Efficient Selection of Data Samples for Fault Classification by the Clustering of the SOM

Diego P. Sousa†, Guilherme A. Barreto†, and Cláudio M. S. Medeiros‡

†Federal University of Ceará
Department of Teleinformatics Engineering
Campus of Pici, Center of Technology, Fortaleza, Ceará, Brazil
Emails: diegoperdigao@gmail.com, gbarreto@ufc.br

‡Federal Institute of Ceará (IFCE)
Department of Industry, Fortaleza, Ceará, Brazil
Email: claudiosa@ifce.edu.br

Abstract. In this paper we propose a sample selection procedure for improving accuracy of supervised classifiers in fault classification tasks. To generate faulty samples, a laboratory testbed is constructed and to avoid loss of a 3-phase AC induction motor (due to high short-circuit currents) resistors are used to limit current levels. This gives rise to short-circuit faults of different impedance levels, which may generate data samples difficult to classify as normal or faulty ones, specially if the faults are of high impedance (easily misinterpreted as non-faulty samples). Aiming at reducing misclassification, we use the clustering of the SOM approach [1] with modified information criteria for cluster validation. By means of comprehensive computer simulations, we show that the proposed approach is able to cluster successfully the different types of short-circuit faults and can be used for the purpose of sample selection.

1 Introduction

Several unsupervised machine learning algorithms, such as the self-organizing map (SOM) [2], the K-means [3], and the growing neural gas (GNG) [4], have been often used for data samples selection in building pattern classifiers. However, they have been mostly used as vector quantizers (VQ) for simple data volume reduction. In words, the original data samples are replaced either by the much smaller set of prototype vectors (to which are attached the dominant class labels), or by selecting just a few data samples around the learned prototypes.

In this paper we introduce an alternative methodology for generating a compact representative realistic datasets for fault detection/classification of a 3-phase AC induction motor. Instead of a VQ-based approach, we use a clustering strategy to dig deep down into the class labels distribution per cluster and decide for the removal of irrelevant or ambiguous samples. For this purpose, we use the clustering of the SOM approach introduced by [1] together with modified information criteria for finding a suitable number of clusters.

Our target task is the identification of inter-turn short-circuit faults in the stator winding, which we have been investigating lately using standard powerful nonlinear classifiers, such as the MLP and the SVM [5]. For this purpose, we built a lab scale testbed for simulating faults with different degrees of severity. In order to avoid loss of the electric motor (due to high short-circuit currents) resistors are used to limit current levels.

This approach gives rise to short-circuit faults of different impedance levels, some of them very difficult to classify as normal or faulty ones. This occurs particularly for high impedance faults, which can be misclassified as normal samples even by human experts, because the resulting short-circuit current is still low (a condition called incipient fault). By means of a careful selected example, we show that the proposed sample selection procedure is capable to increase considerably the accuracy rate of a linear classifier to the same level of those obtained by the aforementioned nonlinear classifier.

The remainder of the paper is divided into 5 additional sections. In Section 2, we briefly describe the clustering of the SOM approach. In Section 3, the basics of cluster validation techniques are presented and our proposal is introduced. In Section 4, it is described the experimental test bed from which the the original fault classification dataset was generated. In Section 5 the results are shown and discussed. The paper is concluded in Section 6.

2 Clustering of the SOM

As mentioned before, the SOM is essentially a vector quantization algorithm [6], which can be used as data reduction and information compress method. The SOM learns from examples a mapping from a high-dimensional continuous input space $\mathcal{X}$ onto a low-dimensional discrete space (output array) $\mathcal{A}$ of C neurons which are arranged in fixed topological forms, e.g., as a rectangular 2-dimensional array. The map $i^*(\mathbf{x}) : \mathcal{X} \rightarrow \mathcal{A}$, defined by the weight matrix $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_C\}, \mathbf{w}_i \in \mathbb{R}^d \subset \mathcal{X}$, assigns to the n-th input vector $\mathbf{x}_n \in \mathbb{R}^d \subset \mathcal{X}$ a winning neuron $i_n^* \in \mathcal{A}$, determined by

$$i_n^* = \arg\min_{\forall i} \|\mathbf{x}_n - \mathbf{w}_i\|, \tag{1}$$

where $\|\cdot\|$ denotes the Euclidean norm. In this paper, we use the batch learning algorithm for training the SOM. Hence, the weight vectors of all C neurons are adjusted at the end of the k-th epoch according to the following rule:

$$\mathbf{w}_i(k+1) = \frac{\sum_{n=1}^{N} h(i, i_n^*; k)\mathbf{x}_n}{\sum_{n=1}^{N} h(i, i_n^*; k)} \tag{2}$$

where $h(i, i_n^*; k)$ is the neighborhood function, defined here as

$$h(i, i_n^*; k) = \exp\left(-\frac{\|\mathbf{r}_i - \mathbf{r}_{i_n^*}\|^2}{2\sigma^2(k)}\right), \tag{3}$$

where $\mathbf{r}_i$ and $\mathbf{r}_{i_n^*}$ are respectively, the coordinates of the neurons i and i_n^* in the output array $\mathcal{A}$, and $\sigma(k) > 0$ defines the radius of the neighborhood function at the k-th epoch. The variable $\sigma(k)$ must decay with time to guarantee convergence of the weight vectors to stable steady states. In this paper, we adopt an exponential decay rule: $\sigma(k) = \sigma_0 \left(\sigma_T/\sigma_0\right)^{(k/T)}$, where σ_0 and σ_T are the initial and final values of $\sigma(k)$, respectively. Weight adjustment is performed until a steady state of global ordering of the weight vectors has been achieved. The resulting map preserves the topology of the input samples in the sense that adjacent data samples are mapped into adjacent regions on the map.

Once the SOM is trained, we can apply a standard clustering algorithm, such as the K-means, over the SOM prototypes. This approach is known as *clustering of the SOM* [1, 7] and can be understood as a two-stage unsupervised data processing approach. First, the SOM is used in order to generate a compact representation of the available dataset. Then, the K-means algorithm – with the help of cluster validation (CV) techniques – is applied over the SOM prototypes aiming at finding relevant clusters of prototypes. This hierarchical SOM-based scheme is supposed to facilitate cluster discovery by enhancing proximity relationships among data samples and filtering out irrelevant samples (e.g. outliers).

More specifically, the 2nd level of data processing requires the computation of $K = 2, \ldots, K_{max}$ partitions of the SOM prototypes and the corresponding values of the chosen cluster validity indices as well. The optimal partitioning, represented by K_{opt} partitions, is then chosen by the following search procedure:

$$K_{opt} = \arg \min_{K=2,\ldots,K_{max}} CV(\mathbf{W}, \mathbf{P}^K), \tag{4}$$

where $\mathbf{W} = \{\mathbf{w}_i\}_{i=1}^{C}$, $\mathbf{w}_i \in \mathbb{R}^d$, is the set of C prototypes of the SOM (1st level of data processing), while $\mathbf{P}^K = \{\mathbf{p}_j\}_{j=1}^{K}$, $\mathbf{p}_j \in \mathbb{R}^d$, denotes the set of K prototypes of the K-means algorithm (2nd level of data processing).

3 Basics of Cluster Validation Techniques

Techniques for cluster validation are used *a posteriori* to evaluate the results of a given clustering algorithm. It should be noted, however, that each cluster validation techniques has it own set of assumptions, so that the final results may vary across the chosen techniques.

3.1 Cluster Validity Indices

Some well-known indices available in the clustering literature are described next. We denote K as the number of clusters, K_{max} is the maximum allowed number of clusters, d as the number of features, $\bar{\mathbf{x}}$ as the centroid of the $d \times N$ data matrix $\mathbf{X}$, n_i as the number of objects in cluster C_i, $\mathbf{c}_i$ as the centroid of cluster C_i, and $\mathbf{x}_l^{(i)}$ as the l-th feature vector, $l = 1, \ldots, n_i$, belonging of the cluster C_i.
(i) The *Davies-Bouldin* (DB) index [8] is a function of the ratio of the sum of within-cluster scatter to between-cluster separation, and it uses the clusters'

centroids for this purpose. Initially, we need to compute the scatter within the i-th cluster and the separation between the i-th and j-th clusters, respectively, as

$$S_i = \left[\frac{1}{n_i} \sum_{l=1}^{n_i} \|\mathbf{x}_l^{(i)} - \mathbf{c}_i\|^2 \right]^{1/2} \quad \text{and} \quad d_{ij} = \|\mathbf{c}_i - \mathbf{c}_j\| \tag{5}$$

where $\|\cdot\|$ is the Euclidean norm. Finally, the DB index is defined as

$$DB(K) = \frac{1}{K} \sum_{i=1}^{K} R_i, \quad \text{where} \quad R_i = \max_{j \neq i} \left\{ \frac{S_i + S_j}{d_{ij}} \right\}. \tag{6}$$

The value of K leading to the smallest $DB(K)$ value is chosen as the optimal number of clusters.

(*ii*) The *Dunn* index [9] is represented generically by the following expression:

$$Dunn(K) = \frac{\min_{i \neq j} \{\delta(C_i, C_j)\}}{\max_{1 \leq l \leq k} \{\Delta(C_l)\}}, \tag{7}$$

where

$$\delta(C_i, C_j) = \min_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} \{d(\mathbf{x}, \mathbf{y})\}, \quad \text{and} \quad \Delta(C_i) = \max_{\mathbf{x}, \mathbf{y} \in C_i} \{d(\mathbf{x}, \mathbf{y})\}, \tag{8}$$

with $d(\cdot, \cdot)$ denoting a dissimilarity function (e.g. Euclidean distance) between vectors. Note that, while $\delta(C_i, C_j)$ is a measure of separation between clusters C_i and C_j, $\Delta(C_i)$ is a measure of the dispersion of data within the cluster C_i. The value of K resulting in the largest $Dunn(K)$ value is chosen as the optimal number of clusters.

(*iii*) The *Calinski-Harabasz* (CH) index [10] is a function defined as

$$CH(K) = \frac{trace(\mathbf{B}_K)/(K-1)}{trace(\mathbf{W}_K)/(N-K)} \tag{9}$$

where $\mathbf{B}_K = \sum_{i=1}^{K} n_i(\mathbf{c}_i - \bar{\mathbf{x}})(\mathbf{c}_i - \bar{\mathbf{x}})^T$ is the between-group scatter matrix for data partitioned into K clusters, $\mathbf{W}_K = \sum_{i=1}^{K} \sum_{l=1}^{n_i} (\mathbf{x}_l^{(i)} - \mathbf{c}_i)(\mathbf{x}_l^{(i)} - \mathbf{c}_i)^T$ is the within-group scatter matrix for data clustered into K clusters. The $trace(\cdot)$ operator computes the sum of the elements on the main diagonal of a square matrix. The value of K resulting in the largest $CH(K)$ value is chosen as the optimal number of clusters.

(*iv*) The *Silhouette* (Sil) index [11] is defined as

$$Sil(K) = \frac{\sum_{i=1}^{N} S(i)}{N}, \qquad S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \tag{10}$$

with $a(i)$ representing the average dissimilarity of the i-th feature vector to all other vectors within the same cluster (except i itself), and $b(i)$ denoting the lowest average dissimilarity of the i-th feature vector to any other cluster of which it is not a member. The silhouette can be calculated with any dissimilarity metric, such as the Euclidean or Manhattan distances. The value of K producing the largest $Sil(K)$ value is chosen as the optimal number of clusters.

3.2 Information criteria techniques

By understanding clustering as a data-driven process, one can make use of several criteria rooted in information theory for evaluating model selection procedures. Among the most common information criteria, we mention the *Akaike's Final Prediction Error* (FPE) [12], the *Akaike's Information Criterion* (AIC) [13], the *Bayesian Information Criterion* (BIC) [14] and the *Minimum Description Length* (MDL) [15]. These criteria are briefly described next in the context of order selection of a linear autoregressive (AR) model with p coefficients when fitted to a stationary time series.

The general expression of an information criterion for model selection purpose has the following form:

$$IC(p) = N \ln \left(\frac{RSS(p)}{N} \right) + \text{penalty}(p), \tag{11}$$

where N is the number of samples, p is the model order and $RSS(p)$ is the residual sum of squares[1] for a model with p parameters. As the number of parameters p increases, the first term on the right-hand side of Eq. (11) has a decreasing exponential trend as p increases. The second term of this equation acts as a penalty term for the excess parameters and, therefore, should exhibit an increasing tendency as p increases. The smaller the $IC(p)$, the better is the model selection.

Different choices for the penalty term give rise to different information criteria. For example, $\text{penalty}(p) = N \ln \left(\frac{N+p}{N-p} \right)$ (FPE), $\text{penalty}(p) = 2p$ (AIC), $\text{penalty}(p) = p \ln N$ (BIC), and $\text{penalty}(p) = \frac{p}{2} \ln N$ (MDL).

3.3 Efficient Use of Information Criteria in Clustering Tasks

As mentioned, all the information criteria previously presented were developed for order selection of AR models in time series modeling/prediction tasks. In this paper we modify the information criteria by proposing a simple modification that allowed us to correctly cluster the faulty samples into different categories. In summary, we replace the $RSS(p)$ with a more suitable figure of merit for clustering and vector quantization tasks. In this regard, we selected the *mean squared quantization errors* (MSQE) for N training data samples:

$$MSQE(p) = \frac{1}{N} \sum_{i=1}^{K} \sum_{l=1}^{n_i} \|\mathbf{x}_l^{(i)} - \mathbf{c}_i\|^2, \tag{12}$$

where we additionally set the number of parameters $p = K \times d$, with K denoting the number of clusters and d is the dimension of the feature vector.

[1] Also known as sum of squared residuals (SSR) or the sum of squared errors of prediction (SSE).

4 Experimental Test Bed

A 3-phase squirrel-cage induction motor built by WEG[2] industry is used in this study. Its main characteristics are 0.75 kW (power), 220/380 V (nominal voltage), 3.02/1.75 A (nominal current), 79.5% (efficiency), 1720 rpm (nominal rotational speed), Ip/In = 7.2 (peak to nominal current ratio), and 0.82 (power factor). The dataset is generated with this motor operating in different working conditions. The modules of the laboratory scale test bed are shown in Fig. 1, and are hereafter explained.

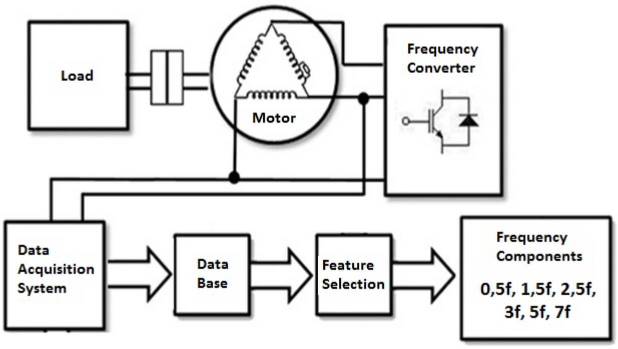


Fig. 1. Modules of the laboratory test bed and the data acquisition system.

Firstly, a Foucault's braking system is used in order to apply three different levels of load: 0% (no load), 50% of nominal load and 100% (full load). In order to vary the speed of the motor, a frequency converter (also known as inverter drive) WEG CFW-09 is utilized with seven different frequencies: 30 Hz, 35 Hz, 40 Hz, 45 Hz, 50 Hz, 55 Hz and 60 Hz. It is worth mentioning that only open loop operation is used with this frequency converter. Three Hall effect sensors are used to measure the line currents of each phase of this frequency converter.

The motor was rewound so that some extra taps were made available by exposing the stator winding turns of each phase. This was done in order to simulate different inter-turn short-circuit scenarios. In this work, three different levels of fault are used. In the lowest level (level 1), 5 turns were short-circuited, totaling 1.41% of the turns of one phase. In the intermediate level (level 2), 17 turns (4.8%) were short-circuited. Finally, in the highest level (level 3), 32 turns (9.26%) were short-circuited.

An auxiliary command system was built to execute two kinds of short-circuit schemes: high impedance (aiming at simulating the initial low-current state of a short-circuit) and the low impedance. With these two short-circuit schemes and three levels of faults, there are six different fault conditions of the motor.

[2] `http://www.weg.net/institutional/BR/en/`

Short-circuit current levels leading to either low or high impedance faults are controlled by resistors in order to protect the motor from permanent damages.

All the operation conditions of the motor are shown in Table 1, where the load level applied to the motor, the phase identification, the frequency of the voltage applied by the frequency converter and the *fault extension* are specified. In this last operation condition, the letter **H** denotes a high impedance fault, the letter **L** denotes a low impedance fault, while the numbers 1, 2 and 3 stands for the level of the fault. All these conditions sums up to total of 441 (3 × 3 × 7 × 7) time domain sample vectors.

Table 1. Tested operational conditions of the motor for data generation.

Load Level	0%	50%	100%	–	–	–	–
Converter Phase	Phase 1	Phase 2	Phase 3	–	–	–	–
Converter Frequency	30Hz	35Hz	40Hz	45Hz	50Hz	55Hz	60Hz
Fault Extension	Normal	HI1	HI2	HI3	LI1	LI2	LI3

As shown in Fig. 1, the motor was delta connected. In this configuration, two line currents of the frequency converter are directly connected to the faulty phase of the motor. As we aim at developing a monitoring system able to detect faults using just one phase of the converter, just one of these previously mentioned phases was used in order to avoid redundancy of information. Thus, 294 samples are used: 147 from phase 1 (directly connected to the fault current) and 147 from phase 3 (indirectly connected to the fault current). As can be inferred from Table 1, the task of interest can be approached as a multiclass problem, if one considers each fault extension as a class (normal, H1, H2, H3, L1, L2, L3). As such, each class has 42 samples. Alternatively, one can rearrange data samples into three classes, namely: normal condition (with 42 samples), high impedance fault (with 126 samples, merging the classes HI1, HI2 and HI3) or low impedance fault (also with 126 samples, merging the classes LI1, LI2 and LI3).

In the dataset, by a "sample" we mean a current signal stored as a vector of 100,000 components, resulting from 10 seconds of acquisition with a 10 kHz sampling frequency. To generate the feature vectors for classification purposes, the Fast Fourier Transform (FFT) is used. The procedure for building the feature vector for each current signal is comprised of the following steps:

Step 1 - Define the load condition of the motor.
Step 2 - Define the fundamental frequency (f_c) of the converter drive.
Step 3 - Read the current signal for 10s at a 10KHz sampling rate.
Step 4 - Apply the FFT to the current signal. Since the output of the FFT is comprised of a sequence of complex numbers, take their absolute values.
Step 5 - Find the frequency corresponding to the maximum value of the computed spectrum. Denote it as $\hat{f}_c$, since it is an estimate of f_c (see Step 2).
Step 6 - Build the associated 6-dimensional feature vector by selecting the corresponding FFT output values for the following harmonics of $\hat{f}_c$: $\{0.5\hat{f}_c$, $1.5\hat{f}_c$, $2.5\hat{f}_c$, $3\hat{f}_c$, $5\hat{f}_c$, and $7\hat{f}_c\}$.

In summary, the dataset is comprised of 294 6-dimensional labeled feature vectors, in which the attribute values represents the FFT values for the chosen 6 harmonics of the fundamental frequency of the converter drive. It should be mentioned, however, that we are interested in data selection by clustering methods. For this purpose, unlabeled data is presented to the evaluated clustering algorithms. We want to know if the evaluated clustering algorithms with the help of presented cluster validation techniques are able to distinguish between high- and low-impedance faults (without class information). This is a particularly challenging task because high-impedance faults can be easily misinterpreted as normal (i.e. non-faulty) samples, as observed in our previous works on fault classification [16, 5]. We hypothesize that by selecting those samples correctly identified by the clustering algorithms, we can further improve the recognition rates of supervised classifiers trained with the selected dataset.

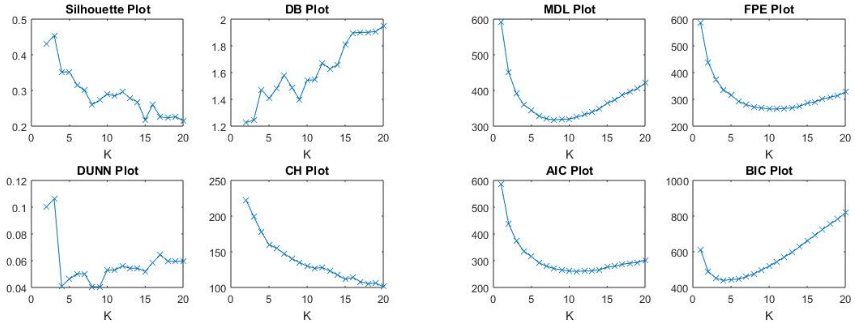
5 Results and Discussion

In this section we apply the clustering of the SOM technique to find *dominant clusters* for the 3 types of classes existing in the generated dataset, which are represented by the labels Normal (NO), High Impedance (HI) and Low Impedance (LI). By *dominant clusters*, we mean either clusters containing only samples of 1 (out of 3) class, or clusters in which there is a clear dominance of one class label over the two others. Samples belonging to non-dominant clusters are removed from the dataset. By means of this removal procedure, we hope to end up with a set of representative samples for the problem of interest that can be used for improving recognition rates of supervised classifiers. Experiment with a linear classifier corroborates our hypothesis.

The training methodology is comprised of two stages. In the first stage, the training of a 10 × 10 SOM network is repeated for 50 independent runs[3]. The initial and final values of the width of the Gaussian neighborhood are set to $\sigma_0 = 5$ and $\sigma_T = 1$, respectively. Since our goal is data selection via clustering, all the available data samples are used to train the SOM. Each training run lasts 100 epochs, with the MSQE (see Eq. (12)) computed at the end of training. Once the 50 training runs are finished, we select the SOM weights $\{\mathbf{w}_i\}_{i=1}^{C}$ that produced the lowest MSQE value. For the second stage, we apply the K-means algorithm over the chosen SOM prototypes for $K = 2, 3, \cdots, K_{max} = 20$. For each value of K, 50 independent runs of the K-means algorithm are executed. For a specific K, we choose (out of 50) the set of prototypes $\{\mathbf{p}_j\}_{j=1}^{K}$ that produces the lowest MSQE. Using this selected set, we compute the corresponding values of the cluster validity indices and the information criteria.

As can be seen in Fig. 2(a), the optimal number of clusters suggested by the DB and CH indices is $K_{opt} = 2$, while the Silhouette and Dunn indices suggested $K_{opt} = 3$. The corresponding class label distributions per cluster are shown in Table 2. By analyzing this table, one can easily see that there is no

[3] This size was chosen because it corresponds approximately to one third of the total number of data samples.



(a) Validity Indices.

(b) Information Criteria.

Fig. 2. Values of the validity indices and information criteria for different values of K.

Table 2. Class label distributions per cluster for $K_{opt} = 2$ and $K_{opt} = 3$.

	$K_{opt} = 2$		$K_{opt} = 3$		
Labels	**Cluster 1**	**Cluster 2**	**Cluster 1**	**Cluster 2**	**Cluster 3**
NO	11	31	27	11	4
HI	42	84	72	42	12
LI	43	83	71	43	12

Table 3. Class label distributions per cluster for $K_{opt} = 8$.

Labels	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8
NO	14	0	9	9	4	0	0	6
HI	7	0	30	21	12	35	0	21
LI	5	20	23	20	10	18	16	14

dominant cluster at all, neither for $K_{opt} = 2$, nor for $K_{opt} = 3$. In words, we note that for all clusters most class labels belong to samples representing high- and low impedance faults (in approximately equal proportions). Worse, there is no dominant group of normal samples. Thus, the data partitions recommended by the cluster validity indices are not useful for our purposes of sample selection.

We then decided to tackle this awkward situation by investigating the results provided by the information criteria using the modification proposed in Subsection 3.3. As shown in Fig. 2(b), the optimal number of clusters suggested by MDL, FPE and AIC is within the interval from 8 to 10, while the BIC suggested that the K_{opt} is between 4 and 6.

Following a majority voting scheme, we carried out careful analyses of the resulting partitions for $K_{opt} = 8$, 9 and 10. We then verified that the most coherent data partitions were obtained for $K_{opt} = 8$. The corresponding class labels distribution is shown in Table 3.

Table 4. Class label distributions of subgroups of faulty samples within Cluster 1.

	NO	HI1	LI2	LI3
Cluster 1	14	7	3	2

Table 5. Class label distribution across the clusters of the cleaned dataset

Labels	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8
NO	14	0	0	0	0	0	0	0
HI	0	0	30	21	12	35	0	21
LI	0	20	23	20	10	18	16	14

A closer look at this table reveals the occurrence of a cluster comprised exclusively of LI faulty samples (Cluster 2). Cluster 6 and Cluster 7 are comprised exclusively of faulty samples (LI and HI). Clusters 3, 4, 5 and 8 are comprised predominantly of faulty samples, with just a few normal samples being mapped to these clusters. The normal samples of Clusters 3, 4, 5 and 8 are strong candidates to be removed from the original dataset.

Cluster 1 demands a deeper analysis. In Table 4 we show the class label distribution of subgroups of faulty samples. These subgroups correspond to different levels of severity of the faults (see Table 1). That said, we observe in Cluster 1 that the number of normal samples is far more expressive than the LI ones. Recall that LI samples differ considerably from normal ones due to the high short-circuit current they produce. If these LI samples are grouped together with normal samples by the clustering algorithm, this occurs probably because the LI samples have been mislabeled by the human experts. Thus, the five LI samples in this cluster (3 samples labeled as LI2 and 2 samples as LI3) are strong candidates to be removed from the original dataset.

In what concern the HI samples mapped to Cluster 1, they are all of low intensity (i.e. they produce low short-circuit current), a condition that could be easily misinterpreted as normal by a human expert. Thus, the 7 samples labeled as HI1 are tagged as strong candidates to be removed from the original dataset.

Finally, by removing the faulty samples from Cluster 1 and the normal samples from Cluster 3, Cluster 4, Cluster 5 and Cluster 8, we eventually produced a new *cleaned* 3-class dataset containing 254 samples: 14 labeled as normal, 119 labeled as HI and 121 labeled as LI. The class label distribution of the cleaned dataset can be seen in Table 5.

The ultimate experiment for validating our hypothesis[4] consists in comparing the performance of a given classifier when trained with the original and cleaned datasets. For this purpose, we use the simple linear least squares (LS) classifier and treated the problem as a binary classification by merging the LI and HI samples into a single class. Thus, each sample is labeled as +1 if it represents

[4] That of using the clustering of the SOM for sample selection.

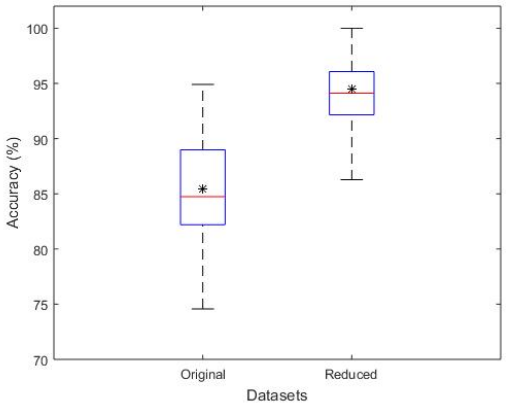


Fig. 3. Boxplots of the distribution of the accuracy rate achieved by the linear LS classifier for the original dataset and for the cleaned one.

normal operation condition or as −1 it corresponds to a faulty condition. For each dataset (original and cleaned), 100 independent training-testing runs are executed. For each run, the samples are randomly divided into two groups: 80% for training and 20% for testing.

The boxplots of the correct classification (i.e. accuracy) rate for test data along the 100 runs are shown in Fig. 3. The central mark is the median of the distribution, the asterisk is the mean value. by comparing the two boxplots in the figure, it can be seen that the median value of the accuracy rate improved considerably and the accuracy dispersion decreased for the cleaned dataset. Numerically, the median of the accuracy rate for the original dataset was 84.7%, while for the cleaned one it reached 94.1%.

As a final remark, it is worth mentioning that the high accuracy rate achieved by the linear classifier on the cleaned dataset is equivalent to the ones achieved by nonlinear classifiers (e.g. MLP and SVM) in previous works on the original dataset. This way, we can conclude that the proposed sample selection procedure succeeded in achieving its goal.

6 Conclusions and Further Work

In this paper, we introduced a clustering-based approach for data sample selection aiming at improving accuracy rates of pattern classifiers on fault classification tasks. The target task of detecting inter-turn short-circuit faults is challenging (even for human experts) because of the high probability of misinterpretation of high impedance faults as normal ones. We succeeded in reporting a sharp increase in the accuracy rate of a linear classifier when used the cleaned dataset. The achieved rates were equivalent to those obtained by powerful nonlinear classifiers. Currently, we are investigating a fuzzy clustering approach to the same sample selection task, with the hope of further increase the accuracy rates for different types of classifiers.

Acknowledgments: The authors thank CNPq (grant 309451/2015-9), UFC and IFCE for the financial support.

References

1. J. Vesanto and E. Alhoniemi, “Clustering of the self-organizing map,” *IEEE Transactions on neural networks*, vol. 11, no. 3, pp. 586–600, 2000.
2. A. R. R. Neto and G. A. Barreto, “Opposite maps: Vector quantization algorithms for building reduced-set SVM and LSSVM classifiers,” *Neural Processing Letters*, vol. 37, no. 1, pp. 3–19, 2013.
3. G. H. Nguyen, S. Phung, and A. Bouzerdoum, “Efficient SVM training with reduced weighted samples,” in *Proceedings of the 2010 IEEE World Congress on Computational Intelligence (WCCI'2010)*, 2010, pp. 1764–1768.
4. A. L. Suarez-Cetrulo and A. Cervantes, “An online classification algorithm for large scale data streams: iGNGSVM,” *Neurocomputing*, vol. 262, pp. 67–76, 2017.
5. D. N. Coelho, G. A. Barreto, C. M. S. Medeiros, and J. D. A. Santos, “Performance comparison of classifiers in the detection of short circuit incipient fault in a three-phase induction motor,” in *Proceedings of the 2014 IEEE Symposium on-Computational Intelligence for Engineering Solutions (CIES'04)*, 2014, pp. 42–48.
6. T. Kohonen, “Essentials of the self-organizing map,” *Neural Networks*, vol. 37, pp. 52–65, 2013.
7. S. Wu and T. W. S. Chow, “Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density,” *Pattern Recognition*, vol. 37, no. 2, pp. 175–188, 2004.
8. D. L. Davies and D. W. Bouldin, “A cluster separation measure,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 2, pp. 95–104, 1979.
9. J. C. Dunn, “A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters,” *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.
10. R. B. Calinski and J. Harabasz, “A dendrite method for cluster analysis,” *Communications in Statistics*, vol. 3, no. 1, pp. 1–27, 1974.
11. P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, no. 1, pp. 53–65, 1987.
12. H. Akaike, “Fitting autoregressive models for prediction,” *Annals of Institute of Statistical Mathematics*, vol. 21, pp. 243–247, 1969.
13. ——, “A new look at the statistical model identification,” *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.
14. G. E. Schwarz, “Estimating the dimension of a model,” *Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.
15. J. Rissanen, “Modeling by shortest data description,” *Automatica*, vol. 14, pp. 465–471, 1978.
16. D. N. Coelho and C. M. S. Medeiros, “Short circuit incipient fault detection and supervision in a three-phase induction motor with a SOM-based algorithm,” in *Advances in Self-Organizing Maps.* Springer, 2013, pp. 315–323.