Electric Power Systems Research 140 (2016) 413–423

Contents lists available at ScienceDirect

Electric Power Systems Research

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

Unsupervised non-technical losses identification through optimum-path forest

Leandro Aparecido Passos Júnior[a], Caio César Oba Ramos[c,*], Douglas Rodrigues[a], Danillo Roberto Pereira[b], André Nunes de Souza[c], Kelton Augusto Pontara da Costa[b], João Paulo Papa[b]

[a] Department of Computing, Federal University of São Carlos, São Carlos, Brazil
[b] Department of Computing, São Paulo State University, Bauru, Brazil
[c] Department of Electrical Engineering, São Paulo State University, Bauru, Brazil

ARTICLE INFO

Article history:
Received 7 January 2016
Received in revised form 24 May 2016
Accepted 31 May 2016
Available online 28 June 2016

Keywords:
Non-technical losses
Optimum-path forest
Clustering
Anomaly detection

ABSTRACT

Non-technical losses (NTL) identification has been paramount in the last years. However, it is not straightforward to obtain labelled datasets to perform a supervised NTL recognition task. In this paper, the optimum-path forest (OPF) clustering algorithm has been employed to identify irregular and regular profiles of commercial and industrial consumers obtained from a Brazilian electrical power company. Additionally, a model for the problem of NTL recognition as an anomaly detection task has been proposed when there are little or no information about irregular consumers. For such purpose, two new approaches based on the OPF framework have been introduced and compared against the well-known *k*-means, Gaussian mixture model, Birch, affinity propagation and one-class support vector machines. The experimental results have shown the robustness of OPF for both unsupervised NTL recognition and anomaly detection problems. In short, the main contributions of this paper are fourfold: (i) to employ unsupervised OPF for non-technical losses detection, (ii) to model the problem of NTL as being an anomaly detection task, (iii) to employ unsupervised OPF to estimate the parameters of the Gaussian distributions, and (iv) to present an anomaly detection approach based on unsupervised optimum-path forest.

1. Introduction

Non-technical losses (NTL) identification has been paramount in the last years, mainly in development countries where the energy taxes are based on the amount of thefts in the distribution system. The main concept related to NTL refers to the amount of energy that has been used but not billed, and may also reflect on the quality of distributed energy. Aiming to avoid thefts in power distribution systems, or even cyber-attacks [1], one can refer to a considerable collection of works that deal with NTL identification using machine learning techniques.

In regard to supervised-based NTL identification, Nagi et al. [2], for instance, proposed a hybrid approach composed of a support vector machines (SVMs) classifier and fuzzy models that outperformed their previous work, which employed only SVMs for this task [3]. Later, the same group of authors presented an interesting work comparing several supervised learning techniques for NTL detection [4]. Ramos et al. [5] introduced the supervised optimum-path forest (OPF) classifier in the context of theft detection in Brazilian consumers, obtaining results comparable to the ones achieved by SVMs, but much faster considering the training and testing phases. Further, Ramos et al. [6,7] presented two feature selection techniques for NTL characterization, i.e., the authors were concerned into finding the set of features that best discriminate legal and illegal profiles.

Since NTL-oriented datasets are usually imbalanced, Martino et al. [8] presented an approach to deal with such shortcoming, given most part of datasets usually contain much less positive (theft) samples than regular consumers. León et al. [9] presented an expert system for supervised NTL identification in Spain, and the aforementioned group of authors have tackled the same problem through regression analysis [10]. In addition, the MIDAS system was proposed by Monedero et al. [11] aiming to combat thefts in electrical power distribution systems based on neural networks and statistical analysis. Recently, Guerrero et al. [12] used a knowledge-based system that employs text mining,

* Corresponding author at: Av. Eng. Luiz Edmundo C. Coube, 14-01, 17033-360 Bauru, SP, Brazil. Tel.: +55 14 31036115.
E-mail addresses: leandropassosjr@gmail.com (L.A. Passos Júnior), papa@fc.unesp.br (J.P. Papa).

http://dx.doi.org/10.1016/j.epsr.2016.05.036
0378-7796/

neural networks, and statistical techniques to detect non-technical losses.

However, most part of works address the problem of NTL identification using supervised techniques. Since to obtain labelled datasets in this context is usually a hard task, it is often necessary to evaluate unsupervised techniques for NTL identification. Donadel et al. [13], for instance, presented a methodology to estimate the energy consumption based on clustering and sampling techniques, and Tasić and Stojanović [14] employed a fuzzy-based clustering algorithm for energy losses identification. In India, Babu et al. [15] aimed at identifying NTL and suspect profiles of irregular energy consumption by determining similar groups through fuzzy *c*-means clustering.

Recently, Rocha et al. [16] proposed an unsupervised version of the OPF classifier, in which the pattern recognition task is modelled as a graph partition problem where some key samples (prototypes) compete among themselves in order to partition the graph into clusters. Such approach has been used in several applications, but it has been noticed only one work that employed OPF for unsupervised NTL identification so far [17]. Therefore, this work aims at evaluating OPF clustering for non-technical losses identification in power distribution systems in two private datasets provided by a Brazilian electrical power company, being one dataset composed of commercial profiles and the another one composed of industrial consumers.

Another main contribution of this paper is to model the problem of non-technical losses identification as an anomaly detection task, in which the classifier is trained with the regular consumers only. Therefore, when a new sample comes up to be classified, it is identified if this sample belongs to the "normal" pattern learned by the classifier. Otherwise, such sample is then classified as being an anomaly, i.e., a profile from a thief user. One of the most commonly used approaches to detect anomalies is the so-called Multivariate Gaussian Distribution, in which each cluster is modelled as a Gaussian distribution with its parameters estimated by some machine learning technique. After that, when a new test sample appears to be classified, it is verified its closest Gaussian distribution, and if this distance is greater than a threshold, such sample is then classified as an anomaly. In this paper, the unsupervised OPF has been introduced to estimate the Gaussian parameters, showing it can be more robust than some techniques that are often used for such purpose. Additionally, OPF is compared against the well-known *k*-means, Birch, affinity propagation (AP), and Gaussian mixture model (GMM) considering both approaches, i.e., unsupervised NTL recognition and anomaly detection (in regard to this last task, one-class SVMs [18] has also evaluated either; and AP and Birch have been applied to unsupervised NTL identification only). Finally, an anomaly detection approach purely based on the optimum-path forest has been proposed as well, which was evaluated in the context of NTL identification.

In short, the main contributions of this paper are fourfold: (i) to employ unsupervised OPF for non-technical losses detection, (ii) to model the problem of NTL as being an anomaly detection task, (iii) to employ unsupervised OPF to estimate the parameters of the Gaussian distributions, and (iv) to present an anomaly detection approach based on unsupervised optimum-path forest. The remainder of this paper is organized as follows: Sections 2 and 3 present the OPF clustering theory and the experimental methodology, respectively. Section 4 discusses the experiments, and Section 5 states conclusions and future works.

2. Optimum-path forest clustering

Let $\mathcal{Z}$ be a dataset such that for every sample $s \in \mathcal{Z}$ there exists a feature vector $\vec{v}(s)$. Let $d(s, t)$ be the distance between s and t in the feature space. For instance, $d(s,t) = \|\vec{v}(t) - \vec{v}(s)\|$ – the Euclidean distance between $\vec{v}(t)$ and $\vec{v}(s)$. A graph $(\mathcal{Z}, \mathcal{A}_k)$ can be defined such that the arcs $(s, t) \in \mathcal{A}$ connect k-nearest neighbours in the feature space. The arcs are weighted by $d(s, t)$ and the nodes $s \in \mathcal{Z}$ are weighted by a probability density value $\rho(s)$:

$$\rho(s) = \frac{1}{\sqrt{2\pi\sigma^2}|\mathcal{A}_k(s)|} \sum_{\forall t \in \mathcal{A}_k(s)} \exp\left(\frac{-d^2(s,t)}{2\sigma^2}\right), \tag{1}$$

where $|\mathcal{A}_k(s)| = k$, $\sigma = d_f/3$, and d_f is the maximum arc weight in $(\mathcal{Z}, \mathcal{A}_k)$. This parameter choice considers all adjacent nodes for density computation, since a Gaussian function covers most samples within $d(s, t) \in [0, 3\sigma]$. Moreover, since $\mathcal{A}_k$ is asymmetric, symmetric arcs must be added to it on the plateaus of the probability density function (pdf) in order to guarantee a single root per maximum.

The traditional method to estimate a pdf is by Parzen-window. Eq. (1) can provide the Parzen-window estimation based on an isotropic Gaussian kernel when the arcs are defined by $(s, t) \in \mathcal{A}_k$ if $d(s, t) \leq d_f$. However, this choice presents problems with the differences in scale and sample concentration. Solutions for this problem lead to adaptive choices of d_f depending on the region of the feature space. By taking into account the k-nearest neighbours, the method handles different concentrations and reduces the scale problem to the one of finding the best value of k, say k^* within $[k_{\min}, k_{\max}]$, for $1 \leq k_{\min} < k_{\max} \leq |\mathcal{Z}|$.

The solution proposed by Rocha et al. [16] to find k^* considers the minimum graph cut among all clustering results for $k \in [1, k_{\max}]$ ($k_{\min} = 1$), according to the normalized measure $GC(\mathcal{A}_k, L, d)$ suggested by Shi and Malik [19]:

$$GC(\mathcal{A}_k, L, d) = \sum_{i=1}^{c} \frac{W'_i}{W_i + W'_i}, \tag{2}$$

$$W_i = \sum_{\forall (s,t) \in \mathcal{A}_k | L(s)=L(t)=i} \frac{1}{d(s,t)}, \tag{3}$$

$$W'_i = \sum_{\forall (s,t) \in \mathcal{A}_k | L(s)=i, L(t) \neq i} \frac{1}{d(s,t)}, \tag{4}$$

where $L(t)$ is the label of sample t, W'_i uses all arc weights between cluster i and other clusters, and W_i uses all arc weights within cluster $i = 1, 2, \ldots, c$.

The method defines a path π_t as a sequence of adjacent samples starting from a root $R(t)$ and ending at a sample t, being $\pi_t = \langle t \rangle$ a trivial path and $\pi_s \cdot \langle s, t \rangle$ the concatenation of π_s and arc (s, t). It assigns to each path π_t a value $f(\pi_t)$ given by a connectivity function f. A path π_t is considered optimum if $f(\pi_t) \geq f(\tau_t)$ for any other path τ_t.

Among all possible paths π_t from the maxima of the pdf, the method assigns to t a path whose minimum density value along it is maximum. That is, the method finds $V(t) = \max_{\forall \pi_t \in (\mathcal{Z}, \mathcal{A}_k)} \{f(\pi_t)\}$ for $f(\pi_t)$ defined by:

$$f(\langle t \rangle) = \begin{cases} \rho(t) & \text{if } t \in \mathcal{R} \\ \rho(t) - \delta & \text{otherwise,} \end{cases}$$
$$f(\langle \pi_s \cdot \langle s, t \rangle \rangle) = \min\{f(\pi_s), \rho(t)\}, \tag{5}$$

for $\delta = \min_{\forall (s,t) \in \mathcal{A}_k | \rho(t) \neq \rho(s)} |\rho(t) - \rho(s)|$ and $\mathcal{R}$ being a root set, discovered on-the-fly, with one element per each maximum of the pdf. It should be noted that higher values of δ reduce the number of maxima. In this work, we used $\delta = 1.0$ and $\rho(t) \in [1, 1000]$. The OPF algorithm maximizes the connectivity map $V(t)$ by computing an optimum-path forest – a predecessor map P with no cycles that assigns to each sample $t \notin \mathcal{R}$ its predecessor $P(t)$ in the optimum path from $\mathcal{R}$ or a marker *nil* when $t \in \mathcal{R}$. Algorithm 1 implements this procedure.

Algorithm 1. OPF clustering algorithm.

INPUT: Graph $(\mathcal{Z}, \mathcal{A}_k)$ and distance function d.
OUTPUT: Optimum-path forest P, connectivity map V and label map L.
AUXILIARY: Priority queue Q, density map ρ, variables tmp and $l \leftarrow 1$.

1. *For each* $s \in \mathcal{Z}$, *do*
2. *Compute* $\rho(s)$ *using Equation (1).*
3. *Set* $P(s) \leftarrow nil$, $V(s) \leftarrow \rho(s) - \delta$, *and insert* s *in* Q.
4. *While* Q *is not empty, do*
5. *Remove from* Q *a sample* s *such that* $V(s)$ *is maximum.*
6. *If* $P(s) = nil$, *then*
7. *Set* $L(s) \leftarrow l$, $l \leftarrow l + 1$, *and* $V(s) \leftarrow \rho(s)$.
8. *For each* $t \in \mathcal{A}_k(s)$ *such that* $V(t) < V(s)$, *do*
9. *Compute* $tmp \leftarrow \min\{V(s), \rho(t)\}$.
10. *If* $tmp > V(t)$ *then*
11. *Set* $L(t) \leftarrow L(s)$, $P(t) \leftarrow s$, $V(t) \leftarrow tmp$.
12. *Update position of* t *in* Q.
13. *Return a classifier* $[P, V, L]$.

In Algorithm 1, Lines 1–3 initialize the variables, and also inserts all samples in the priority queue Q. The main loop in Lines 4–13 is responsible to run the OPF clustering algorithm. It first removes a sample s from Q with maximum connectivity value $V(s)$. If s has not been conquered by any other sample, then $P(s) = nil$ (Line 6) and s is a root of the connectivity map (a maximum of the pdf). Since $s \in \mathcal{R}$, by Eq. (5), its connectivity value is reset to $\rho(s)$ (Line 7), which in addition to the fact that $\mathcal{A}_k$ is symmetric on plateaus of the pdf will make root s to conquer the remaining samples of its plateau. It is also assigned to it a new distinct label (cluster) for optimum-path propagation to the rest of its dome. The inner loop in Lines 8–12 evaluates all adjacent sample t of s to which s can offer a better connectivity value (i.e., $V(t) < V(s)$). If the path $\pi_s \cdot \langle s, t\rangle$ offers a higher connectivity value to t (Lines 9–10), then the current path π_t is substituted by the new path $\pi_s \cdot \langle s, t\rangle$, being the maps $V(t)$, $L(t)$, and $P(t)$ updated accordingly (Lines 11–12).

3. Methodology

In this section, the datasets and a brief explanation about the techniques employed in this paper are presented, as well as and the quality measure used to evaluate the effectiveness of the clustering techniques.

3.1. Datasets

Two labelled datasets obtained from a Brazilian electric power company, named B_i and B_c, have been used. The former is a dataset composed of 3178 industrial profiles, and the latter contains 4948 commercial profiles. Each industrial and commercial profile is represented by eight features, as follows:

1. Demand Billed (DB): demand value of the active power considered for billing purposes, in kilowatts (kW);
2. Demand Contracted (DC): the value of the demand for continuous availability requested from the energy company, which must be paid whether the electric power is used by the consumer or not, in kilowatts (kW);
3. Demand Measured or Maximum Demand (D_{max}): the maximum actual demand for active power, verified by measurement at 15-minute intervals during the billing period, in kilowatts (kW);
4. Reactive Energy (RE): energy that flows through the electric and magnetic fields of an AC system, in kilovolt-amperes reactive hours (kVArh);
5. Power Transformer (PT): the power transformer installed for the consumers, in kilovolt-amperes (kVA);
6. Power Factor (PF): the ratio between the consumed active and apparent power in a circuit. The PF indicates the efficiency of a power distribution system;
7. Installed Power (P_{inst}): the sum of the nominal power of all electrical equipment installed and ready to operate at the consumer unit, in kilowatts (kW);
8. Load Factor (LF): the ratio between the average demand ($D_{average}$) and maximum demand (D_{max}) of the consumer unit. The *LF* is an index that shows how the electric power is used in a rational way.

The commercial dataset contains 4680 regular consumers (94.58%) and 268 irregular profiles, while the industrial dataset contains 2984 samples (93.89%) that represent regular consumers, and 194 samples that denote irregular consumers. Notice all the aforementioned features are measured over one month.

3.2. Clustering techniques

In this section, the unsupervised techniques employed for comparison purposes against OPF are presented, as well as the modelling used for both unsupervised NTL recognition and anomaly detection tasks.

3.2.1. k-Means

Such technique requires the number of clusters as an input, and it groups data trying to separate samples in groups with similar variance in order to minimize some criterion. This algorithm scales well to large number of samples, and it has been widely used in many different application areas [20–22].

Roughly speaking, the k-means algorithm works by dividing a set of N samples into k disjoint clusters C_j, $j = 1, 2, \ldots, k$, being each one described by the mean μ_j of its samples. The means are the so-called cluster "centroids", and may not be points from the dataset. The k-means algorithm aims at choosing centroids that minimize the within-cluster sum of squared criterion [22], given by:

$$\sum_{i=0}^{N} \min_{\mu_j \in C_j}(\left\|x_i - \mu_j\right\|^2), \quad \forall \ j = 1, 2, \ldots, k, \tag{6}$$

being x_i a sample from the training data.

3.2.2. GMM

A Gaussian mixture model is a probabilistic model that considers all the data points are created from a mixture of a finite number of

Gaussian distributions with unknown parameters. It is possible to imagine such approach as a generalization of k-means clustering by adding the information about the covariance structure of the data, as well as the centres of the latent Gaussians [22].

The expectation–maximization (E–M) algorithm is usually implemented by GMM in order to fine-tune each Gaussian mixture, which basically aims at learning the mean and covariance of each model, as well as its weight to be used in the mixture computation. Finally, test samples are classified by associating them to the most likely Gaussian distribution. Since it might be difficult to learn Gaussian mixture models from unlabeled data, E–M tries to circumvent this problem by an iterative process that assumes initial random components and, for each point, it estimates the probability of being part of each component of the model. Then, one takes the parameters to maximize the probability of the data given those assignments. This process is repeated until it met some convergence criterion [22].[1]

3.2.3. Affinity propagation

Affinity propagation is an interesting clustering technique that does not require the number of clusters beforehand [25]. The main idea is to find the most representative exemplar for each dataset sample based on an iterative process. The algorithm keeps updating two matrices called "responsibility" and "availability", being the former in charge of modelling the level of suitability of each sample to be the most representative one concerning another sample, and the latter matrix encodes how appropriate it would be for given sample to pick another one as its exemplar instead of others.

3.2.4. Birch

Birch is a hierarchical clustering technique often used to handle data stream, since it is considerably fast [26]. Also, it is based on the assumption that all samples may not be equally important for clustering computation. Given that Birch is incremental, it does not need all data in advance.

3.2.5. SVM

Considering the task of anomaly detection, one-class support vector machines have been employed as well [18], which has a slightly different formulation from that used to describe SVMs. Instead of trying to learn a hyperplane that maximizes the distance between elements from different classes, one-class SVM build a hyperplane that maximizes the distance between the sample in the feature space from the origin of that coordinate system. Therefore, a test sample near the other ones will be labelled as +1, and the ones far apart will be labelled as -1 (anomaly).

3.3. Evaluation analysis

In this work, two distinct experiments are conducted: (i) the first is related to the unsupervised NTL identification (Section 4.1), and (ii) the second one is related to the application of unsupervised techniques to model the problem of NTL identification as an anomaly detection task (Section 4.2). Although the label information is used to design the datasets, such knowledge is not employed during the learning process. In regard to the anomaly detection task, a new dataset composed of legal consumer profiles has been built. Such data are considered "normal" samples, and right after the unsupervised learning process, any sample that does not fit into the learned model, is thus considered an "anomaly".

Considering both experiments, a classification accuracy proposed by Papa et al. [27] that considers unbalanced datasets has been adopted, which is a common problem in the context of NTL detection. Such classification accuracy strongly penalizes errors on small classes, being adequate to the context of this work, since there are much more samples from regular consumers than irregular ones. The accuracy is measured by taking into account the classes may have different sizes in a dataset $\mathcal{Z}$. Let us define:

$$e_{i,1} = \frac{FP_i}{|\mathcal{Z}| - |\mathcal{Z}^i|} \tag{7}$$

and

$$e_{i,2} = \frac{FN_i}{|\mathcal{Z}^i|}, \quad i = 1, 2, \ldots, K, \tag{8}$$

where Z stands for the number of classes, $|\mathcal{Z}^i|$ concerns with the number of samples in $\mathcal{Z}$ that come from class i, and FP_i and FN_i stand for the false positives and false negatives for class i, respectively. That is, FP_i is the number of samples from other classes that were classified as being from the class i in $\mathcal{Z}$, and FN_i is the number of samples from the class i that were incorrectly classified as being from other classes in $\mathcal{Z}$. The error terms $e_{i,1}$ and $e_{i,2}$ are then used to define the total error from class i:

$$E_i = e_{i,1} + e_{i,2}. \tag{9}$$

Finally, the accuracy Acc is then defined as follows:

$$Acc = 1 - \frac{\sum_{i=1}^{K} E_i}{2K}. \tag{10}$$

Additionally to the aforementioned accuracy measure, the F-measure is employed either [28], a well-known measure that considers both the *precision* and *recall* information.

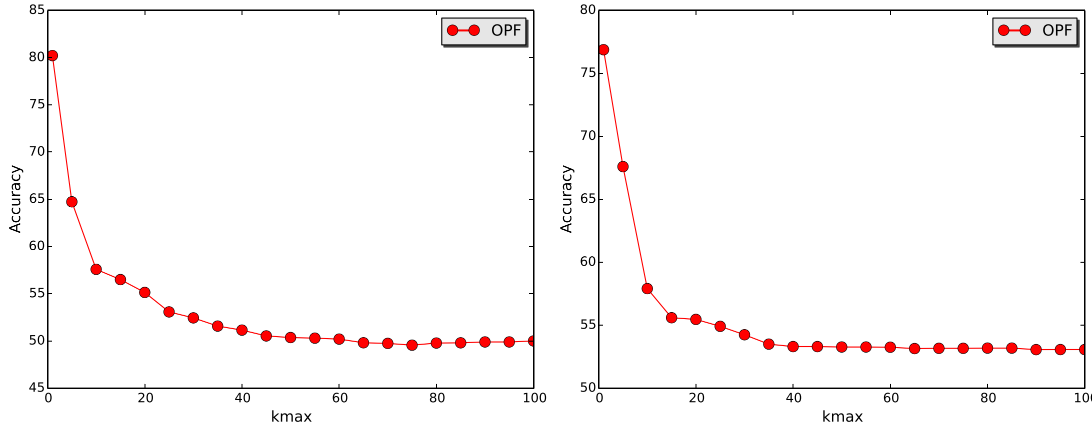
Although the experiments work with unsupervised techniques, a labelled dataset is used to compute the classification accuracies, as follows:

- OPF: after the clustering procedure, a collection of optimum-path trees (clusters) rooted at each prototype is obtained. Then, all samples of a given cluster are labelled with the same label of their root;
- k-Means: after the clustering procedure, a collection of clusters with samples associated with their nearest cluster's centre is obtained. Soon after, all samples from the same cluster are labelled with the very same label of their cluster's centre;
- GMM: a similar procedure adopted for k-means is also used here. After estimating the parameters of each cluster (Gaussian distribution) using E–M algorithm, the cluster's centre (mean parameter) is labelled with the same label of its nearest neighbour, and then such label is propagated to all elements that fall in the very same cluster;
- AP: a similar procedure adopted for k-means and GMM is also employed. After clustering, the centroid of each cluster is labelled with the same class of its nearest neighbour, and such label is propagated to all samples that fall in the same cluster; and
- Birch: The very same approach adopted on the three aforementioned techniques regarding labelling process is also applied for Birch. After clustering, the label closest to the center of each distribution is propagated to the whole cluster.

3.4. Statistical evaluation

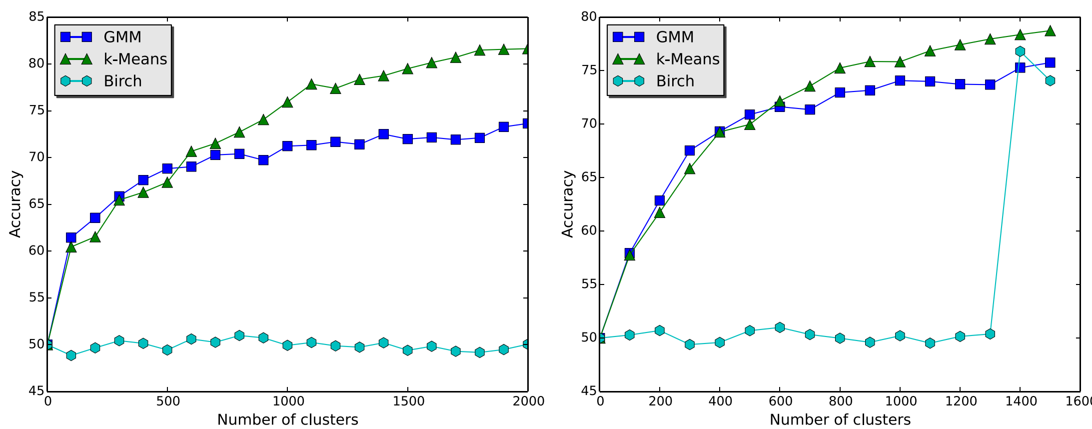
In order to provide a robust evaluation of the techniques employed in this work, a cross-validation procedure with 20 runnings for mean accuracy computation is performed, as well as the standard deviation for both experiments (i.e., unsupervised NTL recognition and anomaly detection). After that, the Wilcoxon

[1] Notice it has been employed the scikit-learn [23] for k-means, AP, GMM and SVM implementations, and LibOPF [24] concerning the OPF classifier.



(a) (b)

Fig. 1. Accuracy values for different k_{max} values over (a) B_c and (b) B_i datasets. The accuracy is computed over the validation set.



(a) (b)

Fig. 2. Accuracy values for different numbers of clusters over (a) B_c and (b) B_i datasets considering Birch, GMM and k-means. The accuracy is computed over the validation set.

signed-rank statistical test [29] is applied to allow a pairwise comparison against all techniques used in this paper with a significance level of 5%.

4. Experimental results

As aforementioned, two rounds of experiments are conducted, being the first one used to asses the robustness of OPF when dealing with unsupervised non-technical losses identification, and the second round aims at evaluating OPF for anomaly detection environments. The next sections describe in more details both experiments.

4.1. Unsupervised non-technical losses detection

This section presents the results concerning unsupervised non-technical losses identification using OPF, AP, GMM, Birch, and k-means. Suppose a fully labelled dataset $\mathcal{Z} = \mathcal{Z}_1 \cup \mathcal{Z}_2 \cup \mathcal{Z}_3$, in which $\mathcal{Z}_1$, $\mathcal{Z}_2$, and $\mathcal{Z}_3$ stand for training, validating, and test sets, respectively. The idea is to employ $\mathcal{Z}_1$ and $\mathcal{Z}_2$ to find the number of clusters for GMM, Birch and k-means, as well as the best preference for AP and a k_{max} value that maximizes the accuracy over $\mathcal{Z}_2$ considering OPF.[2] After that, the best set of parameters is then used to assess the final accuracy value of each classification technique over $\mathcal{Z}_3$.

In order to find out the best parameters for each technique, a near-exhaustive search for $k_{max} \in [1, 100]$ with steps of 5 has been performed, as well as the preference concerning AP between the range of $[-100, 100]$ with steps of 10 and the number of clusters considering GMM, Birch and k-means within the range [1, 2000] and [1, 1500] with steps of 100 for B_c and B_i datasets, respectively. Fig. 1 shows the behaviour of accuracy values for each k_{max} value using OPF considering both datasets, and Fig. 2 shows the evolution of the accuracy curve for each number of clusters considering Birch, GMM and k-means.[3] From those figures, it is possible to realize the landscape of OPF accuracy function is slightly smoother than GMM and k-means ones, which means that one can easily apply some optimization function over OPF accuracy curve in order to find suitable k_{max} values much faster than the near-exhaustive

[2] Notice it has been used 50% of the original dataset for $\mathcal{Z}_1$, 20% for $\mathcal{Z}_2$ and 30% for $\mathcal{Z}_3$. These percentages have been empirically chosen. The idea is to provide a larger training set for learning purposes, as well as a validating and testing sets of similar sizes, since the idea of the validating set is to somehow simulate the real-word.

[3] Considering Birch, we used a threshold of 0.05, which was obtained empirically.

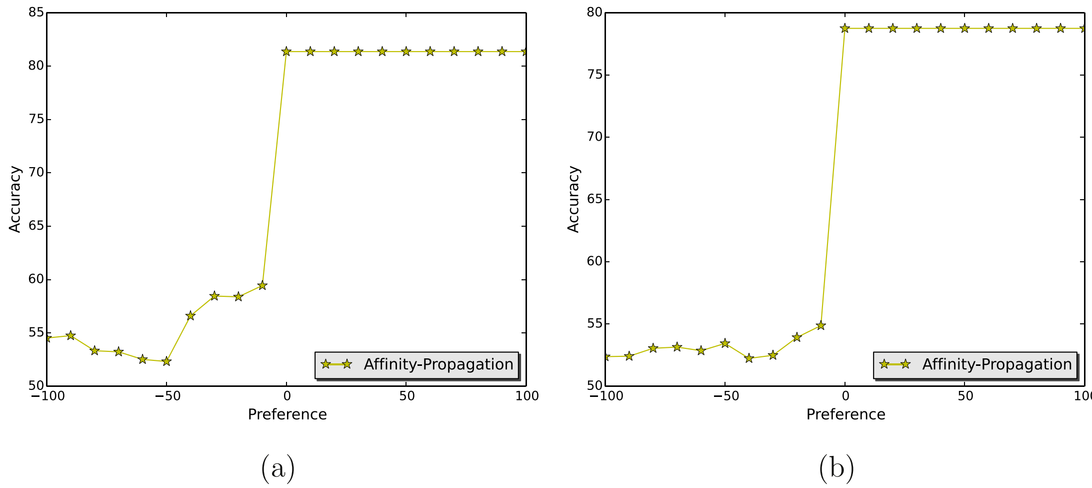


Fig. 3. Accuracy values for different preference values over (a) B_c and (b) B_i datasets. The accuracy is computed over the validation set.

Table 1
Best set of parameters obtained over the validating set.

Technique	B_c		B_i	
	Clusters	k_{max}	Clusters	k_{max}
OPF	1768	1	1142	1
GMM	1357	–	1142	–
k-Means	1736	–	1200	–
Affinity propagation	2471	–	1587	–
Birch	1105	–	1347	–

Table 2
Average accuracy for each clustering technique considering B_c and B_i datasets.

Technique	B_c	B_i
OPF	81.57% ± 2.48	78.30% ± 3.11
GMM	74.64% ± 3.90	74.80% ± 4.35
k-Means	81.51% ± 3.71	77.88% ± 3.48
Affinity propagation	**82.56%** ± **3.00**	**79.51%** ± **2.33**
Birch	51.65% ± 2.23	72.15% ± 12.21

search employed here. Additionally, the OPF fine-tuning procedure is much faster than GMM and *k*-means, since it requires much less optimization steps (the best OPF result is obtained using the first tentative, i.e., with $k_{max} = 1$).

In regard to affinity propagation, it has been tried different values for the “Preference” parameter, as displayed in Fig. 3. Such parameter models the amount of affinity among samples, and points with large preference values are more likely to become exemplars, i.e. centroids of clusters. Clearly, if one uses positive values for such parameter, one can obtain much better results.

Table 1 presents the best number of clusters and k_{max} values obtained in the aforementioned experiment. As a cross-validation procedure with 20 runnings was executed, such values are averaged and rounded to the next nearest integer.

Additionally, Tables 2 and 3 present the average accuracy and *F*-measure for each clustering technique using the parameters obtained in the previous step, respectively. Notice the most accurate results according to the Wilcoxon statistical test are in bold.

Table 3
Average *F*-measure for each clustering technique considering B_c and B_i datasets.

Technique	B_c	B_i
OPF	**0.98** ± **0.002**	**0.97** ± **0.003**
GMM	0.97 ± 0.003	0.97 ± 0.005
k-Means	**0.98** ± **0.003**	**0.97** ± **0.004**
Affinity propagation	**0.98** ± **0.002**	**0.97** ± **0.003**
Birch	0.94 ± 0.015	0.96 ± 0.016

As one can observe, OPF and *k*-means have obtained similar results considering both measures, followed by GMM. Such behaviour is in agreement with the results obtained during the fine-tuning step. The most accurate technique concerning accuracy is the affinity propagation, although its results were quite close to the ones obtained by both OPF and *k*-means. In fact, one can observe all techniques, except GMM and Birch, obtained very good results considering both datasets. Additionally, although all techniques have obtained very close *F*-measure values considering B_c dataset (Table 3), OPF achieved an *F*-measure of 0.98, *k*-means obtained 0.98 and GMM achieved 0.97, which may explain the bolded results as being the best ones for OPF, AP and *k*-means.

Tables 4 and 5 present the Wilcoxon results over B_c and B_i datasets, respectively, where the symbol ‘ $\neq$ ’ denotes there exists a difference between methods, and symbol ‘=’ represents the techniques are similar to each other. As aforementioned, OPF and *k*-means obtained similar results for both datasets considering the accuracy and *F*-measure. Also, the statistical evaluation confirms affinity propagation obtained the best results so far. The values in parenthesis stand for the results considering the *F*-measure.

4.2. Non-technical losses recognition as an anomaly detection problem

This section shows how to handle NTL identification using the framework of anomaly detection. Roughly speaking, anomaly detection techniques aid us to address pattern recognition-oriented problems when little or no information about some specific class

Table 4
Wilcoxon signed-rank test for B_c dataset considering the unsupervised NTL detection task.

Technique	OPF	GMM	*k*-Means	Affinity propagation	Birch
OPF					
GMM	$\neq$ ($\neq$)				
k-Means	= (=)	$\neq$ ($\neq$)			
Affinity propagation	$\neq$ (=)	$\neq$ ($\neq$)	$\neq$ (=)		
Birch	$\neq$ ($\neq$)	$\neq$ ($\neq$)	$\neq$ ($\neq$)	$\neq$ ($\neq$)	

Table 5
Wilcoxon signed-rank rest for B_i dataset considering the unsupervised NTL detection task.

Technique	OPF	GMM	k-Means	Affinity propagation	Birch
OPF					
GMM	$\neq$ ($\neq$)				
k-Means	= (=)	$\neq$ ($\neq$)			
Affinity propagation	$\neq$ (=)	$\neq$ ($\neq$)	$\neq$ (=)		
Birch	= ($\neq$)	= ($\neq$)	= ($\neq$)	$\neq$ ($\neq$)	

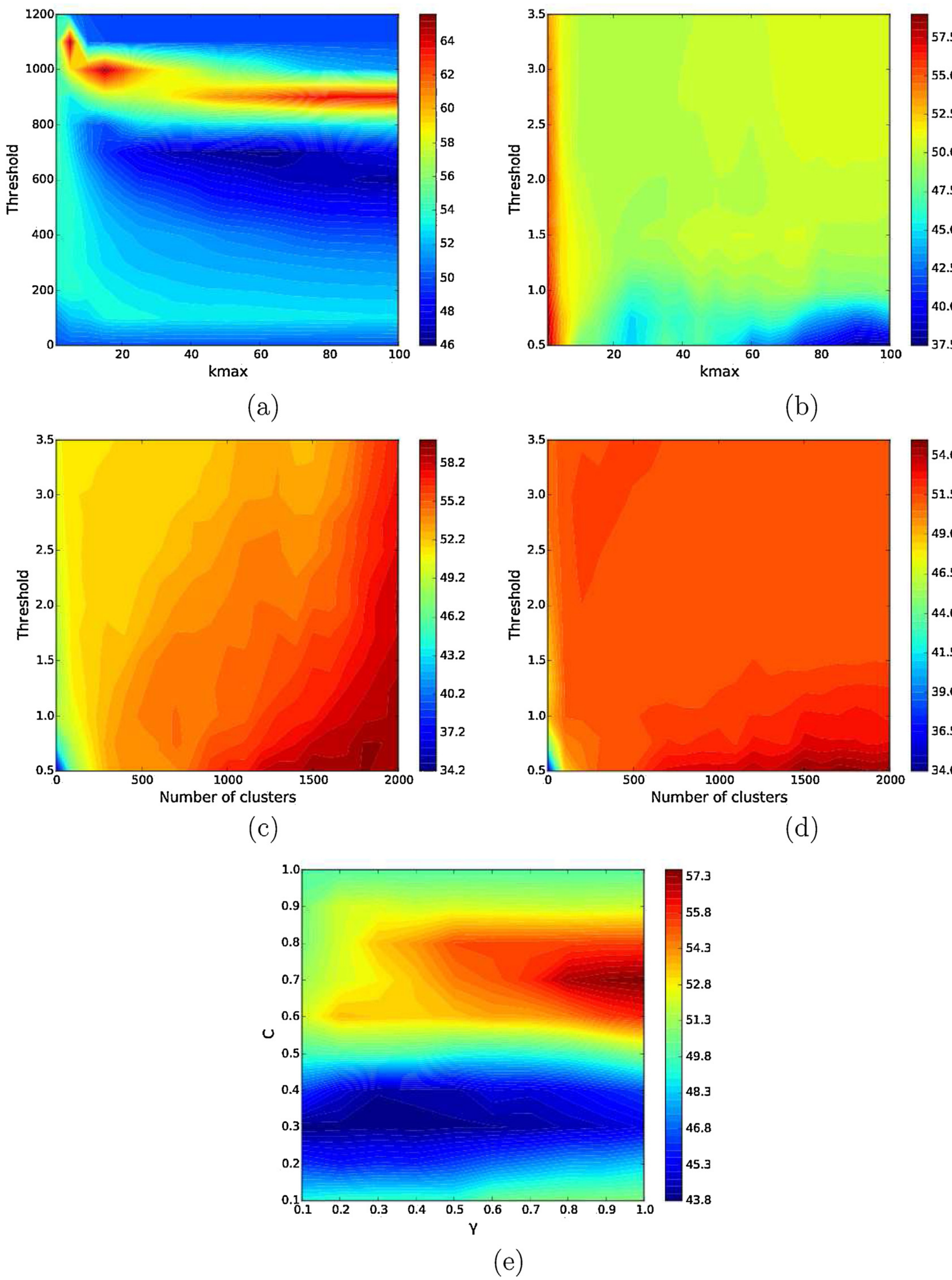


Fig. 4. Grid-search experiment over B_c dataset considering (a) OPF-AD, (b) MGD-OPF, (c) MGD-k-means, (d) MGD-GMM and (e) SVMs. (For interpretation of the references to colour in text, the reader is referred to the web version of the article.)

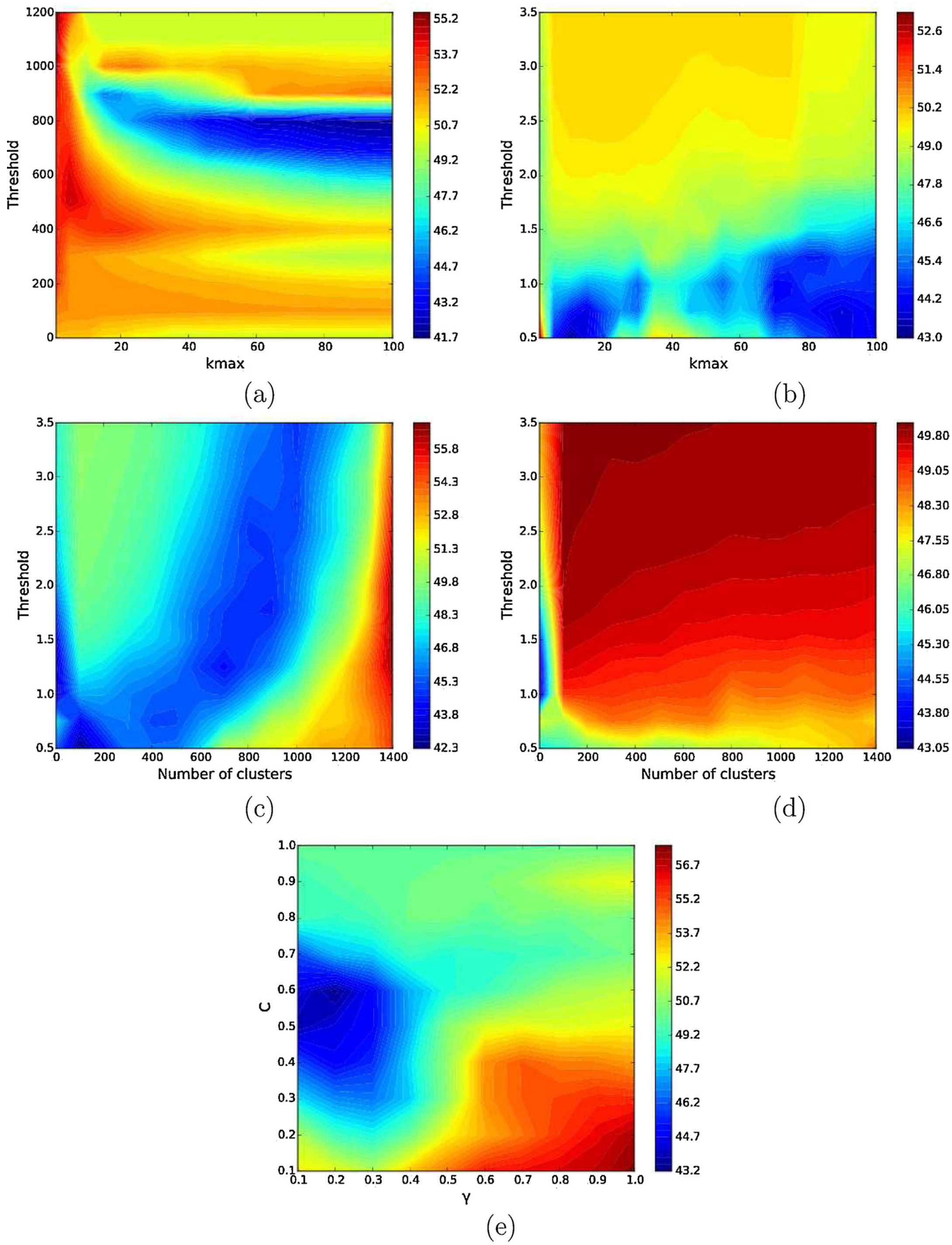


Fig. 5. Grid-search experiment over B_i dataset considering (a) OPF-AD, (b) MGD-OPF, (c) MGD-*k*-means, (d) MGD-GMM and (e) SVMs. (For interpretation of the references to colour in text, the reader is referred to the web version of the article.)

are obtained. Considering the problem of non-technical losses identification, two problems are usually faced: (i) it is not straightforward to design a labelled dataset for such purposes, and (ii) it is difficult to build a balanced dataset, since the amount of irregular consumers is often lower than regular consumers. The main reason for that concerns the problem of associating to a given consumer the thief label, since some legal problems are usually associated with that task.

Therefore, anomaly detection techniques allow us to learn the behaviour of one class only (regular consumers in our case), and then when a new sample comes to be labelled, it is decided whether this sample belongs to the "normal" behaviour learned by the classifier (regular consumers) or it will be classified as an anomaly (irregular consumers).

One of the most used approaches for anomaly detection is the multivariate Gaussian distribution (MGD), which aims at modelling each cluster of samples as a Gaussian distribution, and further when a new sample comes to be classified, it is computed its probability of belonging to each cluster (Gaussian). After that, one takes the highest probability and verifies whether it is less than a predefined threshold (usually the Mahalanobis distance) to be classified as a normal sample; otherwise, it is classified as an anomaly.

Table 6
Average accuracy and standard deviation considering the anomaly detection task.

Dataset	MGD-OPF	MGD-k-means	MGD-GMM	OPF-AD	SVMs
B_i	52.60% ± 3.66	53.64% ± 4.51	48.41% ± 1.77	55.20% ± 2.92	**57.44% ± 3.68**
B_c	57.94% ± 3.14	59.52% ± 2.52	55.61% ± 2.33	**64.54% ± 3.00**	56.11% ± 4.50

Table 7
F-measure and standard deviation considering the anomaly detection task.

Dataset	MGD-OPF	MGD-k-means	MGD-GMM	OPF-AD	SVMs
B_i	0.72 ± 0.10	0.64 ± 0.06	**0.88 ± 0.11**	**0.78 ± 0.27**	0.82 ± 0.13
B_c	0.85 ± 0.04	0.82 ± 0.03	**0.93 ± 0.01**	0.70 ± 0.09	0.48 ± 0.06

Table 8
Wilcoxon signed-rank rest for B_i dataset considering the anomaly detection task.

Technique	MGD-OPF	MGD-GMM	MGD-k-means	OPF-AD	SVMs
MGD-OPF					
MGD-GMM	≠ (≠)				
MGD-k-means	= (≠)	≠ (≠)			
OPF-AD	≠ (=)	≠ (=)	= (=)		
SVMs	≠ (≠)	≠ (≠)	≠ (≠)	≠ (=)	

However, the main problem with MGD-based anomaly detection concerns with the Gaussian parameter estimation, which is usually performed using an unsupervised fashion with k-means or expectation–maximization. The problem with k-means concerns with the priori knowledge about the number of classes required as an input, and E–M usually may get trapped from local optima during the convergence process. Such assumptions motivated us to employ OPF to estimate Gaussian parameters, since the prototype nodes are usually located at the centre of the classes, being suitable representatives for the expectation (mean) of the Gaussian distribution [30]. Therefore, OPF, k-means and GMM are evaluated to estimate Gaussian distribution parameters in the context of anomaly detection, henceforth called as MGD-OPF, MGD-k-means and MGD-GMM.

Another contribution of this work is to propose an anomaly detection approach purely based on OPF classifier. Therefore, instead of using MGD for modelling samples of regular consumers, unsupervised OPF is employed to cluster a dataset composed of regular samples only. In order to identify a new sample as regular or irregular consumer, the naïve OPF classification process described in Section 2 is performed. After that, the cost given to the test label by the winner sample (conqueror) is compared, in order to check whether it is lower than a threshold value to label this sample as a regular consumer; otherwise, this sample is then considered an anomaly. This approach is called OPF for anomaly detection (OPF-AD).

First of all, MGD-OPF, MGD-k-means, MGD-GMM, OPF-AD and one-class SVMs parameters are fine-tuned using a validating set with 20% of the dataset samples by means of a grid-search. In regard to MGD-OPF and OPF-AD, the only parameter to be optimised is $k_{max} \in [1, 100]$ with step of 5, while for MGD-k-means and MGD-GMM is necessary to optimise the number of clusters within the range [1, 2000] with step of 100. One-class SVMs require the optimization of both $C \in [0.1, 1]$ and $\gamma \in [0.1, 1]$ parameters with step of 0.1, being the last one related to the RBF kernel. Figs. 4 and 5 display the grid-search results over B_c and B_i datasets, respectively, where the "hot zones" (red colours) stand for the configuration of parameters that led to the most accurate results. It is important to shed light over this experiment aimed at selecting the most suitable set of parameters that maximized the accuracy presented in Papa [27]. Although any other measure could be used, this work opted to consider such one since it can fit well to the problem of unbalanced datasets.

Although MGD-GMM has a larger area of hot zones considering B_c dataset (Fig. 4d), which is desirable since the probability of choosing a point within such areas is higher than other techniques using some random search, MGD-GMM has achieved lower recognition rates when compared to MGD-k-means, MGD-OPF and OPF-AD, being the latter one the most accurate technique in the grid-search experiment.[4] In regard to B_i dataset, SVMs have achieved the best recognition rates, followed by OPF-AD.

Tables 6 and 7 present the mean recognition rates and mean F-measure values, respectively, considering the anomaly detection task for both datasets, in which similar techniques according to Wilcoxon signed-rank test are in bold. Observe that OPF-based approaches have obtained suitable results, being OPF-AD the most accurate technique considering B_c dataset, and SVMs obtained the best result over B_i dataset. The main problem regarding MGD-based techniques for anomaly detection is to assume the clusters can be represented by Gaussian distributions, which may not occur in practice. On the other hand, OPF does not assume distribution-based classes, and its parameter k_{max} is much less sensitive than the number of classes required by k-means, for instance. Such assumption can be observed over B_c dataset, in which OPF-AD has been the best classifier, with recognition rates about 11.93% better than MGD-k-means technique, for instance.

Table 7 presents the results concerning the F-measure, where MGD-GMM obtained the top results over both datasets, with results similar to OPF-AD considering B_i dataset. Such measure computes the harmonic mean between precision and recall, which are often used in the context of information retrieval. Since precision is defined as the number of true positive samples divided by both true positive and false negative samples, one can obtain some estimative about the effectiveness of the methods when dealing with classification (precision ranges within [0, 1], where 1 is the best value). On the other hand, recall measures the amount of samples from a given class that were correctly labelled as being from that class (recall ranges within [0, 1], where 1 is the best value). The combination of

[4] Notice the y-axes (threshold) in OPF-AD experiment are different to the other techniques, since we are working with the cost associated with each sample, while in MGD-OPF, MGD-k-means and MGD-GMM the y-axes are within the same interval. Therefore, a grid-search in the interval [0, 1200] with steps of 100 for OPF-AD was performed to obtain suitable threshold values, and with respect to the remaining techniques the grid-search was conducted within the interval [0.5, 3.5] with steps of 0.25. Notice all step values have been empirically chosen.

Table 9
Wilcoxon signed-rank rest for B_c dataset considering the anomaly detection task.

Technique	MGD-OPF	MGD-GMM	MGD-k-means	OPF-AD	SVMs
MGD-OPF					
MGD-GMM	$\neq$ ($\neq$)				
MGD-k-means	$\neq$ ($\neq$)	$\neq$ ($\neq$)			
OPF-AD	$\neq$ ($\neq$)	$\neq$ ($\neq$)	$\neq$ ($\neq$)		
SVMs	= ($\neq$)	= ($\neq$)	$\neq$ ($\neq$)	$\neq$ ($\neq$)	

both measures allows an interesting quantitative evaluation of the results. Since the datasets used in this work are composed of regular consumers mostly, MGD-GMM seemed to better manage such unbalancing concerning B_c dataset. Probably the behaviour of samples followed a Gaussian distribution, which indeed has favoured the method.

Tables 8 and 9 present the Wilcoxon statistical test results for B_i and B_c datasets, respectively. The symbols in parenthesis stand for the results considering the F-measure, while the ones outside refer to the results considering the accuracy measure. As aforementioned, with respect to B_i dataset, SVMs obtained the best results considering the accuracy measure only, while MGD-GMM and OPF-AD achieved the top F-measure values. Since OPF-AD has a considerably greater standard deviation when compared to SVM and MGD-GMM, it has been considered similar to the latter one by the statistical evaluation. In regard to B_c dataset, OPF-AD obtained the best results considering the accuracy rate, whereas MGD-GMM obtained the best F-measure value.

5. Conclusions

In this paper, the problem of NTL recognition by means of two distinct paradigms is tackled: (i) unsupervised NTL recognition, and (ii) NTL recognition as an anomaly detection problem. The latter approach is more appropriate when little or no information about a given class are obtained. Therefore, the classification technique is trained with profiles of regular users only, and when a new consumer comes to be classified, the machine learning technique does not try to fit it as belonging to the regular or irregular consumer, as usually occurs with traditional pattern recognition techniques. Instead, the classifier tries to identify such sample as being a regular consumer, and if its signature does not fit to a regular profile, it is then labelled as an anomaly (irregular consumer).

Two rounds of experiments were conducted in order to assess the robustness of OPF classifier for both unsupervised NTL recognition and anomaly detection against k-means and GMM techniques, as well as AP and Birch for the task of unsupervised NTL recognition and one-class SVM for the task of anomaly detection. OPF has obtained the most accurate results considering both applications on two datasets composed of commercial and industrial profiles of regular and irregular consumers. Therefore, this work also contributed to the literature related to unsupervised NTL detection, which lacks on such sort of works. Additionally, a model for the problem of NTL recognition as an anomaly detection problem is proposed.

In regard to future works, we aim at comparing OPF against several other clustering techniques, as well as to evaluate the robustness of unsupervised OPF together with feature selection techniques. Also, we aim at studying different methodologies to estimate the pdf of each sample concerning OPF clustering, since different neighbourhood sizes may affect the quality of samples' density computation.

Acknowledgements

The authors would like to thank CAPES, FAPESP grants #2009/16206-1, #2012/14158-2, #2013/20387-7, #2014/16250-9 and #2015/00801-9, and also CNPq grants #303182/2011-3, #470571/2013-6 and #306166/2014-3.

References

[1] S. Pan, T. Morris, U. Adhikari, Developing a hybrid intrusion detection system using data mining for power systems, IEEE Trans. Smart Grid 6 (2015) 3104–3113.
[2] J. Nagi, K. Yap, S. Tiong, S. Ahmed, F. Nagi, Improving SVM-based nontechnical loss detection in power utility using the fuzzy inference system, IEEE Trans. Power Deliv. 26 (2011) 1284–1285.
[3] J. Nagi, K. Yap, S. Tiong, S. Ahmed, M. Mohamad, Nontechnical loss detection for metered customers in power utility using support vector machines, IEEE Trans. Power Deliv. 25 (2010) 1162–1171.
[4] K. Yap, S. Tiong, J. Nagi, J. Koh, F. Nagi, Comparison of supervised learning techniques for non-technical loss detection in power utility, Int. Rev. Comput. Softw. 7 (2012) 626–636.
[5] C. Ramos, A. Sousa, J. Papa, A. Falc ao, A new approach for nontechnical losses detection based on optimum-path forest, IEEE Trans. Power Syst. 26 (2011) 181–189.
[6] C. Ramos, A. Souza, G. Chiachia, A. Falc ao, J. Papa, A novel algorithm for feature selection using harmony search and its application for non-technical losses detection, Comput. Electr. Eng. 37 (2011) 886–894.
[7] C.C.O. Ramos, A.N. de Souza, A.X. Falc ao, J.P. Papa, New insights on non-technical losses characterization through evolutionary-based feature selection, IEEE Trans. Power Deliv. 27 (2012) 140–146.
[8] M. Martino, F. Decia, J. Molinelli, A. Fernández, A novel framework for non-technical losses detection in electricity companies, in: P. Latorre Carmona, J. Sánchez, A. Fred (Eds.), Pattern Recognition - Applications and Methods, vol. 204 of Advances in Intelligent Systems and Computing, Springer Berlin Heidelberg, 2013, pp. 109–120.
[9] C. León, F. Biscarri, I. Monedero, J. Guerrero, J. Biscarri, R. Millán, Integrated expert system applied to the analysis of non-technical losses in power utilities, Expert Syst. Appl. 38 (2011) 10274–10285.
[10] I. Monedero, F. Biscarri, C. León, J. Guerrero, J. Biscarri, R. Millán, Using regression analysis to identify patterns of non-technical losses on power utilities, in: Knowledge-Based and Intelligent Information and Engineering Systems, vol. 6276 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2010, pp. 410–419.
[11] I. Monedero, F. Biscarri, C. León, J. Biscarri, R. Millán, MIDAS: detection of non-technical losses in electrical consumption using neural networks and statistical techniques, in: Computational Science and Its Applications, vol. 3984 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2006, pp. 725–734.
[12] J. Guerrero, C. León, I. Monedero, F. Biscarri, J. Biscarri, Improving knowledge-based systems with statistical techniques, text mining, and neural networks for non-technical loss detection, Knowl. Based Syst. 71 (2014) 376–388.
[13] C. Donadel, J. Anicio, M. Fredes, F. Varejao, G. Comarela, G. Perim, A methodology to refine the technical losses calculation from estimates of non-technical losses, in: Proceedings of the 20th International Conference and Exhibition on Electricity Distribution – Part 1, 2009, pp. 1–4.
[14] D. Tasić, M. Stojanović, Fuzzy approaches to distribution energy losses calculation, Acta Electrotech. Inf. 5 (2005) 1–7.
[15] T. Babu, T. Murthy, B. Sivaiah, Detecting unusual customer consumption profiles in power distribution systems - APSPDCL, in: IEEE International Conference on Computational Intelligence and Computing Research, 2013, pp. 1–5.
[16] L. Rocha, F. Cappabianco, A. Falcão, Data clustering as an optimum-path forest problem with applications in image analysis, Int. J. Imaging Syst. Technol. 19 (2009) 50–68.
[17] C. Ramos, A. Souza, R. Nakamura, J. Papa, Electrical consumers data clustering through optimum-path forest, in: 16th International Conference on Intelligent System Application to Power Systems, 2011, pp. 1–4.
[18] B. Schölkopf, R.C. Williamson, A.J. Smola, J. Shawe-Taylor, J.C. Platt, Support vector method for novelty detection, in: S.A. Solla, T.K. Leen, K. Müller (Eds.), Advances in Neural Information Processing Systems, MIT Press, 2000, pp. 582–588.
[19] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 888–905.
[20] C. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, New York, NY, USA, 2008.
[21] D. Arthur, S. Vassilvitskii, K-means++: the advantages of careful seeding, in: Proceedings of the 18th Annual ACM–SIAM Symposium on Discrete Algorithms, SODA '07, 2007, pp. 1027–1035.

[22] Scikit-Learn Developers, User Guide, 2016, Available at http://scikit-learn.org/dev/user_guide.html.
[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[24] J. Papa, C. Suzuki, A. Falcão, LibOPF: a library for the design of optimum-path forest classifiers, 2014, Software version 2.1. Available at http://www.ic.unicamp.br/afalcao/LibOPF.
[25] B.J. Frey, D. Dueck, Clustering by passing messages between data points, Science 315 (2007) 972–976.
[26] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering method for very large databases, in: 1996 ACM SIGMOD International Conference on Management of Data, SIGMOD '96, ACM, New York, NY, USA, 1996, pp. 103–114.
[27] J. Papa, A. Falc ao, C. Suzuki, Supervised pattern classification based on optimum-path forest, Int. J. Imaging Syst. Technol. 19 (2009) 120–131.
[28] D.M.W. Powers, Evaluation: from precision, recall and *F*-measure to roc, informedness, markedness and correlation, Int. J. Mach. Learn. Technol. 2 (2011) 37–63.
[29] F. Wilcoxon, Individual comparisons by ranking methods, Biom. Bull. 1 (1945) 80–83.
[30] G. Rosa, K. Costa, L. Passos Junior, J. Papa, A. Falcão, J. Tavares, On the training of artificial neural networks with radial basis function using optimum-path forest clustering, in: 22nd International Conference on Pattern Recognition, 2014, pp. 1472–1477.