(BIC_{j,t}) - BIC_{j,l}}{\beta_j} \tag{25}$$

and

$$\beta_j = (\max_{t\in L}(BIC_{j,t}) - \min_{t\in L}(BIC_{j,t})) \tag{26}$$

with l a strategy of initialisation and j one period.

4.4.3. Time and Parameters

In order to evaluate the efficiency of each initialisation strategy, the elapsed time required by the EM algorithm and the initialisation is measured. The time was measured from the beginning of the initialisation for the first test to the end of the clustering for the last test of the database. Then, the mean elapsed time per period for each set of parameters and each

initialisation method is computed.

In order to assess the performance of an initialisation method, a design of experiments is proposed. Four factors are used:

1. The initialisation method: k-means, Random or Otsu;
2. The convergence threshold;
3. The number of iterations of the short EM part of the algorithm;
4. The number of initialisations.

For the Otsu initialisation method, the only possible factor is the convergence threshold. When the convergence threshold is less than the lower bound gain on the likelihood, the EM iterations stop. The convergence threshold tested values are $[10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}]$. The tested number of iterations of the short EM part are $[10, 20, 50, 100, 200, inf]$, when the number is infinite, then there is no short EM stage. The tested values of the number of initialisation are $[10, 20, 40, 50, 60]$. The presented results are obtained with 100 tests and 14 periods for each test.

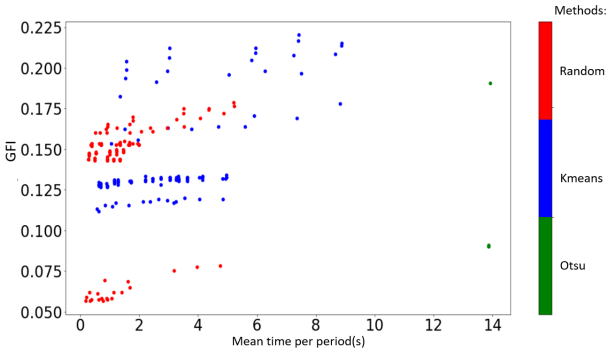

Figure 3. Best global solution vs mean time per period

By analysing the results presented in Figure 3, it can be observed that the k-means method has globally the best solutions in terms of GFI. In the meanwhile, the solutions provided by the random initialisation method are faster than those obtained with k-means but provide less fitted solutions. The Otsu initialisation method has a GFI score close to that of k-means only for the smallest convergence threshold. Furthermore, the initialisation time of the Otsu method seems to be too large to be competitive against the k-means method. For these reasons, the k-means initialisation method is chosen in the following part and the different scores are then recomputed for the solutions given by the k-means initialisation method.

Figure 5 and Figure 4 respectively give the GFI and the Δ scores as a function of the mean time per period. One set of parameters seems to be the best compromise between computation time and fitting: the k-means initialisation method, a convergence threshold of 10^{-5}, no short EM stage and 10 initialisations.

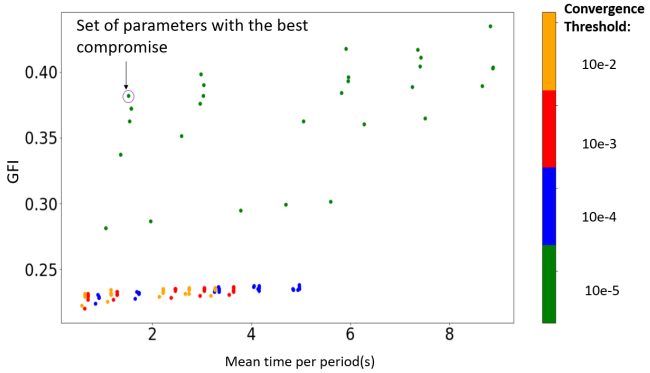

Figure 4. Best global solution vs mean time per period for k-means initialisation method solutions

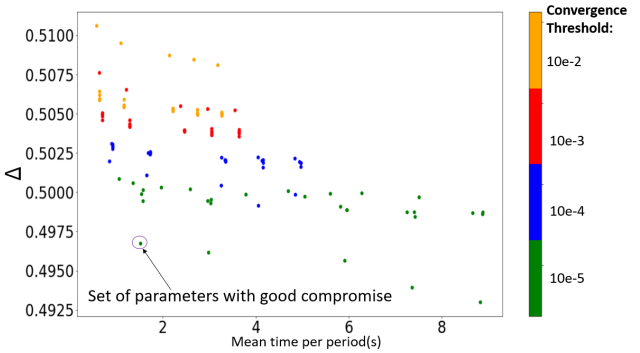

Figure 5. Distance to the optimal solution vs mean time per period for k-means initialisation method solutions

5. Application of the proposed method for one test version

The method described in Section 2.3 was implemented on a dataset formed by the test values for one test version and for two interfaces. The method was implemented with Python 3. The EM algorithm and Random Forest are from the scikit-learn library. The studied version has 1012 tests per product. In order to choose the thresholds of the CPk index and of the PUI, a directory of maintenance operations is used. Three tests lead to maintenance operations. The value of threshold 1 for the CPk index and 0.8 for PUI allow the detection of these three tests found in the maintenance operation directory. Moreover, the method also detects ten other tests whose distributions are in contradiction with the specification limits previously defined as UL and LL in Equation 1. This section illustrates the health monitoring methodology on one test with product-related faults. The shown tested component is a resistor with a nominal value of 1000Ω, a lower limit of 970Ω and an upper limit of 1030Ω.

The time series is presented in Figure 6 where different colours correspond to different periods. Period number 3 is identified by the orange rectangle, and seems to be abnormal as the tested components group has a mean closer to the lower limit

than the mean of the groups in the other periods. Moreover, this period is particularly interesting as it seems to have at least two groups with different means.

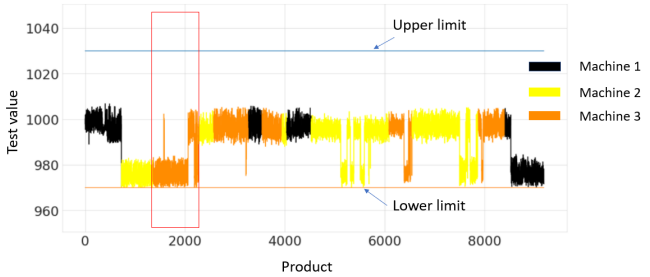

Figure 6. Time series of resistance example for the year 2020 and one test version

The method starts with the identification of clusters with a GMM algorithm. Both BIC and Mixed-criteria method find 4 clusters with the initialisation parameters chosen as concluded in Section 4.4: k-means initialisation method, a convergence threshold of 10^{-5}, no short EM stage and 10 initialisations. The Mixed-criteria method uses an aggregation function with a random forest classifier trained on the synthetic database created in Section 3.3

The results of the clustering stage are presented in Figure 7: the best number of clusters is 4, according to the BIC method. The figure shows the number of samples for each test value for the period number 3.

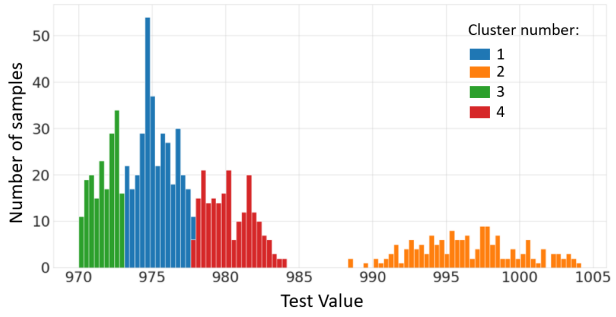

Figure 7. Histogram of the classes of the period number 3

Four clusters are identified by the clustering algorithm: clusters 1, 3 and 4 correspond to the mixture components with a mean close to the lower limit while cluster 2 corresponds to a distribution with a mean close to the nominal value.

Figure 8 illustrates the actual density and the density estimation for GMM if only 2 clusters are selected: the fitting error is clearly important. Figure 9 illustrates the actual density and the density estimation for GMM if 4 clusters are selected. The GMM with 4 clusters exhibits a lower fitting error and higher cluster overlapping than the GMM with 2 clusters.

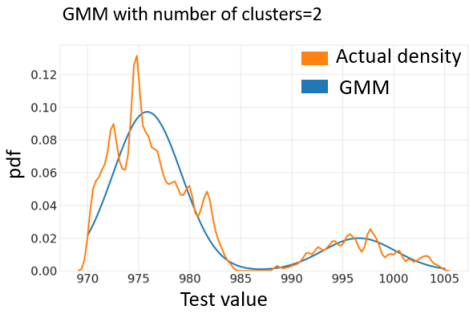

Figure 8. Kernel density estimation for GMM with 2 mixture components

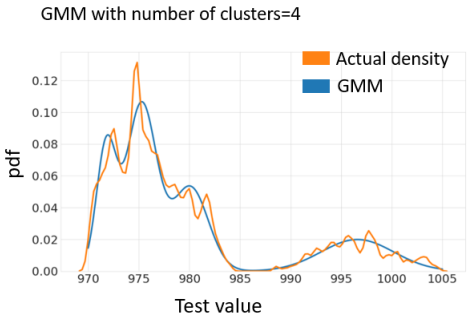

Figure 9. Kernel density estimation for GMM with 4 mixture components

During the detection stage, the CPk and the PUI indexes are computed and compared to the chosen thresholds. The criticality index computed for each cluster is presented in Table 3. Cluster 3 has a criticality index smaller than the threshold, so this cluster triggers an alarm. The PUI value is 1, that is higher than the chosen threshold. The isolation stage must then be launched for cluster 3.

As an alarm is triggered by the detection stage for cluster 3, the isolation stage of the method is then started. Table 4 presents the criticality indexes of the other product positions and interfaces for the same test. The studied interface is interface number 1160 and the position on the panel is 1. The studied product has two positions and two interfaces (1160 and 1161).

The results of Table 4 imply an anomaly related to the product itself. Indeed, for this period, all the positions and interfaces have one mixture component (clusters in bold in Table 4) with a criticality index lower than the acceptable threshold. It is

Table 3. CPk per cluster for period 3

cluster	Number of points	CPk
1	168	1.44
2	165	2.46
3	212	**0.68**
4	373	2.20

Table 4. CPk for period 3 for the two positions and the two interfaces

Cluster number	Position	Interface	CPk
1	1	1161	4.81
2	1	1161	**0,63**
3	1	1161	4.25
1	2	1161	4.19
2	2	1161	**0.69**
3	2	1161	4.76
4	2	1161	2.29
1	1	1160	1.44
2	1	1160	2.46
3	1	1160	**0.67**
4	1	1160	2.20
1	2	1160	1.58
2	2	1160	2.42
3	2	1160	**0.72**

impossible for two different machines and two interfaces to have the same anomaly at the same period. Considering this, it can be concluded that the found anomalies ($CPk < 1$) come from the products and the equipment.

6. Conclusion and future work

This article proposes a data-based health monitoring method for both fault detection and isolation that uses the CPk as a health indicator. The results show that the CPk is linked to the health of the equipment or product group. The study of initialisation strategies allows to choose a strategy and a set of parameters adapted to our application. The proposed method allows to detect more tests close to the acceptable limits without triggering false alarms. It also provides fault isolation by comparing the criticality index from different product positions, machines and interfaces. It can also provide decision support to ICT machine operators and maintenance personnel.

The CPk is directly linked to the probability of having one value out-of-bounds if the hypothesis of data normality is respected. One improvement could be to check the normality of the built classes. Then, if the data normality hypothesis is not respected, another method should be used to compute the probability of having one out-of-bound test value. A good solution could be the Extreme Value Theory (De Haan & Ferreira, 2007). Future work will also study a prognosis stage based on the history of the CPk values and the computed clusters.

References

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike* (pp. 199–213). Springer.

Baraldi, P., Di Maio, F., Rigamonti, M., Zio, E., & Seraoui, R. (2015). Clustering for unsupervised fault diagnosis in nuclear turbine shut-down transients. *Mechanical Systems and Signal Processing*, *58*, 160–178.

Barron, A., Yang, Y., & Yu, B. (1994). Asymptotically optimal function estimation by minimum complexity criteria. In *Proceedings of 1994 ieee international symposium on information theory* (p. 38).

Biernacki, C., Celeux, G., & Govaert, G. (2003). Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics & Data Analysis*, *41*(3-4), 561–575.

Bishop, C. M., et al. (1995). *Neural networks for pattern recognition*. Oxford university press.

Boyles, R. A. (1991). The taguchi capability index. *Journal of quality technology*, *23*(1), 17–26.

Bradley, P. S., Fayyad, U., & Reina, C. (1999). Efficient probabilistic data clustering: Scaling to large databases.”..

Celeux, G., & Soromenho, G. (1996). An entropy criterion for assessing the number of clusters in a mixture model. *Journal of classification*, *13*(2), 195–212.

De Haan, L., & Ferreira, A. (2007). *Extreme value theory: an introduction*. Springer Science & Business Media.

Fraley, C., & Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, *97*(458), 611–631.

Hansen, M., & Yu, B. (1999). Bridging aic and bic: an mdl model selection criterion. In *Proceedings of ieee information theory workshop on detection, estimation, classification and imaging* (Vol. 63).

Hsiang, T. C. (1985). A tutorial on quality control and assurance-the taguchi methods. In *Asa annual meeting la, 1985.*

Liao, P.-S., Chen, T.-S., Chung, P.-C., et al. (2001). A fast algorithm for multilevel thresholding. *J. Inf. Sci. Eng.*, *17*(5), 713–727.

McLachlan, G., & Thriyambakam, K. (1997). The em algorithm and extensions new york wiley.

Mehrjou, A., Hosseini, R., & Araabi, B. N. (2016). Improved bayesian information criterion for mixture model selection. *Pattern Recognition Letters*, *69*, 22–27.

Montesinos-López, et al. (2020). Maximum a posteriori threshold genomic prediction model for ordinal traits. *G3: Genes, Genomes, Genetics*, *10*(11), 4083–4102.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, *9*(1), 62–66.

Peng, Y., Zhang, Y., Kou, G., & Shi, Y. (2012). A multicriteria decision making approach for estimating the number of clusters in a data set. *PLoS one*, *7*(7), e41713.

Qiu, W., & Joe, H. (2006a). Generation of random clusters with specified degree of separation. *Journal of Classification*, *23*(2), 315–334.

Qiu, W., & Joe, H. (2006b). Separation index and partial membership for clustering. *Computational statistics & data analysis*, *50*(3), 585–603.

Ren, H., Xu, B., Wang, Y., Yi, C., Huang, C., Kou, X., ... Zhang, Q. (2019). Time-series anomaly detection service at microsoft. In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining* (pp. 3009–3017).

Roeder, K., & Wasserman, L. (1997). Practical bayesian density estimation using mixtures of normals. *Journal of the American Statistical Association*, *92*(439), 894–902.

Schwarz, G., et al. (1978). Estimating the dimension of a model. *Annals of statistics*, *6*(2), 461–464.

Shireman, E., Steinley, D., & Brusco, M. J. (2017). Examining the effect of initialization strategies on the performance of gaussian mixture modeling. *Behavior research methods*, *49*(1), 282–293.

Steinley, D., & Brusco, M. J. (2011). Evaluating mixture modeling for clustering: recommendations and cautions. *Psychological Methods*, *16*(1), 63.

Wu, C.-W., Pearn, W., & Kotz, S. (2009). An overview of theory and practice on process capability indices for quality assurance. *International journal of production economics*, *117*(2), 338–359.

Yang, Y. (2005). Can the strengths of aic and bic be shared? a conflict between model indentification and regression estimation. *Biometrika*, *92*(4), 937–950.