# Partial discharges and noise classification under HVDC using unsupervised and semi-supervised learning

N Morette, L C Castro Heredia, Thierry Ditchi, A Rodrigo Mor, Y Oussar

Electrical Power and Energy Systems 121 (2020) 106129

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

# Partial discharges and noise classification under HVDC using unsupervised and semi-supervised learning

N. Morette[a,*], L.C. Castro Heredia[b], Thierry Ditchi[a], A. Rodrigo Mor[b], Y. Oussar[a,*]

[a] Laboratoire de Physique et d'Étude des Matériaux (LPEM), ESPCI Paris, PSL Research University, CNRS, Sorbonne Université, France
[b] Department of Electrical Sustainable Energy Delft University of Technology Delft, the Netherlands

ARTICLE INFO

*Keywords:*
Semi-supervised learning
Transductive SVMs
K-means
Dunn index
Partial discharges
HVDC

ABSTRACT

This paper tackles the problem of the classification of partial discharge (PD) and noise signals by applying unsupervised and semi-supervised learning methods. The first step in the proposed methodology is to prepare a set of classification features from the statistical moments of the distribution of the Wavelet detail coefficients extracted from a dataset of signals acquired from a test cell under 40 kVDC. In a second step, an unsupervised learning framework that implements the k–means algorithm is applied to reduce the dimensionality of this initial feature set. The Silhouette index is used to evaluate the number of natural clusters in the dataset while the Dunn index is used to determine which subset of features produces the best clustering quality. Since the unsupervised learning does not provide any method for result validation, then the third step in the methodology of this paper consists of applying a semi-supervised learning framework that implements Transductive Support-Vector Machines. The labeling of the test set that is required in this framework for the result validation is carried out by visual checking of the signal waveforms assisted by GUI tools such as the software *PDflex*. The results using this methodology showed a high classification accuracy and proved that both learning frameworks can be combined to optimize the selection of classification features.

## 1. Introduction

Partial Discharge (PD) phenomena and measurements have become a vital technique to assess the condition of the insulation of High-Voltage (HV) power apparatus and cables [1,2]. In this context, accurate measurement of PD activity is crucial to ensure a reliable monitoring and diagnostics of the insulation of HV equipment. Under DC voltage, PD events recur far less frequently than under AC conditions. In order to acquire enough data for diagnosis under DC, acquisition time are longer than under AC voltage. Therefore, the risk of triggering the acquisition on a noise signal instead of a PD is much more important [3,4,5]. Thus, errors in the interpretation of PD measurements are more likely to happen under DC voltage and may lead to false conclusions in the diagnostics (e.g., unnecessary disconnections of the equipment or unexpected failures).

Partial discharge measurements by unconventional systems [6] pose the problem of recording PD and non-PD signals jointly during one single measurement. Therefore, the post-processing of the data demands classification techniques. Several approaches have been developed in order to discriminate different PD and noise sources, all of them are based on the extraction of characteristic parameters from individual registered pulses. Supervised classification tools have shown very good results for noise and PD discrimination purposes. In [7], the authors use neural network (NN) for the automatic discrimination of partial discharge (PD) signal from external noise in PD measurements of XLPE cables under AC. In this study, the input pattern of the NN is directly related to the three-dimensional phi-q-n profiles of already known PD and noise pulses detected in the experiment. The NN which separately learned both PD and noise patterns discriminated unknown PD patterns from accompanying external noise with a correct response rate of only 52% in average. The correct responses of the NN rose to 89% in average when the NN learned PD patterns inclusive of external noise instead of those without noise. The NN could correctly discriminate all unknown input patterns for a signal to noise ratio greater than or equal to unity. However, these techniques require a previous manual labeling of the data by the user. In many classification problems with large datasets, the manual labeling of data is a labor-intensive task. Moreover, it can lead to human errors, especially when signals are not easily distinguishable, resulting in identification problems. In order to increase the unsupervised character of PD monitoring, there have been strong efforts to develop and improve PD and noise separation techniques using different unsupervised clustering methods.

* Corresponding authors.
*E-mail addresses:* nathalie.morette@espci.fr (N. Morette), yacine.oussar@espci.psl.eu (Y. Oussar).

https://doi.org/10.1016/j.ijepes.2020.106129
Received 9 October 2019; Received in revised form 20 March 2020; Accepted 19 April 2020
Available online 14 May 2020

Wavelets techniques, spectral power ratios analysis, time frequency maps, among many other more, have been applied for the extraction of features, that combined with different unsupervised clustering algorithms have shown good results for PD and noise signals separation in multiple experimental setups.

For example, authors in [8] use a power ratio approach where the total spectral power and the power ratio in selected frequency bands of each detected pulse are calculated and represented in a 2D map to identify the PD and noise sources. Pulse source identification are verified using PRPD patterns (Phase-Resolved PD patterns) for three typical types of PD sources: corona, surface, internal discharges and noise as well.

In this paper, spectral power analysis was demonstrated to be a promising technique for PD and noise identification in high frequency measurements. Signal power ratios result in clearly different clusters for noise and discharges for all the test objects studied. However, the identification requires the associated PRPD to each PR cluster for the identification of the phenomenon.

In [9], a new pulse classification tool based on the waveform analysis of the recorded signals is presented. Three characteristic parameters are calculated for each pulse; one characterizes their frequency content while the other ones describe the waveform of their normalized associated envelope. A graphical tool based on two, three-dimensional representations of the characteristic parameters, makes it possible to identify different types of defects, and noise sources simultaneously present in a test object. For each cluster, its individual PRDP pattern has been obtained and enable the identification of the different PD and noise sources involved in the cable systems.

In [10], a PD and noise identification method based on TF (Time-Frequency) map is used. The data are obtained from measurements relevant to cable models having an artificial defect made by knife cut. The TF map allows an effective pulse separation and noise rejection under DC.

In [11] the wavelet-decomposition and PCA method was applied to pulses produced by known noise and PD sources during experiment. The three main energies of the signals associated with each decomposition level were selected using PCA and used to form a 3-D plot. Three different clusters were obtained on the 3D map. One cluster corresponds to pulses produced by micro-voids within the test samples and the two other ones are due to noise signals. The application of DBSCAN allowed the optimum separation of the different groups minimizing the losses of isolated data. This proposed algorithm proved to be effective at separating different PD sources and noise and the analysis of the PRPD patterns confirmed the quality of separation.

As mentioned in these previous studies, the clustering results were verified using a database of well-known phase-resolved (PRPD) or time-resolved PD patterns (TRPD) for AC or DC respectively. These typical patterns are commonly used as a reference for visual verification. In addition, the waveforms can also serve the purpose of validating the results. Nonetheless, the visual or manual validation process may grow in complexity as the datasets become larger. Moreover, PRPD or TRPD patterns are able to identify different PD and noise sources when the noise level is low compared to the amplitudes of the PDs. However, real insulation systems usually exhibit several PD sources and the noise level is high, especially if measurements are performed on-line.

As important as the validation of the results is the selection of classification features. In general, a feature can be any attribute that better describes a class. After a space of features has been defined, the next steps are to determine the optimal number of clusters and the application of criterion metrics that evaluate the clustering quality. In this study, this procedure is researched by using waveforms acquired from a surface-test-cell under 40 kVDC. The space of features comprises the statistical moments mean, standard deviation, skewness and kurtosis of the Wavelet detail coefficient distribution for five levels of decomposition. The Silhouette index is employed to determine the optimal number of clusters as the input of the supervised k-means clustering algorithm and the Dunn index serves as quantifier of the cluster quality and to reduce the dimensionality of the feature space.

Since the waveform of each signal in the experimental dataset is available, in the second part of this paper, a semi-supervised classification technique based on Transductive Support-Vector Machines (TSVMs) is implemented. An advantage of semi-supervised learning is that a test set can be built from labeled data to evaluate the classification performance of the algorithm. This contributes to reduce the complexity of clustering results validation in unsupervised learning and in turn, the validation can be performed automatically, without the need of visual verification from an expert. Semi-supervised learning has recently become popular due to the variety of cases where a lot of unlabeled data are available, for example text classification [12] or image processing [13]. However, this field has not been fully investigated for Partial Discharge monitoring and especially for PD-noise pattern classification. This procedure exploits both labeled and unlabeled data to build the best classifier for PD-noise discrimination. Moreover, it requires only a reduced set of labeled data compared to unlabeled data. In our approach, we use the values of peak amplitude and charge from the signals [14] to assist the user in the labeling of test set. A dataset of 100 PD signals and 100 non-PD signals were so labeled. Finally we discussed the high classification performance achieved by labeling a small share of the dataset.

## 2. Test set-up and dataset

### 2.1. Experimental setup

For this study, an unconventional PD measuring system was used in combination with a test set-up to produce surface discharges as shown in Fig. 1. A testing voltage of 40 kVDC was applied to the test cell filled with $SF_6$ at 3 bar pressure. Upon a partial discharge event, a current pulse flows along the high-frequency and low-impedance path provided by the coupling capacitor of 500 pF. An High Frequency Current Transformer (HFCT) sensor placed in this current loop measures the PD current. The sensor was built on a N30 ferrite core which had 5 turns of 3 mm copper stripes wound onto it. This construction resulted in a bandwidth of the HFCT is 62 kHz–136 MHz and its gain is 9.1 V/A. The measured frequency response and pictures of the construction of the sensor can be found in [15].

As can be seen in Fig. 1, the output of the HFCT was fed directly into one channel of the oscilloscope MSO Series 5 from Tektronix. Individual waveforms were acquired via FrastFrame mode of the oscilloscope. Thus, 4993 single signals were captured and transferred as a matrix of [4993 × 6314], being 6314 the samples in each single signal. The length of the pulses was approximately 1us, sampled at a rate of 6.25

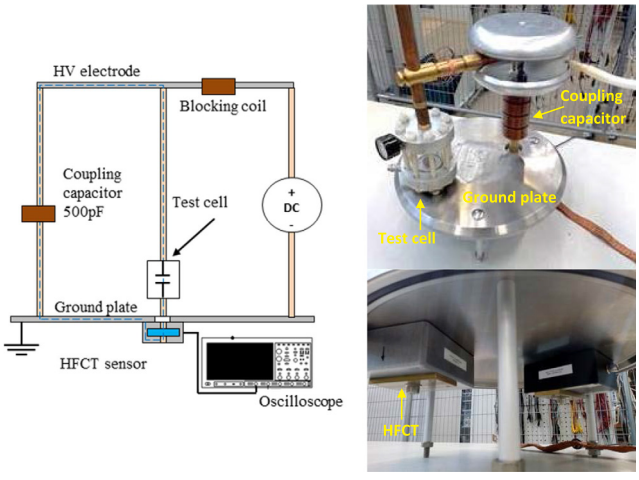


**Fig. 1.** PD measuring setup of surface discharges in $SF_6$ under DC voltage.

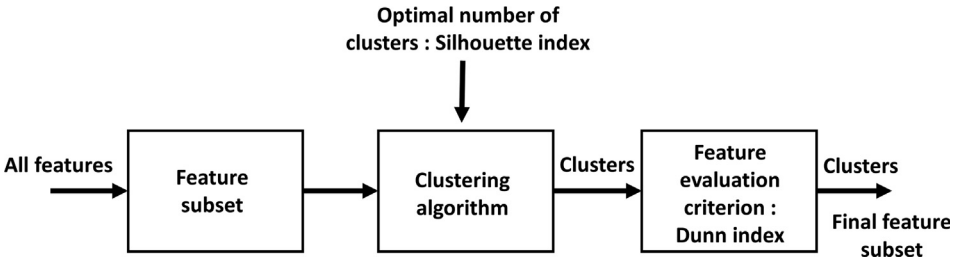


**Fig. 2.** Framework for unsupervised learning.

GSa/s. The experiments were conducted in a non-shielded room which resulted in acquisition of both PD and non-PD signals (hereafter the non-PD signals will be referred to as noise).

### 2.2. Features extraction and building of the database

One of the most challenging issues in clustering and classification problems is to extract informative features from measurements. Wavelet analysis has demonstrated high efficiency for the extraction of relevant features from PD data hence the reason it is commonly applied to PD denoising in HV equipment [16,17,18] and defect recognition [19].

A typical Discrete Wavelet Transform (DWT) decomposition equation can be formulated as:

$$DWT(m, k) = \frac{1}{\sqrt{a}} \sum_{n=0}^{N-1} s(n) g^*\left(\frac{n-b}{a}\right) \quad (1)$$

where $s(n)$ is the original signal, $N$ is the number of samples in the windowed signal, $g(.)$ is the mother wavelet function, $a = 2^m$ and $b = k2^m$ are the scaling and translation parameters where $m$ is the decomposition level index and $k \in \mathbb{Z}$. * denotes the complex conjugate. DWT can be interpreted as a multi-stage filter process that decomposes the original signal into high and low frequency components using series of high-pass and low-pass filters. The coefficients obtained after the high-pass filters are called detail coefficients and those after the low-pass filters are the approximation coefficients. At each level, the approximation/detail coefficients represent a filtered signal spanning half of the frequency band. The decomposition is repeated to further increase the frequency resolution until the desired decomposition level is achieved. The mother wavelet used in this work is the Daubechies wavelet because it is suitable for the analysis of fast transients, non-periodic pulses such as compactness, limited duration, orthogonality and asymmetry [20].

The selection of the initial features to be used as input of the classification algorithm is done in a heuristic manner. In fact, to avoid biases choices, it is important to first consider a large panel of features, also features that were not considered to be relevant in a first approach. After, feature selection techniques will be applied to choose the most relevant variables.

In this contribution, each of the 4993 signals in the dataset was decomposed by using the 'db10′ version of Daubechies wavelet [20] and the detail coefficient distributions up to the fifth level; cD1, cD2, cD3, cD4 and cD5 were used as signal features. This large data set was further reduced in dimensionality by representing the $cD_i$ vectors by their statistical moments mean, standard deviation, skewness and kurtosis.

The mean and standard deviation are defined as followed:

$$x_{cD(i,j)} = \frac{1}{N} \sum_{n=1}^{N} cD_{i,j}(n) \quad (2)$$

$$\sigma_{cD(i,j)} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} [cD_{i,j}(n) - x_{cD(i,j)}]^2} \quad (3)$$

where $cD_{i,j}$ is the n-th detail coefficient at level j, extracted from the i-th signal and $N$ is the total number of detail coefficients at level j.

The distributions of the detail coefficients at each level of decomposition have different shapes that can be described using the skewness and kurtosis. If the skewness is positive, the coefficients are positively skewed, meaning that the right tail of the distribution is longer than the left. If the skewness is negative, the coefficients are negatively skewed, meaning that the left tail is longer. If skewness = 0, the distribution is symmetric. The kurtosis can be explained in terms of the central peak of the distribution. Higher values indicate a higher, sharper peak while lower values indicate a lower, less distinctive peak.

Thus, the original feature dataset for each signal was reduced from $5xn_{cD}$ ($n_{cD}$ the number of detail coefficients) to $5 \times 4$ features.

## 3. Noise discrimination using unsupervised learning

### 3.1. Framework for feature selection

While feature selection is a well-studied problem in the area of supervised learning, it is less understood in unsupervised learning where no class labels are available to verify the feature extraction. All the 20 extracted features may not be relevant, some may be redundant and some can even misguide clustering algorithms [21]. In this section, a framework is proposed for unsupervised feature selection. This framework is illustrated in Fig. 2. The idea behind this approach is to cluster the data using each candidate feature subspace according to a certain criterion, and select the subspace that gives the best clustering quality with the minimum number of features.

To select the feature subset that best discovers relevant groupings from data, we need a measure to assess cluster quality. In this work, the criterion selected is the Dunn index. This metric considers both the separation between cluster centroids and the dispersion of the element in the clusters. Thus, it provides a good measure of how well clusters are separated and compact.

The Dunn index [22] is defined as follow:

$$D = \min_{1 \le i \le n} \left\{ \min_{1 \le j \le n_{i \ne j}} \left\{ \frac{\delta(C_i, C_j)}{\max_{1 \le k \le n} \Delta C_k} \right\} \right\} \quad (4)$$

$n$ is the number of clusters, $\delta(C_i, C_j)$ is the inter-cluster distance metric between clusters $C_i$and $C_j$. $\Delta C_k$ is a measure of the cluster dispersion (which can be defined as the diameter of the cluster). Compact and well-separated clusters exhibit a large Dunn index value.

First, all features are used separately as input of the clustering algorithm. The feature that provides the largest Dunn index value is selected. The same process is repeated for all possible couples, triplets and quadruplets of the 20 features. This combinatorial evaluation method selects the combination of features that give the best criterion value.

### 3.2. k-means Algorithm

K-means is a commonly used clustering algorithm in Partial Discharge studies [23]. This algorithm requires the user to specify the

number of clusters $k$ to be generated. Since the objective is to separate noise from PD signals regardless of the possible several sub-categories inside the PD and noise groups, we assume that the number of clusters is two. Silhouette analysis [24] is used to verify this assumption given our dataset. It measures the separation distance between the resulting clusters for different values of $k$. The Silhouette index has a range of [-1,1] where a high value indicates that the object is well matched to its own cluster and poorly matched to the others. It is defined as follows:

$$s(i) = \begin{cases} 1 - \frac{a(i)}{b(i)}, & if\ a(i) < b(i) \\ 0, & if\ a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1, & if\ a(i) > b(i) \end{cases} \quad (5)$$

Where $a(i)$ is the average distance between $i$ and all other data points in the same cluster.

$$a(i) = \frac{1}{|C_i - 1|} \sum_{j \in C_i, i \neq j} d(i,j) \quad (6)$$

$d(i,j)$ is the distance between data points $i$ and $j$ in the cluster $C_i$. The smaller the value of$a(i)$, the better is $i$ assigned to its cluster.

$b(i)$ is the smallest average distance of $i$ to all points in the other cluster (of which $i$ doesn't belong to). $b(i)$ is a measure of how dissimilar $i$ is to its neighboring cluster. A large value means that $i$ is badly matched to its neighboring cluster.

$$b(i) = \min_{i \neq j} \frac{1}{|C_j|} \sum_{j \in C_j} d(i,j) \quad (7)$$

Fig. 3 illustrates the average Silhouette values of all data points according to the number of clusters, from $k = 2$ to $k = 6$.

The average Silhouette value is optimized for $k = 2$. Thus, the partitioning of our dataset into $k = 2$ sub-groups seems to be the best natural way to cluster the data, minimizing the risk of cluster overlapping and assignment errors.

In order to perform k-means clustering, two initial centroids are then randomly selected that correspond to the number of clusters desired. Each data point is allocated to its nearest mean based on the Euclidian distance between each point and the two means. Two initial clusters are then formed. After, the centroids of each of the two clusters become the new means. These allocating and updating steps continue until the in-cluster sum of squares is minimized [25].

The k-means algorithm is known to have a time complexity of $O(n^2)$, where $n$ is the input data size [26].

If all features are used separately as input of the k-means algorithm,

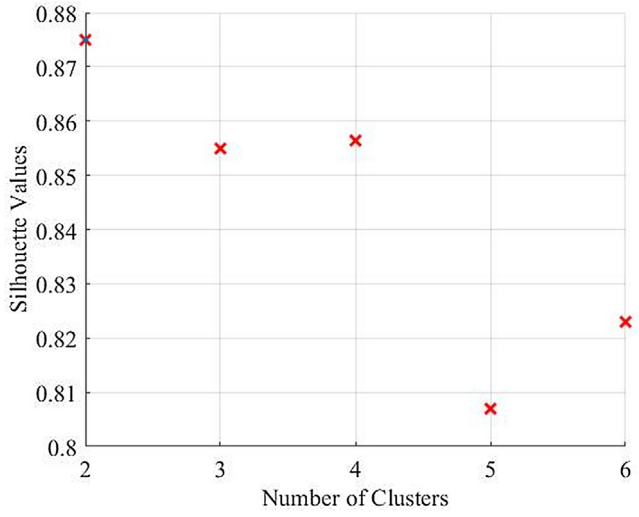


**Fig. 3.** Average Silhouette values of all data points according to the number of cluster.

**Table 1**
Subsets of features that give the best clustering quality and corresponding Dunn index values.

| Nomenclature | Subset of features | Selected subsets that maximize the Dunn index value |
|---|---|---|
| Feature n°4 = kurt(cD1) | All features | Feature n°4 |
| Features n°7 = skew(cD2) | (separately) | D = 0.1147 |
| Feature n°8 = kurt(cD2) | All couples | Features (n°4, n°7) |
| Feature n°15 = skew(cD4) | | D = 0.1170 |
| Feature n°19 = skew(cD5) | All triplets | Features (n°4, n°8, n°19) |
| Feature n°20 = kurt(cD5) | | D = 0.1287 |
| | All quadruplets | Features (n°4, n°8, n°15, n°19) |
| | | D = 0.1361 |

the complexity becomes $O(n_v n^2)$ with $n_v$ the number of features. For each couple, the complexity is $O(n_c n^2)$ with $n_c$ the number of couples. The same reasoning can be applied with each triplets and quadruplets.

### 3.3. k-means clustering results

Features that give the best clustering quality according to the maximization of the Dunn index value are summarized in Table 1.

The search for the best subset of features in unsupervised learning leads to a new problem: that the number of clusters, $k$, depends on the feature subset. Using a fixed number of clusters for all feature sets does not model the data in the respective subspace correctly. To be sure that the optimal number of cluster was still $k = 2$ for all feature subsets selected in Table 1, the average Silhouette values of all data points is computed for the respective subsets of features for different number of clusters. Fig. 4 shows that the optimal number of cluster for all feature subsets is still $k = 2$.

Feature n°4 maximizes the Dunn index value when all features are used separately as input of k-means algorithm. Moreover, feature n°4 appears in all selected subsets shown in Table 1. Thus, it means that this feature permits to obtain the best clustering quality. When this feature is combined with others, the clustering quality is slightly improved (the value of the Dunn index increases).

Fig. 5 illustrates the resulting clusters using couple (n°4, n°7) as input of k-means algorithm. Using pairs of features enables to better visualize the grouping of data into the two resulting clusters. As can be

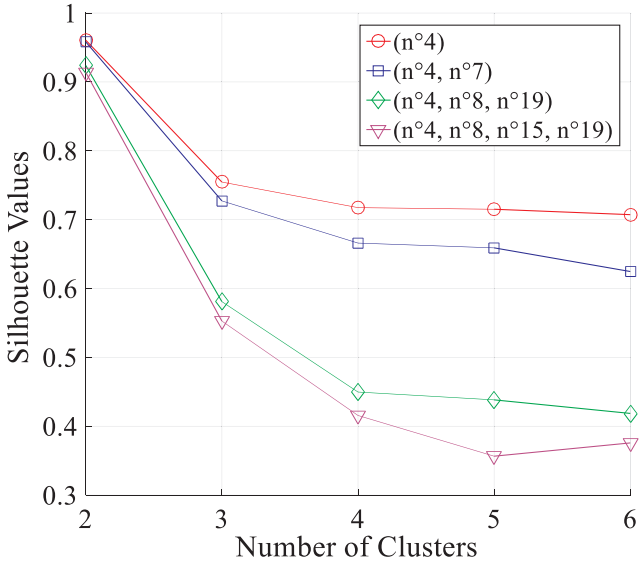


**Fig. 4.** Average Silhouette values of all data points according to the number of cluster for the selected subsets of features.

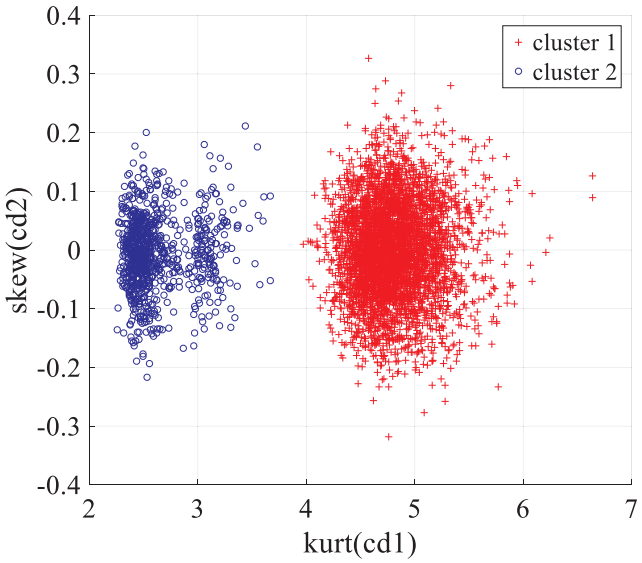


**Fig. 5.** Resulting clusters using feature n°4 and feature n°7 as input variables.

**Table 2**
Features that decrease the Dunn index values when combined with feature n°4.

| Nomenclature | Subsets of features and corresponding Dunn index value |
|---|---|
| Feature n°4 = kurt(cD1)<br>Features n°12 = kurt(cD3)<br>Feature n°16 = kurt(cD4)<br>Feature n°20 = kurt(cD5) | Features (n°4, n°12)<br>D = 0.00048<br>Features (n°4, n°16)<br>D = 0.0036<br>Features (n°4, n°20)<br>D = 0.0033 |

observed, data points are well matched to their own cluster and badly matched to the other. However, the Dunn index decreases if variable n°4 is combined with variable n°12, n°16 or n°20 (Table 2).

For example, the couple of features (n°4, n°20) gives a Dunn value of 0.0033. In Fig. 6, the clusters obtained using couple (n°4, n°20) as input of k-means algorithm are badly separated and overlapped.

In order to further visualize the data configuration into the two clusters using different subset of features, we plotted the corresponding Silhouette graph for feature pairs (n°4, n°7) and (n°4, n°20) in Figs. 7

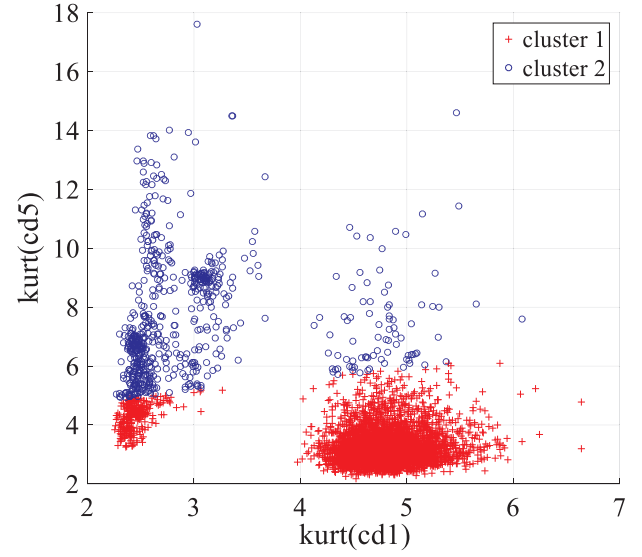


**Fig. 6.** Resulting clusters using feature n°4 and feature n°20 as input variables.

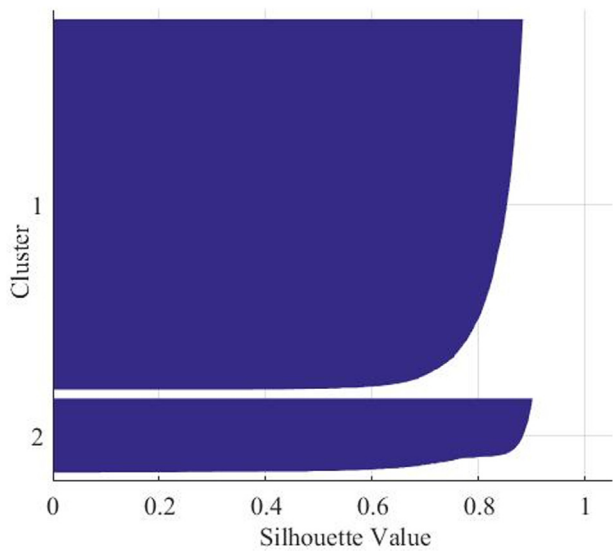


**Fig. 7.** Silhouette plot of data points using couple of features n°4 and n°7 as input of k-means algorithm.

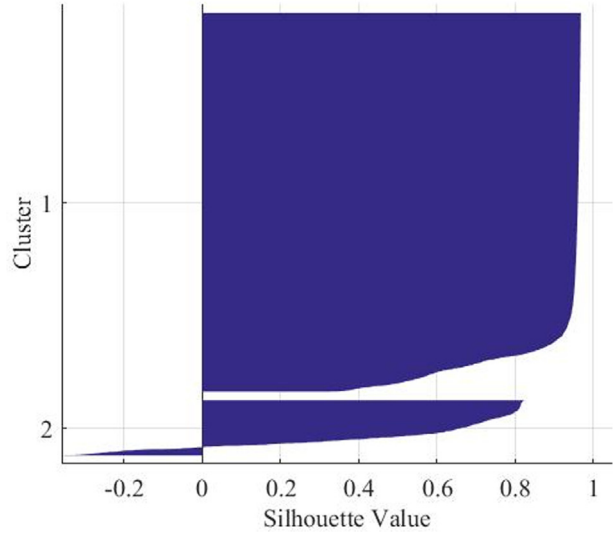


**Fig. 8.** Silhouette plot of data points using couple of features n°4 and n°20 as input of k-means algorithm.

and 8 respectively. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring cluster. The silhouette value of each data points are represented on the x-axis of the plots, for both clusters. Silhouette coefficients near + 1 indicate that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters and negative values indicate that those samples might have been assigned to the wrong cluster. Also, the cluster size can be visualized from the thickness of the silhouette plot. Cluster 1 is larger than cluster 2, because the first cluster contain more objects than the second one. The thickness of the Silhouette plot represents the size of the resulting clusters. In the first case (Figure n°7), the Silhouette coefficients of data points are near + 1 for both clusters, which means that they were classified with the least amount of doubt. Samples belonging to one cluster are far away from the neighboring cluster. On the contrary, the clustering results obtained when feature n°20 is paired to feature n°4 show that some samples have a negative Silhouette value (Fig. 8). These samples might have been assigned to the wrong cluster by the k-means algorithm. The clustering quality is decreased.

By integrating cluster validation metrics in our framework, we investigated two key challenges in unsupervised cluster analysis: the estimation of the number of clusters by using Silhouette value, and the issue of feature selection (Dunn index). As a result, we can automatically estimate the number of clusters and the best features subsets for PD-noise classification. However, this unsupervised framework does not provide any method for the validation of the results, thus it is not possible to assert if the signals, for example, clustered in red in Fig. 5 are PD or noise signals.

In the next section, we investigate the performance of a supervised learning framework in which a separated set of *labeled* data is available to test automatically the classification performance of the algorithm.

## 4. PD-noise discrimination using semi-supervised learning

Data labeling is expensive and time consuming. In most cases, data are unlabeled. For this reason, semi-supervised learning is interesting because only a small set of labeled data is required to help the algorithm determining the appropriate classifier. In addition, since a part of the labeled data is used to build a test set, then the classification performance can be evaluated automatically.

### 4.1. Transductive SVMs

Transductive Support-Vector Machines (TSVM) have been extensively used to process partially labeled data in semi-supervised learning [27]. Transductive SVMs is a kernel-based semi-supervised approach. It implements algorithms which search for the best separating hyperplane in the kernel space with a transductive process that includes both labeled and unlabeled samples in the training phase. Similarly to standard SVM, the best separating hyperplane is the one, which is as far as possible from the nearest training examples. The procedure is based on an iterative algorithm:

At the initial iteration, a standard SVM classification is used to obtain a first separating hyperplane based on the labeled data only. Samples are classified according to the sign of the SVM discriminant function:

$$f(x) = \sum_{i=1}^{M} \alpha_i y_i k(x, x_i) + b \tag{8}$$

where: $k$ is the kernel function. In this study, a linear kernel function is used.

$x_i$ are the support vectors, $y_i$ are the corresponding class labels ($\pm 1$) and $M$ is the number of support vectors. $\alpha_i$ and $b$ are the parameters of the classifier adjusted during the training process that leads to maximizing:

$$L(\alpha) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{M} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \tag{9}$$

Subject to $\sum_{i=1}^{M} \alpha_i y_i = 0$ and $0 \le \alpha_i \le C$, for $1 \le i \le M$

The hyperparameter $C$ controls the trade-off between classification errors on training data and margin maximization, thus regularization.

Following the first step, the resulting hyperplane (Eq. (8)) is used to assign pseudo labels to the unlabeled points in the training set which are called semi labeled data.

The second stage consists of an optimization problem where the hyperplane is forced to be as far as possible from the unlabeled data points. This is done by minimizing a cost function composed of a regularization and two error-penalization parameters. One parameter is used for the initial labeled examples, and the other for the semi-labeled examples (which were initially unlabeled, and for which labels were predicted). Permutations of labels that lead to a reduction of the cost function are implemented during the optimization process until no additional labels permutations are feasible [27,28].

The value of the regularization hyperparameter $C$ is estimated during the validation procedure, as in the case of standard SVMs [29]. It involves the partitioning of the labeled data into different subsets on which the generalization performance of the classifier can be estimated. The data partitioning is illustrated in Table 3. A test set is randomly built from labeled data. The remaining labeled data are split into two groups: a validation set composed of labeled data only and a training set that is mixed with all the unlabeled data. In our study, the cross-validation procedure is used for the selection of hyperparameter $C$. The remaining labeled data are divided into $K$ sets called folds. Only one fold is used for the validation and the classifier is trained on the training set composed of the $K - 1$ remaining folds and all the unlabeled data. The training and validation phases are repeated K times and the validation fold changes at each training [30].

**Table 3**
Data partitioning scheme for semi-supervised learning.

| All available data : 4993 | | |
|---|---|---|
| Labeled data: 100 | Unlabeled data : 4793 | Labeled data : 100 |
| **Validation set**: 20 labeled data (10 PD signals/10 noise signals) | **Training set:** 80 labeled data (40 PD signals/40 noise signals) + 4793 unlabeled data | **Test set:** 50 PD signals 50 noise signals |

The cross-validation procedure is iterated 10 times for each value of the hyperparameter with random shuffling of the labeled data into the folds in order to make the validation score independent from the data partitioning into the folds. At each iteration, the average validation score over the folds is computed. The validation score is the percentage of correctly classified examples on the validation set. The same procedure is repeated for different value of the hyperparameter, and hyperparameter that gives the best average validation scores over the 10 partitioning of the labeled data into the folds is selected. The best classifier is then trained with all examples of the training and validation sets and its performance is assessed on the test set, in order to estimate the classifier performance on examples that have never been used before. The TSVM used in our study was implemented using the SVMlight toolbox [31].

The entire procedure for hyperparameter and feature selection using TSVM algorithm with linear kernel is based on the following steps:

**Algorithm.**

**1.**
Normalize the dataset
Define a set of n hyper-parameters C = [C1……..Cn]
Define a value p = 10 random partitioning of the data into the folds
**2.**
—for i = 1 : NMax, with NMax, the number of available features
— First, consider each feature separately as input of the TSVM
— for j = 1 : n
— Consider hyperparameter Cj
— for hyperparameter Cj
— for l = 1 : p
— Draw one random partitioning of the data into the five folds
— For k = 1:5
Set fold k as the validation set
Train model on remaining k-1 folds
Compute and store validation score on fold k
—End for k
Compute and store average validation score over the 5 folds
— End for l
— Compute and store average cross-validation score over the 10 random partitioning of the data over the folds
— End for j
— Select and store hyperparameter Cj with best average cross-validation score over the 10 partitioning
— End for i
— Select feature with hyperparameter Cj that gives the best average cross-validation score over the 10 partitioning
**3.**

Perform step 2 with each couple of features
**4.**
Train a TSVM on the 4893 examples using the selected feature as input and the corresponding hyperparameter Cj
Compute and store the test score
**5.**
Repeat step 4 for the best couple of features
Compute and store the test score

The implementation of TSVM with linear kernel involves a time complexity of $O(U + L)$ where $L$ and $U$ are the numbers of labeled and unlabeled examples [32]. In the proposed approach, 5-folds cross-validation is performed 10 times for each value of hyperparameter C. This entire process is repeated 20 times, for each feature used as input of the TSVM algorithm. In this case, the training complexity of the method is $O(n_v \times h_p \times p \times k \times (U + L))$ with $k$ the number of folds, $p$ the number of random partitioning of the data into the folds, $h_p$ the number of hyperparameters and $n_v$ the number of features.

When all couples of features are used as input, the training complexity of the method is $O(n_c \times h_p \times p \times k \times (U + L))$ with $n_c$ the number of couples (190 in this case).

### 4.2. TSVM results

The data partitioning is implemented as indicated in Table 3, allocating 20 labeled examples to the validation set, 100 labeled examples to the test set and 4873 examples to the training set, 4793 of which are unlabeled. Thus, 4% of the total available data is labeled. All the labeled sets of data contain 50% of PD signals and 50% of noise signals. The test set is used to assess the performance of the classifier built using training and validation sets.

#### 4.2.1. Labeling of data

In this work, the labeling of the 200 data is assisted by the peak amplitude-charge cluster graph reported in [33]. In this cluster graph, the peak amplitude of the signal is represented in the ordinate axis. The PD current signal is approximated by dividing the HFCT's voltage output over the sensor gain. The discrete time integration of the main peak of this current signal is an estimation of the charge of the PD pulse [14]. The charge is represented in the abscissa axis of the cluster graph leading to the result of Fig. 9.

By using the software *PDflex* [34], it is possible to retrieve the waveform of the signals as the user hovers the pointer over the cluster graph and check visually if that given signal corresponds to a PD or noise signal. Due to the compactness of the test set-up shown in Fig. 1, the PD signals were characterized by an almost unipolar waveform, with some variations in the shape of the main peak. Conversely, noise signals had a very distinct oscillatory waveform.

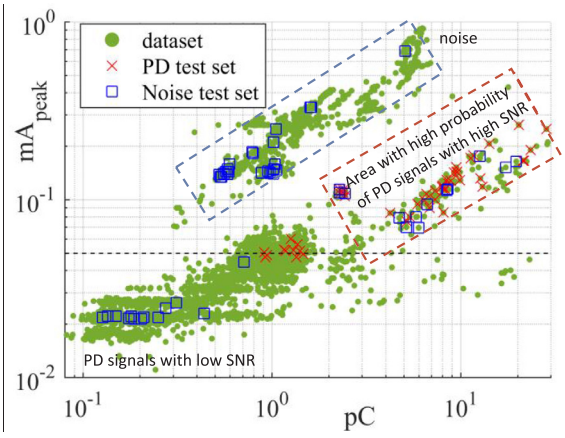


**Fig. 9.** Peak amplitude-Charge cluster graph assisting the labeling of data.

In Fig. 9, the signals in the dash blue square were labeled as noise signals. Signals in this group had a waveform like the one presented in Fig. 10a. On the other hand, the signals in the red dash square were labeled as PD signals with a high SNR. Examples of these are the waveforms shown in Fig. 10b-d. The remaining signals were of both types, even occurring very close to each other.

After the visual checking of the waveforms, the PD test set was defined as the signals with the "x" red marker, while the noise test set corresponded to the ones with the blue marker. In composing the test sets two criteria were added by the user. First, PD signals with a peak amplitude lower than 0.05 mA were labeled as noise signals even if their waveforms matched the ones of PD signals. An example of a signal not passing this criterion can be seen in Fig. 11e. The second criterion was to label as noise those PD signals with EMI disturbances overlapped as shown by Fig. 11f-h. The reason for these criteria is that in practice, no PD-related parameter can be accurately computed from signals with very low SNR or with EMI disturbances.

As a result of these criteria, the noise test set that is shown in Fig. 9 includes signals also occurring in the red dash square and below the 0.05 mA threshold.

#### 4.2.2. Classification results

TSVMs algorithm was implemented using the data partitioning of Table 3 using criteria presented in Section 4.2.1 for labeling the data (the validation and test sets). The performance of the classifier in use phase was evaluated with the labeled test set, also referred to as "real test labels". In addition, another classifier was implemented using different criteria for labeling the validation and test sets. It consists in labeling PD signals with low SNR or EMI (signals of type e, f, g, h in Fig. 11) as PD signals and not as noise signals. In this case, only signals of type a (in Fig. 10) were labeled as noise. The performance obtained was compared to that of the classifier built using labeling criteria presented in Section 4.2.1.

The classification scores of Table 4 correspond to the percentage of correctly classified signals in the test sets. This score is obtained by comparing the vector of "real test labels" with the vector of the predicted test labels by the classifier.

As with the k-means algorithm in Section 3, the TSVM algorithm was fed by a combinatorial of the 20 features. The best classification accuracy obtained by the TSVM algorithm reached 80% on the test set when the feature n°4 and the couple (n°4, n°7) were used as input, which further confirms the results of the unsupervised feature selection framework in Section 3 However, the classifier implemented using the second criteria for labeling the data achieves 100% of accuracy in the recognition of PD signals of the types b, c, d, e, f, g, h (in Figs. 10, 11) from noise signals of the type a (in Fig. 10). This score was also achieved using feature n°4 and the couple (n°4, n°7) as input.

The reason why feature n°4 is such a strong discriminant can be inferred from the comparison of the shape of the cD1 distribution for a PD and for noise signal. For instance, this comparison is shown in Fig. 12 for a representative signal of the type b and of the type a in Fig. 10. It is clear that the central peak of the distribution is sharper in the case of a PD signal, consequently the value of feature n°4 (kurtosis) for a PD signal is higher than for a noise signal.

The comparison of the real test labels (according to criteria defined in Section 4.2.1) and the predicted labels by the TSVM algorithm is shown in Fig. 13. It can be seen that the noise signals of type a in the blue circle in Fig. 13 were correctly labeled by the algorithm. On the other hand, 7 out of the 10 signals that were mislabeled as PD signals when they were labeled as noise by the user occurred within the blue square and correspond to PD signals with very low SNR of type e. The 3 remaining mislabeled signals correspond to PD signals of type f, g and h with high SNR but EMI disturbances. Otherwise, 6 out of the 10 signals that were mislabeled as noise signals when they were PD signals occurred within the blue dash square. The classification errors obtained can be interpreted as following: the labeling of data according to

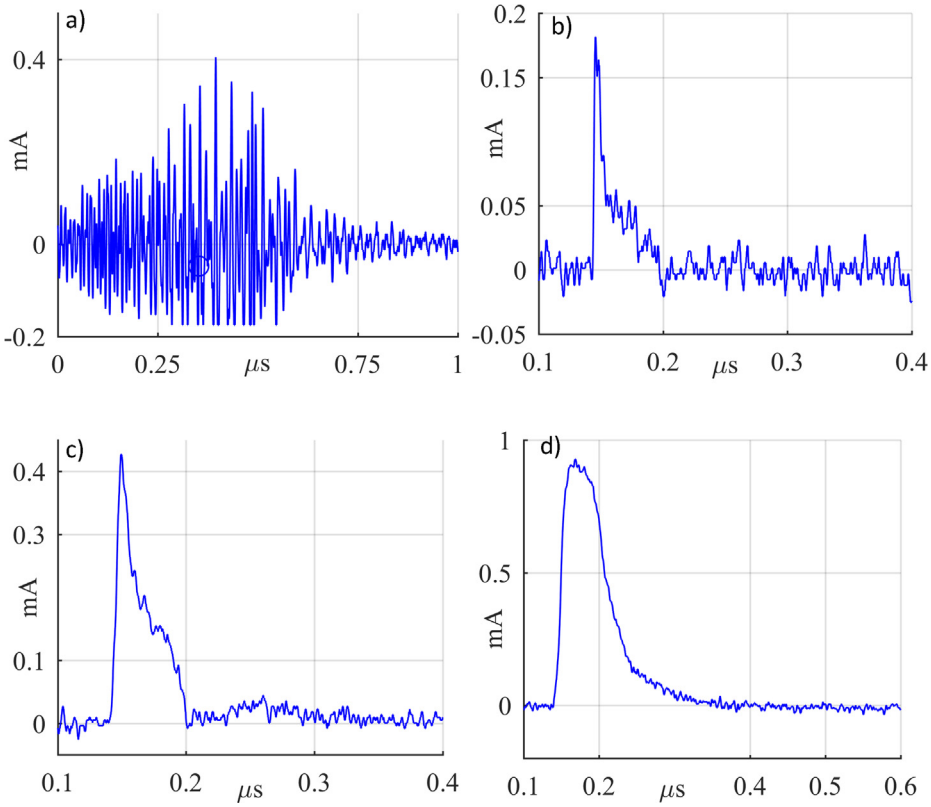


**Fig. 10.** Examples of waveforms from each of the clusters in Fig. 9.

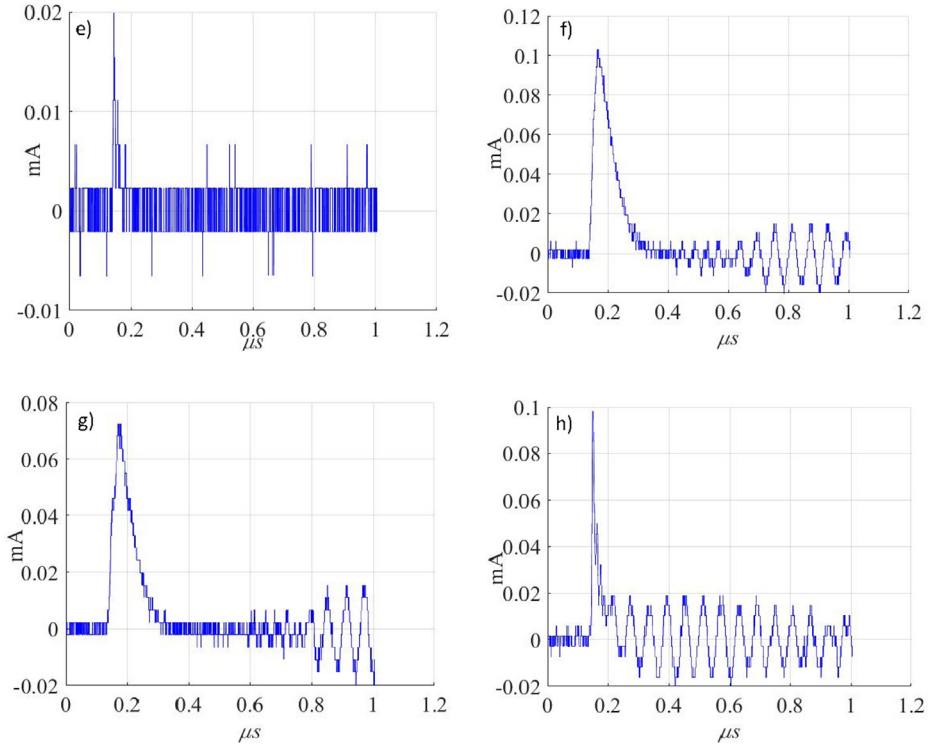


**Fig. 11.** Examples of PD signals with very low SNR and EMI labeled as noise signals.

criteria presented in 4.2.1 somehow misguide the classifier which recognizes a part of PD signals of type e, f, g, h as PD, whereas the user wants to classify them as noise. It also implies that some PD signals with high SNR and no EMI are recognized as noise by the classifier. On the contrary, if the user choose to label signals of type e, f, g and h as PD, the classifier is able to recognize with 100% accuracy PD signals of type b, c, d, e, f, g, h from noise signals of type a (Table 4).

A closer look at the shapes of the cD1 distribution of PD signals of type b, c, d and those PD signals of type f, g and h, labeled as noise by the user, suggested that the shapes of the distribution are very similar.

**Table 4**
Best classification accuracy results on the test set for linear TSVM

| Nomenclature | Classifier | Subset of features | Percentage of correctly classified test set examples using first criteria for labeling the data | Percentage of correctly classified test set examples using second criteria for labeling the data |
|---|---|---|---|---|
| Feature n°4 = kurt(cD1)<br>Feature n°7 = skew(cD2) | TSVM linear C = 10 | Feature n°4 | 80% | 100% |
| | TSVM linear C = 10 | Features (n°4, n°7) | 80% | 100% |

In Fig. 14, it can be noticed that the cD1 distribution for a PD signal of type f (Fig. 14b) looks much similar to the distribution of a PD signal of type b (Fig. 14a) than to that of a noise signal of type a (Fig. 12b).

This could explain why PD signals with EMI were classified as PD and why the global classification accuracy on the test set did not reach 100%. However, the 80% of accuracy on the entire test set remains a satisfactory result to classify PD and noise signals according to criteria defined in Section 4.2.1

## 5. Conclusion

In this work, unsupervised as well as semi-supervised classification methods were applied to the classification of PD and noise signals collected from a test cell under 40 kVDC. Experimental data were transformed using DWT and decomposed up to five levels. A set of 20 numerical features formed by the mean, variance, skewness and kurtosis of the wavelet detail coefficient distribution at each level of decomposition were extracted from each acquired signal.

A first unsupervised framework was proposed for feature selection and for the determination of the optimal number of clusters based on the Dunn index. The use of feature n°4, which is the kurtosis of the distribution of the detail coefficients at level one, as input of a k-means algorithm resulted in clearly well-separated clusters.

Since the unsupervised framework does not provide any method for the validation of the results, then a semi-supervised learning approach was applied on the same dataset using Transductive SVMs. 4% of the total dataset was labeled as PD or noise and this manual labeling process was assisted by checking the waveforms of the signals.

A fraction of this labeled dataset was then used for automatic testing of the classifier performance. In this test set, some PD signals with very low SNR and EMI were labeled as noise signals by the user

The results obtained using the semi-supervised approach showed a successful separation of PD and noise signals according to criteria defined in 4.2.1, with 80% of accuracy and a reduced set of features (feature n°4 alone or couple (n°4, n°7)), thus decreasing considerably the size of data to be processed as well as the computation time required. Moreover, it confirmed the feature selection results obtained in the unsupervised case. A part of the 20% of misclassified signals comes from PD signals labeled as noise by the user for post-treatment purpose. However, if those signals are labeled as PD by the user, 100% of classification accuracy is achieved. Thus, the performance of the method presented to classify PD and noise signals depends on the criteria defined by the user to label the validation and test sets.

The performance of the linear TSVM classifier implemented has demonstrated that semi-supervised learning is an interesting approach for the classification of PD and noise signals because it requires the user to label only a small amount of the total available data and permits an automatic testing of the classifier performance. Moreover, its implementation involves lower time complexity than that of the unsupervised approach.

This technique is a promising tool to improve the diagnostics of insulation of HV equipment under HVDC voltage, where the need to discard automatically noise signal with high accuracy is of great importance.

Finally, the perspective of transferring this classification methodology from one environment (e.g., one particular discharge configuration) to another would be of great interest. For this purpose, domain adaptation techniques [35] could be implemented in order to make the classifier able to separate noise from PD signals acquired in different discharge configurations.

## Credit authorship contribution statement

**N. Morette:** Conceptualization, Methodology, Software, Formal analysis, Writing - original draft, Writing - review & editing, Resources, Validation. **L.C. Castro Heredia:** Writing - review & editing, Resources, Validation. **Thierry Ditchi:** Validation, Resources, Writing - review & editing, Supervision. **A. Rodrigo Mor:** Writing - review & editing. **Y. Oussar:** Supervision, Project administration, Funding acquisition.

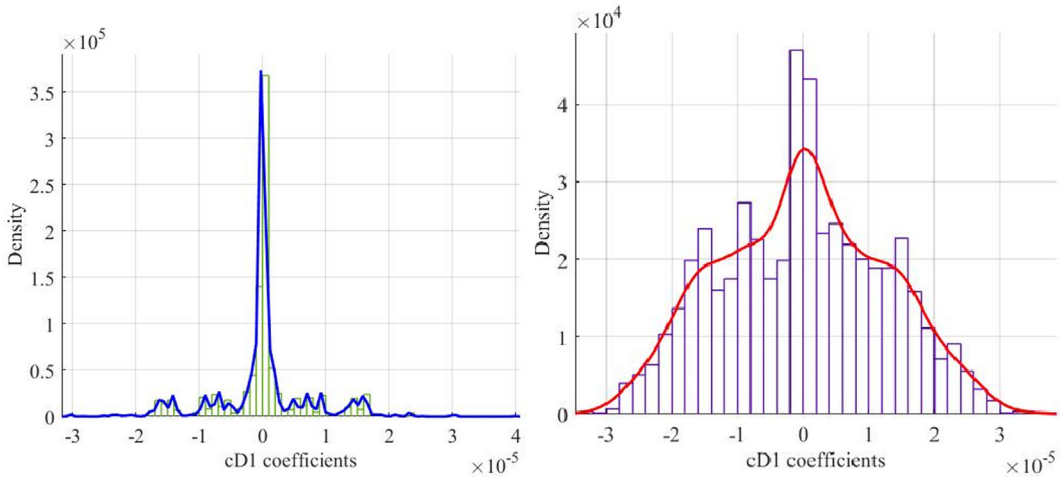


**Fig. 12.** Comparison of the shape of the distribution of cD1 for (a) a PD signal of type b, (b) a noise signal of type a.

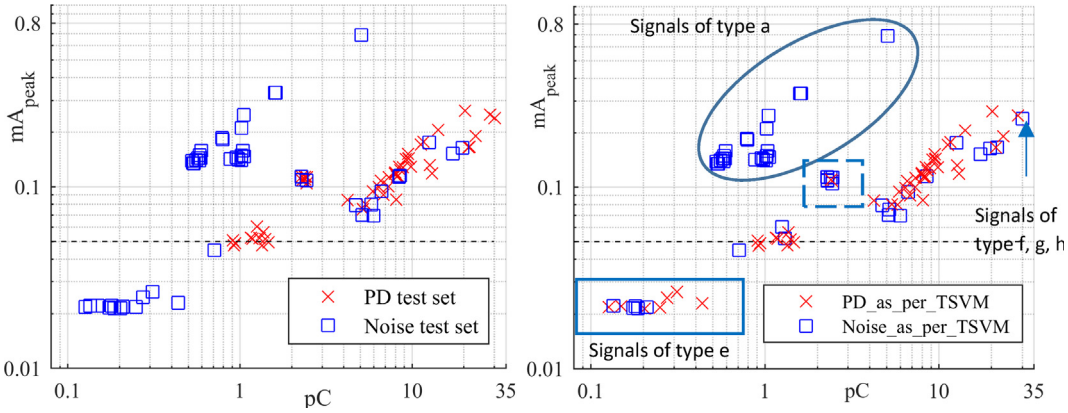


**Fig. 13.** Comparison of (a) the real test labels and (b) the predicted labels by the TSVM method.

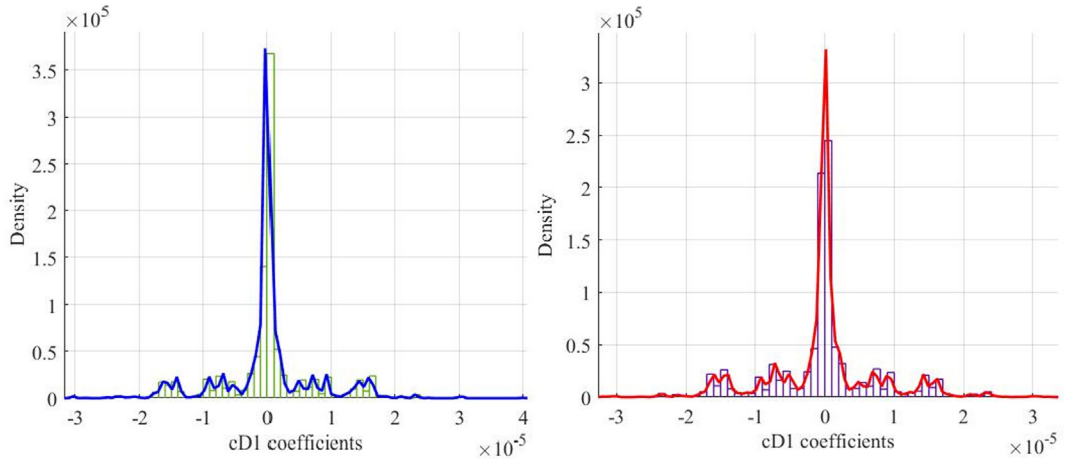


**Fig. 14.** Comparison of the shape of the distribution of cD1 for (a) a PD signal of type b, (b) a PD signal of type f (Fig. 11).

## Declaration of Competing Interest

None.

## References


[1] Morette N, Daassi-Gnaba H, Ditchi T, Oussar Y. Characterization of partial discharges in solid insulators under DC voltage using physical cavity properties. IEEE Int Sympos Electr Insul Mat 2017:374–7.

[2] Morette N, Ditchi T, Oussar Y. Partial discharges measurements and analysis as an evaluation tool for the reliability of polymeric-insulated cables used under HVDC Conditions. ICD 2018;1–4:8514708.

[3] Romano P, Presti G, Imburgia A, Candela R. A new approach to partial discharge detection under DC voltage. IEEE Electr Insul Mag 2018;34(4):32–41.

[4] Morris E, Siew W. A comparison of AC and DC partial discharge activity in polymeric cable insulation. In: 2017 IEEE 21st International conference on pulsed power (PPC)., Jun, 2017. IEEE; 2017.

[5] Morshuis PHF, Smit JJ. Partial discharges at dc voltage: their mechanism, detection and analysis. IEEE Trans Dielect Electr Insul 2005;12(2):328–40.

[6] Castro Heredia LC, Rodrigo Mor A. Density-based clustering methods for unsupervised separation of partial discharge sources. Int J Electr Power Energy Syst 2019;107:224–30. Elsevier BV.

[7] Katsuta G, Suzuki H, Eshima H, Endoh T. Discrimination of partial discharge from noise in XLPE cable lines using a neural network. In: Proceedings of the second international forum on applications of neural networks to power systems. IEEE; 1993.

[8] Ardila-Rey J, Martínez-Tarifa J, Robles G, Rojas-Moreno M. Partial discharge and noise separation by means of spectral-power clustering techniques. IEEE Trans Dielect Electr Insul 2013;20(4):1436–43.

[9] Alvarez F, Ortego J, Garnacho F, Sanchez-Uran MA. A clustering technique for partial discharge and noise sources identification in power cables by means of waveform parameters. IEEE Trans Dielect Electr Insul 2016;23(1):469–81.

[10] Aldrian R, Montanari GC, Cavallini A, Suwarno. Signal separation and identification of partial discharge in XLPE insulation under DC voltage. In: 2017 1st International conference on electrical materials and power equipment (ICEMPE)., May, 2017. IEEE; 2017.

[11] Hao L, Lewin PL, Hunter JA, Swaffield DJ, Contin A, Walton C, et al. Discrimination of multiple PD sources using wavelet decomposition and principal component analysis. IEEE Trans Dielectr Electr Insul 2011;18(5):1702–11.

[12] Joachim T. Transductive inference for text classification using support vector machines. In: International conference on machine learning (ICML); June 1999. p. 200–9.

[13] Jia J, Cai L. A TSVM-based minutiae matching approach for fingerprint verification. In: International workshop on biometric recognition systems (IWBRS); October 2005. p. 85–94.

[14] Rodrigo Mor A, Castro Heredia LC, Munoz FA. Estimation of charge, energy and polarity of noisy partial discharge pulses. IEEE Trans Dielectr Electr Insul 2017;24(4):2511–21.

[15] Rodrigo Mor A, Castro Heredia LC, Muñoz FA. A novel approach for partial discharge measurements on GIS Using HFCT Sensors. Sensors 2018;18:4482.

[16] Cunha CF, Carvalho AT, Petraglia MR, Lima AC. A new wavelet selection method for partial discharge denoising. Electr Power Syst Res 2015;125:184–95.

[17] Zhou X, Zhou C, Kemp IJ. An improved methodology for application of wavelet transform to partial discharge measurement denoising. IEEE Trans Dielect Electr Insul 2005;12(3):586–94. https://doi.org/10.1109/TDEI.2005.1453464.

[18] de Oliveira Mota Hilton, da Rocha LCD, de Moura Salles TC, Vasconcelos FH. Partial discharge signal denoising with spatially adaptive wavelet thresholding and support vector machines. Electr Power Syst Res 2011;81(2):644–59.

[19] Evagorou D, Lewin P, Efthymiou V, Kyprianou A, Georghiou G, Stavrou A, et al. Feature extraction of partial discharge signals using the wavelet packet transform and classification with a probabilistic neural network. IET Sci, Meas Technol, May 2010; 4(3): 177–92. Institution of Engineering and Technology (IET).

[20] Ma X, Zhou C, Kemp IJ. Interpretation of wavelet analysis and its application in partial discharge detection. IEEE Trans Dielect Electr Insul 2002;9(3):446–57.

[21] Adhikary JR, Murty MN. Feature selection for unsupervised learning. Neural Information Processing. Heidelberg: Springer, Berlin; 2012. p. 382–9.

[22] Liao R, Fernandess Y, Tavernier K, Irving MR. Recognition of partial discharge patterns. In: 2012 IEEE power and energy society general meeting., Jul, 2012. IEEE; 2012.

[23] Lin YH. Using K-means clustering and parameter weighting for partial-discharge noise suppression. IEEE Trans Power Deliv 2011;26(4):2380–90.

[24] Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 1987;20:53–65.

[25] Steinley D. K-means clustering: a half-century synthesis. Brit J Math Stat Psychol,

May, 2006; 59(1): . 1–34. Wiley.
[26] Pakhira MK. A linear time-complexity k-means algorithm using cluster shifting. In: 2014 international conference on computational intelligence and communication networks, Bhopal; 2014. p. 1047–51.
[27] Wang J, Shen X, Pan W. On transductive support vector machines. Contem Math 2007;443:7–19.
[28] Oussar Y, Ahriz I, Denby B, Dreyfus G. Indoor localization based on cellular telephony RSSI fingerprints containing very large numbers of carriers. J Wireless Com Network 2011;2011(1).
[29] Cristianini N, Shawe-Taylor J. Support vector machines and other kernel-based learning methods. Cambridge: Cambridge University Press; 2006.
[30] Morette N, Ditchi T, Oussar Y. Feature extraction and ageing state recognition using partial discharges in cables under HVDC. Electr Power Syst Res 2020.
[31] < http://svmlight.joachims.org/ > .
[32] Collobert R, Sinz F, Weston J, Bottou L. Large scale transductive SVMs. J Mach Learn Res 2006;7:1687–712.
[33] Rodrigo Mor A, Castro Heredia LC, Munoz FA. New clustering techniques based on current peak value, charge and energy calculations for separation of partial discharge sources. IEEE Trans Dielectr Electr Insulat. 24. Institute of Electrical and Electronics Engineers (IEEE); feb, 2017. p. 340–8.
[34] PDflex – Signal processing tool. Available at: < http://pdflex.ewi.tudelft.nl > (accessed March 18th 2020).
[35] Glorot X, Bordes A, Bengio Y. Domain adaptation for large-scale sentiment classification: a deep learning approach. In: Proceedings of the 28th international conference on international conference on machine learning. USA; 2011. p. 513–20.