Impact of standardization applied to the diagnosis of LT-PEMFC by Fuzzy C-Means clustering

Damien Chanal
FEMTO-ST Institute, FCLAB
Univ. Bourgogne Franche-Comté, CNRS
Belfort, France
damien.chanal@femto-st.fr

Nadia Yousfi Steiner
FEMTO-ST Institute, FCLAB
Univ. Bourgogne Franche-Comté, CNRS
Belfort, France
nadia.steiner@univ-fcomte.fr

Didier Chamagne
FEMTO-ST Institute, FCLAB
Univ. Bourgogne Franche-Comté, CNRS
Belfort, France
didier.chamagne@univ-fcomte.fr

Marie-Cécile Pera
FEMTO-ST Institute, FCLAB
Univ. Bourgogne Franche-Comté, CNRS
Belfort, France
marie-cecile.pera@univ-fcomte.fr

***Abstract*— In the domain of fuel cell systems, Machine Learning diagnostic tools use signal in operation such as temperature, voltage and current or specific experiments such as Electrochemical Impedance Spectroscopy or Current Interruption. One of the most important tasks in Machine Learning is to generate high-quality features from a database. Just as the choice of features to be extracted is important, it is crucial to correctly standardize the data in order to eliminate distortions of the State of Health space that represents all the possible states of the system. Standardization permits to reduce the computation time and to improve the performance of diagnostic algorithms. In this work, a comparison of the main standardization methods is proposed for a diagnostic approach and two databases are used as study cases. A total of seven standardization approaches are compared: (i) Normalizer, (ii) Min-Max scaler, (iii) Max Absolute scaler (iv) Standard scaler, (v) Yeo-Johnson Power transformer, (vi) Uniform Quantile transformer and (vii) Normal Quantile transformer. Uniform quantile transformer provides very good performances for both datasets, making this method very attractive for potential generic use.**

Keywords—Diagnostic, Machine Learning, Standardization methods, Database quality, Fuel cells, Electrochemical Impedance Spectroscopy.

I. INTRODUCTION

Fuel cells are good candidates as power generation for the future clean energy. They convert hydrogen and oxygen directly into electricity, heat and water with an electrical efficiency of about 50%. They are very relevant in several areas such as transportation and stationary. One of the most advanced fuel cell technologies is the low temperature proton exchange membrane (LT PEMFC) which is capable of starting at temperatures around 0 degree and operating between 60 and 80 degrees. Currently, one of the obstacles to the development of fuel cells is their limited lifetime. According to the Department of Energy (DoE), one of the objectives for 2020 was to increase lifetime of fuel cells for stationary and transportation up to 40 000 and 5 000 hours respectively under realistic operating conditions [1]. In order to achieve and improve these lifetime goals, monitoring and diagnostic tools suitable for fuel cell systems should be used to detect early and allow correction of any abnormal condition that may occur. Indeed, a good diagnostic allows a quick detection of faults which permits to ensure a correct recovery of the performance, to limit irreversible degradation induced by the fault and thus an improvement of reliability and the lifespan. There are different diagnostic methods that can be classified as model based and non-model based approaches. In both cases, artificial intelligence which establishes a relationship between one or more inputs and an output without knowledge of physics have been used successfully [2]–[6]. A database is needed in order to train with known data (off-line part) before being able to analyze unknown data (on-line part) and return the State of Health (SoH) of the system. In order to optimize the diagnostic algorithms using databases, it is generally recommended to standardize the data. This permits to reduce the computational time and improve the results in case optimization algorithms are used. This step is crucial to ensure the robustness of the diagnostic approach against the acquired raw data (noise, outliers, uncertain states…) and decreases the need for the user's expertise to preprocess the data.

This paper is based on the diagnostic approach and databases developed during Health Code project [7] which is followed by RUBY project [8]. During the Health Code project, a diagnostic tool based on the use of Electrochemical Impedance Spectroscopy (EIS) and Fuzzy C-means clustering is developed to detect faulty conditions. Stacks of two different technologies have been used, an H_2/O_2 feeding one and an H_2/Air technology one, were studied in two different laboratories under faulty conditions. For both stacks, faulty conditions were flooding, drying, fuel & reactants starvation. The H_2/Air stack was also tested under fuel poisoning conditions.

In the first section, a presentation of the algorithm developed in Health Code is made, then the second section presents an overview of three main families of standardization methods named Normalization, Linear scaling and Non-linear transformation. Finally, a presentation of the improvements made to the diagnostic approach and a comparison of the diagnostic results according to the chosen normalization methods is made in section 3.

II. DIAGNOSTIC APPROACH

The method used during the Health Code project is based on the use of a Fuzzy C-means classifier to detect the SoH of data recorded during EIS measurements performed online through a relevant control of the fuel cell output converter. A global presentation of the diagnostic approach is presented in Fig. 1 and

This project has received funding from the Fuel Cells and Hydrogen 2 Joint Undertaking (JU) under grant agreement No 875047

detailed in this section, however, more information about this method and data are available in [6].

The offline processing is composed of the following steps. First, features from the EIS are extracted (feature generation). The most interesting information (i.e features) are chosen. Then features are standardized which is the step this paper is focused on. Then, a selection of the ones containing the best information to discriminate the SoH of the fuel cell is done. Finally, data are classified using Fuzzy C-means clustering.

A. Off-line section

In the developed algorithms, the extracted features are: the minimum and maximum magnitudes of impedance respectively named (*mm*) and (*Mm*); the difference between maximum and minimum magnitude (*ΔMag*); the polarization resistance (*R_pola*); the minimum and maximum phase respectively (*mp*) and (*Mp*) ; the phase at a frequency of 0.1 Hz (*P1*); the difference between P1 and Mp (*ΔPha)*. Also, an analysis of phase during a linear part of Bode diagram is done ([0.1 -1] Hz). Equation (1) describes phase as a first order equation of frequency (f):

$$Phase = A \times f + B \tag{1}$$

Coefficients A and B are extracted as features.

The standardization method used is based on quantile information to make data follow a uniform distribution. This method was selected because of its ability to handle outliers and noisy data. The feature selection approach uses the Pearson Correlation Coefficient (PCC) to filter high correlated data and then it uses an ANOVA F-Test to sort features. Once the generation of features done, the diagnostic algorithm consists of using a Fuzzy C-means clustering to create clusters which will be used to detect the State of Health (SoH) of training data. During this step, the experience of the user is required: to optimize the creation of clusters, for each fault, the user will give only data associated to the faults and enter the number of clusters wanted (in the presented diagnostic it was the number of faults' level tested). It permits to optimize the localization of clusters for each fault even if it modifies the non-supervised character of Fuzzy C-means. Concerning fuel poisoning, a specific data clustering is made to identify the CO poisoning in a first place. As a matter of fact, it is easy to detect as it exhibits positive values of the imaginary part of the impedance.

B. On-line section

To classify new data, the algorithm extracts and standardizes the best features determined in the off-line section. To associate this fault with a known SoH, it computes the Euclidean distances between the new data and all clusters. The associated SoH corresponds to the closest cluster.

III. Standardization Methodologies

The key point in the development of machine learning algorithms is the generation of good quality features. Indeed, a good feature generation allows decreasing the predominance of possible outliers and noises, reducing the computation time but also improving the accuracy and the robustness of the results. In the case of classification algorithms that rely on distance calculations, the choice of the used standardization method is crucial. It consists in adjusting data value when they are not in

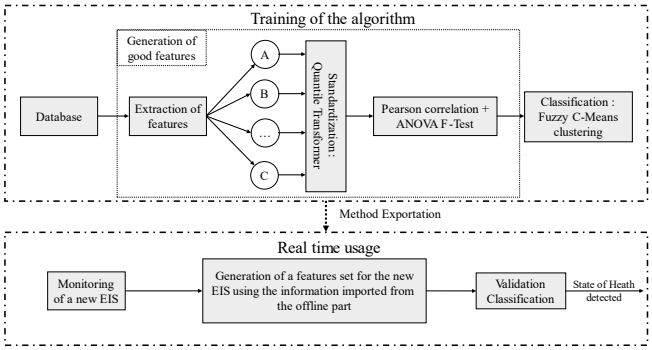

Fig. 1. Global principle of diagnosis tool developed in [8]

the same range to eliminate distortions of the SoH space and make them comparable. Magnitude of features affects algorithms' performances, especially when some features have much larger values than others. There are three main families of methods to standardize data: Normalization, Linear scaling and Nonlinear transformation. A short presentation of main standardization of each family is presented below. Each algorithm presented is implemented in scikit-learn [9].

A. Normalization

In general, the data are standardized by features, however it is possible to standardize each sample so that its norm equals 1. This method of standardization is named normalization. It is interesting to normalize samples when the objective is to quantify the similarity of any pair of samples.

Mathematically a norm is a total size or length of all vectors in a vector space or matrices. The norm of a vector *x* can be calculated at several level *(p)* by using equation below:

$$\|x\|_p = \sqrt[p]{\sum_i |x_i|^p} \tag{2}$$

Where $p \in R$ is the level of the norm and *x* the vector to be normalized. In machine learning, the normalization uses generally 3 levels of norm which are:

The L1 norm, also named "Manhattan norm" which corresponds to the first level of norm (*p*=1) and is the sum of absolute values of vector *x*. Equation (2) can be simplified as shown in (3):

$$\|x\|_1 = \sum_i |x_i| \tag{3}$$

The L2 norm, also named "Euclidean norm" which corresponds to the second level of norm (*p*=2) and is the sum of absolute values of vector x. Equation (2) can be simplified as shown in (4):

$$\|x\|_p = \sqrt{\sum_i x_i^2} \tag{4}$$

The infinite norm also called "Infinite norm" corresponds to the level when $p \to \infty$. In this configuration, the calculation of the norm can be simplified as (5):

$$\|x\|_p = \sqrt[\infty]{\sum_i |x_i|^\infty} \tag{5}$$

Because of the property of infinite, considering *j* as the highest entry in the vector *x*, it is possible to write (6) and (7):

$$x_j^{\infty} \gg x_i^{\infty} \quad \forall\, j > i \tag{6}$$

$$\|x\|_{\infty} = \sqrt[\infty]{\sum_i |x_i|^{\infty}} \cong \sqrt[\infty]{\left|x_j\right|^{\infty}} = \left|x_j\right| = max_i\,(|x_i|) \tag{7}$$

Once the norm is calculated, it is enough to divide each member of the vector x to obtain a unit vector. Formula is presented in (8):

$$x_{normalized} = \frac{x}{\|x\|_p} \tag{8}$$

Normalization is a powerful process, which can be used for tasks where it is possible to observe variability between the different case such as clustering and text classification. However, in the case of noisy data, they are sensitive to outliers which can impact the norm calculation.

B. Linear scaling

Linear standardization methods are the most widely used methods to scale features. They are quite simple to implement and work well for most databases. In addition, linear scalers are very useful to accelerate algorithms which use descent gradient. Indeed, in the case where one characteristic is higher than the other, it is more difficult to converge to the optimal value of the function. There are different linear scaling methods which use several parameters to standardize.

The first scaling method consists in scaling data in the range [0-1], it is also called “Min-Max feature scaling”. It consists of using minimal and maximal data as boundaries and rescaling data. Mathematically, (9) permits to scale a vector *x* in the range [0, 1]:

$$x_{\,scaled} = \frac{x - min(x)}{max(x) - min(x)} \tag{9}$$

One of the advantages of the Min-Max scaler is that it allows putting in the same interval features that can be very different while keeping all information since the distance ratios are kept. In the case of algorithms based on the distance between points, it allows keeping features with small values compared to those with large values.

The second method of scaling data is called "Max Absolute Scaling". It uses the maximum absolute value of a vector x to scale the features in the range [0, 1] or [-1, 1] depending on whether they are negative values. This method consists in dividing the vector *x* by its maximal absolute value as shown in (10):

$$x_{scaled} = \frac{x}{max(abs(x))} \tag{10}$$

Max Absolute scaler is very similar to Min-Max scaler, nevertheless it should be used for data that are already centered on zero.

The third method of linear scaling is called "Standard scaler". The objective of this method is to transform the features so that they have a mean of zero and a standard deviation of one as shown in (11):

$$x_{scaled} = \frac{x - \mu_x}{\sigma_x} \tag{11}$$

With μ the mean and σ is the standard deviation.

Standard scaler allows for data centering and make them easy to use with statistical machine learning algorithms such as Principal Components Analysis (PCA). The main disadvantage of the three linear scalers presented above is that they are very sensitive to outliers in the dataset.

This is why standardization algorithms using statistics were developed. It is the case of robust scaler which uses median and interquartile range (IQR) of data to reduce the importance of outliers. Formula to standardize data is:

$$x_{scaled} = \frac{x - median}{IQR} \tag{12}$$

Equation 12 looks similar to (11), however median and IQR are more robust to outliers than mean and standard deviation because they use the position of the data rather than the values.

C. Non-linear transformation

Even if the "robust scaler" permits to reduce the importance of extreme values, it is preferable sometimes to use non-linear transformations. These non-linear transformations allow transforming the data so that they change their distribution. There are two types of standardization that allow doing this: power transformations and quantile transformations.

Power transformations are parametric and monotonic transformations. They are useful to stabilize variance of features which are heteroscedasticity and map data to make them more gaussian-like. It exists 2 main power transformations: Box-Cox and Yeo-Johnson transformations. Box-Cox transformer [10] is defined by (13):

$$x_i^{(\lambda)} = \begin{cases} \frac{x_i^{\lambda} - 1}{\lambda} & if \lambda \neq 0 \\ ln(x_i) & if\ \lambda = 0 \end{cases} \tag{13}$$

With *x* vector to transform, and λ the power parameter of transformation which is determined through maximum likelihood estimation.

Box-Cox transformer allows transforming a dataset into a Gaussian like distribution. However, it is limited in that it only allows strictly positive values. Because data from EIS are positive and negative, it is not possible to use this transformer. This is not the case of Yeo-Johnson transformer [11] which has no restrictions. It is defined in (14):

$$x_i^{(\lambda)} = \begin{cases} \frac{[(x_i+1)^{\lambda} - 1]}{\lambda} & if \lambda \neq 0,\ x_i \geq 0 \\ ln(x_i + 1) & if \lambda = 0,\ x_i \geq 0 \\ \frac{-[(-x_i+1)^{2-\lambda} - 1]}{(2-\lambda)} & if \lambda \neq 2,\ x_i < 0 \\ -\,ln(-x_i + 1) & if \lambda = 2,\ x_i < 0 \end{cases} \tag{14}$$

The Box-Cox and Yeo-Johnson methods have the same objectives; however, they are slightly different. Indeed, in the case where the values are strictly positive, the Yeo-Johnson transformation is identical to the Box-Cox power transformation of (x+1). However, these two methods are regularly used in many domains such as machine learning. In [12] properties of Box-Cox transformation for pattern classification are presented. In [13] the effect of standardization is study on speech emotion

recognition, Yeo-Johnson transformer is compared to linear scaling and normalizer.

In addition to power transformer which makes data Gaussian-like, it is possible to use quantile transformer which uses information contained in quantile to make data follow a uniform or normal distribution. Quantile transformer formula is presented in (15):

$$G^{-1}\big(F(x)\big) \tag{15}$$

With F the cumulative distribution function of x and G^{-1} the quantile function of output distribution G.

Quantile transformers are very useful to reduce the importance of outliers. The negative point of this function is that it distorts correlations and distances within and across features because it smooths the original distribution. Nevertheless, the characteristics measured at different scales are more easily comparable. In addition, it is worth noting that when a new data is transformed with quantile transformer, it is not possible to extrapolate it unlike others standardization methods. Indeed, if the new data are larger or smaller than those used to determine the transformation boundaries, the standardized value is limited to the minimum or maximum fitted value. For example, in the case of a uniform distribution, the possible range is [0, 1], so if a new outlier appears, the standardized value will be 0 or 1.

IV. IMPACT OF STANDARDIZATION

In order to define if an algorithm is powerful or not, it is necessary to define metrics able to measure the correct classification of data. In addition to correct metrics, it is better to evaluate the classification of data with different training and testing sets to have a more general view of performances. A good method to measure this generality is to use a cross-validation process. It is a statistical method which consists in dividing the database into several parts (k parts) to train it with k-1 parts and test it on the last part. It exists several ways to divide the dataset in k parts but the retained one is the "Leave One Out" which consist of training dataset with all data except one and proceed by iteration to be able to test all data.

A. Evaluation of algorithms

One of the most useful ways to measure the effectiveness of a machine learning algorithm is to define multiple metrics instead of just one. The interest of using several indices is to observe the most common types of errors in order to have a better understanding of the algorithm and perhaps to add extra steps when detecting certain conditions in order to limit the risk of errors. In this study, widely used indices are computed to evaluate the performances and analyze the type of mistakes if any.

The first index is the confusion matrix which permits to observe the 4 cases of classification for a specific condition "f" as shown in Tab. I:

- "*Tp*" the number of samples correctly assigned to "f"
- "*Fn*" the number of samples wrongly assigned to "f"
- "*Fp*" the number of samples wrongly not assigned as "f"
- "*Tn*" the number of samples correctly not assigned as "f"

TABLE I. REPRESENTATION OF CONFUSION MATRIX

	Actual condition	
Detected condition	***True***	***False***
True	Tp	Fp
False	Fn	Tn

The second index is the accuracy score which permits to represent the number of correct classifications under all samples. Equation (16) shows the formula to determine accuracy score:

$$Accuracy = \frac{Tp + Tn}{Tp + Tn + Fp + Fn} \tag{16}$$

The third index is the precision score which is useful to observe the ratio of correct positive classification to all positive detected classifications. The formula of precision score is presented in (17):

$$Precision = \frac{Tp}{Tp + Fp} \tag{17}$$

Fourthly, the recall score, also called sensitivity, is defined as the ratio of correct positive classification to all correct classification as shown in (18):

$$Recall = \frac{Tp}{Tp + Fn} \tag{18}$$

Finally, the F1 score is one of the useful indexes to evaluate an algorithm. It permits to measure the weighted average of precision and recall scores. F1 score formula is presented in (19):

$$F1\ score = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{19}$$

B. Results and discussions

The same standardization methods are applied for both technologies. The objective of this comparison is to visualize the impact of standardization on data with different characteristics, at the same scale as well as on data with outliers. Tab. II and Tab. III show the results obtained using H_2/O_2 and H_2/Air database respectively but also the number of features needed to obtain the best results. Data obtained with H_2/O_2 stack are all in the same scale, which is not the case of data from H_2/Air stack. This is due to the fuel poisoning faults which, at high concentrations, generates data that are not on the same scale as the other faults. Results were obtained using the "Leave One Out Cross-Validation" (LOO CV) methodology. This allows getting as close as possible to a real use in which the EIS would be tested 1 by 1, but also, to use a maximum of data for training since the number of available data is low. In addition to results, Fig. 2 and Fig. 3 which represent the confusion matrix are also studied for a better understanding of misclassified data by the algorithm.

As Tab. II and Tab. III show, standardizing data permits to improve efficiency of diagnosis algorithm. Indeed, the choice of a correct standardization methodology allows improving the F1 score by about 12% and 30% for H_2/O_2 and H_2/Air stacks respectively.

TABLE II. VALIDATION RESULTS OBTAINED FOR H_2 / O_2 DATASET (LOO CV)

	Raw data	*Normalizer L2*	*Normalizer L1*	*Normalizer inf*	*Min-Max scaler*	*Max Absolute scaler*	*Standard scaler*
Accuracy	0.852	0.784	0.784	0.750	0.943	0.852	0.920
F1 score	0.852	0.781	0.780	0.745	0.943	0.853	0.920
Recall score	0.852	0.784	0.784	0.750	0.943	0.852	0.920
Precision score	0.859	0.797	0.797	0.759	0.947	0.856	0.922
Number of features	4	8	7	7	5	5	5

	Robust	*Yeo-Johnson*	*Normal Quantile*	*Uniform Quantile*
Accuracy	**0.977**	**0.966**	0.943	**0.955**
F1 score	**0.977**	**0.966**	0.943	**0.954**
Recall score	**0.977**	**0.966**	0.943	**0.955**
Precision score	**0.979**	**0.966**	0.948	**0.961**
Number of features	**6**	**6**	6	**5**

TABLE III. VALIDATION RESULTS OBTAINED FOR H_2 / AIR DATASET (LOO CV)

	Raw data	*Normalizer L2*	*Normalizer L1*	*Normalizer inf*	*Min-Max scaler*	*Max Absolute scaler*	*Standard scaler*
Accuracy	0.776	0.829	0.882	0.868	0.882	0.855	0.750
F1 score	0.774	0.832	0.883	0.864	0.879	0.851	0.753
Recall score	0.776	0.829	0.882	0.868	0.882	0.855	0.750
Precision score	0.816	0.855	0.886	0.866	0.891	0.861	0.763
Number of features	5	4	7	5	4	4	4

	Robust	*Yeo-Johnson*	*Normal Quantile*	*Uniform Quantile*
Accuracy	0.842	0.908	0.882	**0.961**
F1 score	0.842	0.908	0.883	**0.961**
Recall score	0.842	0.908	0.882	**0.961**
Precision score	0.866	0.909	0.891	**0.961**
Number of features	4	6	6	**6**

In the case of H_2/O_2 stack, the best results are provided by the main linear scaling methods and nonlinear transformations. However, it is interesting to note that the three normalizers generate more confusion in the algorithms (7 to 10% decrease of the F1 score compared to the case with raw data). This loss of performance means that samples are not different enough from each other to obtain good quality features. Max Absolute scaler doesn't improve classification results compared to other scalers which provide F1 score better than 90%. Nevertheless, only three methods allow for obtaining more than 95% of correct classification: Robust scaler, Yeo-Johnson and Uniform Quantile Transformer. The specificity of these three methods is that they take account of outliers which can be present in data even if they are all at the same scale.

Regarding the H_2/Air results, it can be observed that compare to the first database, normalizers improve classification results by 5-10% due to the presence of sample at different scales. However, compare to the first database, almost all standardization methods give results below 90%. In this configuration, poisoning fault highly impacts the standardization of data to have a correct standardization of them even if methods such as Robust scaler and Normal Quantile transformer are dedicated to reduce the outlier importance. Best methods are Yeo-Johnson and Uniform Quantile transformers which allow for obtaining better than 90% of correct classification but only Quantile transformer reach better than 95%.

The results obtained for both datasets confirm the weakness of normalizers and linear scalers in handling outliers. Normalizers need sufficiently different data to work, which makes them more efficient in dealing with these outliers, but the results obtained with them are insufficient compared to other standardization methods. Only the uniform quantile transformer performs well (>95%) for both datasets, making it a good candidate for generic use. In addition to Tab. II and Tab. III which show the results of each standardization method, Fig. 2 and Fig. 3 show the confusion matrix when the uniform quantile transformer is selected. With respect to the H_2/O_2 matrix, there are only confusions between the two starvation phenomena. Since the starvation faults lead to very similar alterations of the EIS, it is complicated to isolate them properly, so a confusion between the two phenomena is not surprising. Concerning the H_2/Air matrix, three data are misclassified. There is an inversion between the sulfur and carbon monoxide poisoning conditions that can be explained by the low severity of the fault condition

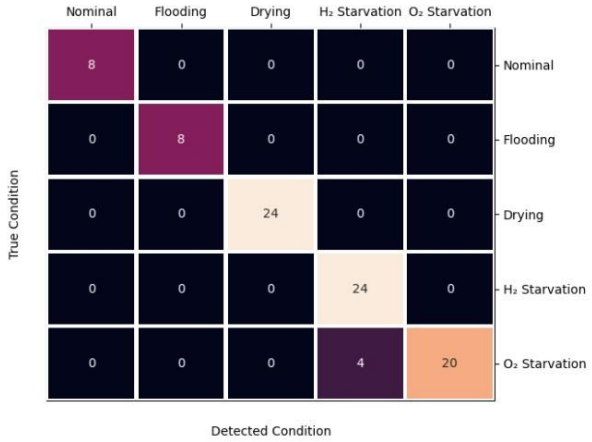

Fig. 2. Confusion matrix for H_2/O_2 dataset using uniform quantile transformer and the 5 best features

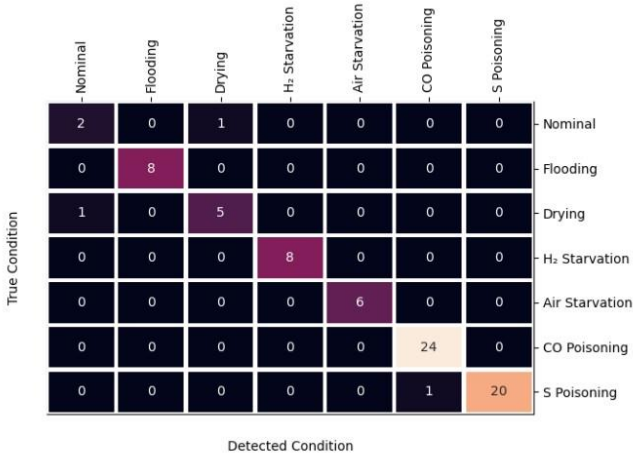

Fig. 4. Confusion matrix for H_2/Air dataset using uniform quantile transformer and the 6 best features

that makes them similar to each other. In addition to the confusion between the poisoning conditions, there are also two confusions between the nominal and drying conditions. At low intensity, drying is very similar to the nominal condition, but the confusion can also be explained by the fact that there are only three spectra in the nominal condition. This lack of data impacts the calculation of the cluster centers possibly resulting in a nominal cluster that is close to the drying ones.

V. Conclusion

This paper presents the effects of several standardizations on EIS extracted features on diagnostic algorithm. A relevant standardization of the data brings significant improvement to the diagnostic of fuel cells by Machine Learning. It can be observed from the results that uniform quantile transformer is a very powerful method giving good results on both technologies. For the H_2/O_2 dataset, which doesn't contain outliers, all linear scalers give similar results except for the normalizers ones which don't provide more than 80% of accuracy. For H_2/Air dataset which contains outliers, normalizers and linear scalers provide poor clustering performance while the uniform quantile transformer reaches the best F1 score (96%).

The present study has allowed to select an efficient standardization method, robust against the characteristics and weaknesses of the datasets. Future research will focus on another crucial step of the diagnostic algorithm which is determining a correct measure to automatically determine the best number of clusters for each fault in order to increase the unsupervised ability of the diagnosis.

Acknowledgment

This project has received funding from the Fuel Cells and Hydrogen 2 Joint Undertaking (JU) under grant agreement No 875047 Website: *https://www.rubyproject.eu/*

This work has been supported by the EIPHI Graduate School (contract ANR-17- EURE-0002) and the Region Bourgogne Franche-Comté.

We acknowledge the European project HEALTH CODE which provide data used in this paper. Website: *http://pemfc.health-code.eu/*

References

[1] "Fuel Cells," *Energy.gov*. https://www.energy.gov/eere/fuelcells/fuel-cells (accessed Jan. 17, 2021).

[2] Z. Zheng, M.-C. Péra, D. Hissel, M. Becherif, K.-S. Agbli, and Y. Li, "A double-fuzzy diagnostic methodology dedicated to online fault diagnosis of proton exchange membrane fuel cell stacks," *J. Power Sources*, vol. 271, pp. 570–581, Dec. 2014, doi: 10.1016/j.jpowsour.2014.07.157.

[3] L. A. M. Riascos, M. G. Simoes, and P. E. Miyagi, "A Bayesian network fault diagnostic system for proton exchange membrane fuel cells," *J. Power Sources*, vol. 165, no. 1, pp. 267–278, Feb. 2007, doi: 10.1016/j.jpowsour.2006.12.003.

[4] Z. Li *et al.*, "Online implementation of SVM based fault diagnosis strategy for PEMFC systems," *Appl. Energy*, vol. 164, pp. 284–293, Feb. 2016, doi: 10.1016/j.apenergy.2015.11.060.

[5] A. Escobet, À. Nebot, and F. Mugica, "PEM fuel cell fault diagnosis via a hybrid methodology based on fuzzy and pattern recognition techniques," *Eng. Appl. Artif. Intell.*, vol. 36, pp. 40–53, Nov. 2014, doi: 10.1016/j.engappai.2014.07.008.

[6] D. Chanal, N. Yousfi-Steiner, R. Petrone, D. Chamagne, and M.-C. Péra, "Online Diagnosis of PEM Fuel Cell by Fuzzy C-means clustering," *Encycl. Energy Storage*, p. 41.

[7] "Real operation pem fuel cells HEALTH-state monitoring and diagnosis based on dc-dc COnverter embeddeD Eis;H2020, European project, Horizon 2020; Health-Code." http://health-code.eu/ (accessed May 04, 2021).

[8] "RUBY – EU project." https://www.rubyproject.eu/ (accessed Feb. 02, 2021).

[9] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Mach. Learn. PYTHON*, p. 6.

[10] G. E. P. Box and D. R. Cox, "An Analysis of Transformations," *J. R. Stat. Soc. Ser. B Methodol.*, vol. 26, no. 2, pp. 211–252, 1964.

[11] I.-K. Yeo and R. A. Johnson, "A New Family of Power Transformations to Improve Normality or Symmetry," *Biometrika*, vol. 87, no. 4, pp. 954–959, 2000.

[12] M. Bicego and S. Baldo, "Properties of the Box–Cox transformation for pattern classification," *Neurocomputing*, vol. 218, pp. 390–400, Dec. 2016, doi: 10.1016/j.neucom.2016.08.081.

[13] T. J. Sefara, "The Effects of Normalisation Methods on Speech Emotion Recognition," in *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, Nov. 2019, pp. 1–8. doi: 10.1109/IMITEC45504.2019.9015895.