Intelligent Data Analysis 14 (2010) 1–21 1
DOI 10.3233/IDA-2010-0461
IOS Press

Active learning and subspace clustering for anomaly detection

Karim Pichara and Alvaro Soto

Abstract. Today, anomaly detection is a highly valuable application in the analysis of current huge datasets. Insurance companies, banks and many manufacturing industries need systems to help humans to detect anomalies in their daily information. In general, anomalies are a very small fraction of the data, therefore their detection is not an easy task. Usually real sources of an anomaly are given by specific values expressed on selective dimensions of datasets, furthermore, many anomalies are not really interesting for humans, due to the fact that interestingness of anomalies is categorized subjectively by the human user. In this paper we propose a new semi-supervised algorithm that actively learns to detect relevant anomalies by interacting with an expert user in order to obtain semantic information about user preferences. Our approach is based on 3 main steps. First, a Bayes network identifies an initial set of candidate anomalies. Afterwards, a subspace clustering technique identifies relevant subsets of dimensions. Finally, a probabilistic active learning scheme, based on properties of *Dirichlet* distribution, uses the feedback from an expert user to efficiently search for relevant anomalies. Our results, using synthetic and real datasets, indicate that, under noisy data and anomalies presenting regular patterns, our approach correctly identifies relevant anomalies.

Keywords: Anomaly Detection, Active Learning, Dirichlet Distribution, Bayessian Network, Probabilistic Model

1. Introduction

In this paper, we propose a new algorithm to detect relevant anomalous records in large datasets. We consider an anomaly as relevant, or interesting, if its detection is valuable to a human user. Depending of the domain, these anomalies might correspond to fraudulent transactions in a financial dataset, new phenomena in scientific information, records of faulty products in a manufacturing database, or fraudulent situations in an insurance company, among others [20]. Our main hypothesis is that, in practical cases, interesting anomalies usually present regular patterns that form micro clusters within the data. Furthermore, these micro clusters appear in selective dimensions, or subspaces, of the input data. As an example, consider the case of detecting strange celestial objects in an astronomical data catalog [12]. These catalogs contain billions of records, each one characterized by dozens or hundreds of attributes. In this case, some relevant and recently discovered objects, such as brown dwarfs, correspond to a tiny fraction of the records in the dataset, which are best characterized by a specific subset of attributes, like particular infrared emissions and presence of lithium in the case of brown dwarfs [41]. A similar situation occurs in the banking sector, where a new type of fraud usually consists of a small number of transactions that share a specific subset of attributes.

In general, the identification of anomalies laying among huge amounts of data and high dimensionality spaces is a difficult task. Huge number of records contributes to "hide" the presence of anomalies behind

*Corresponding author. E-mail:kpb@ing.puc.cl.

normal data points and high dimensionality spaces increases the difficulties to identify suitable subspaces to search for anomalous records. In this work, we deal with these two problems by applying a two-step data-driven scheme. First, our approach uses a Bayesian network (BN) to model the joint probability density function (pdf) of the input data [33,31]. We build upon our previous work [9,42] that efficiently finds a BN in a large and high dimensional dataset. The resulting joint pdf provides a straight forward method to rank records according to their oddness.

Afterwards, as a second step, we apply a subspace clustering algorithm within the reduced set of uncommon records. This second step generates a set of candidate subspaces, and micro clusters within these subspaces, where we search for anomalous records.

While the previous scheme is able to provide a set of micro clusters of uncommon records, located in selected subspaces of the input data, from a practical point of view, most of these anomalous records can be irrelevant to a human user. The main inconvenience is that the relevance of an unusual record is highly dependent of the domain under consideration. For example, in a fraud detection application, identification of already known types of anomalies, or detection of an unusual but otherwise legal business transaction, might be of a null value to a human user. Furthermore, noisy data, inherent to most large datasets, often produces the formation of spurious patterns whose identification might also be irrelevant to a human user. As suggested by these cases, anomaly detection methods constructed only from low-level features stored in a dataset can provide a significant number of semantically irrelevant detections.

The previous observation suggests to augment the two-step data-driven scheme described before by adding a third step that helps us to bridge the gap between an unsupervised low-level data analysis and the semantic knowledge that can be provided by a human expert. We achieve this by using an active learning scheme. Under this scheme, the algorithm selectively asks a human expert for feedback, searching for informative data points which, if labeled, can improve the detection of truly semantically relevant anomalies. Our rationale is that while computers can process huge amounts of low-level data, a human expert can provide high-level semantic knowledge to efficiently lead the search for relevant anomalies.

Our active learning technique is based on a generative Bayesian probabilistic approach that exploits clustering properties of the *Dirichlet* distribution [16]. Under this generative model we keep track of the posterior distribution that expresses the relevance of each anomaly given selective semantic feedback requested from a human expert. In contrast to traditional active learning techniques [28,10], where the main focus is to boost the performance of a classifier, in our application the main interest is to quickly learn to identify records that are relevant to a human user. In other words, our main interest is not the inductive generalization property of the classifier but to quickly capture the underlying knowledge of the expert to guide the algorithm to identify relevant anomalies while minimizing the amount of feedback requested from him.

In summary, our semi-supervised anomaly detection approach is based on 3 main steps. First, a BN is used to reduce the size of the input dataset by building an unsupervised probabilistic model, where an initial set of candidates anomalies is identified as low likelihood records. Afterwards, within this initial set of candidate anomalies, a subspace clustering technique identifies relevant subspaces and micro clusters. These two first steps provide an initial prior distribution about the relevance of each candidate anomalous records. Finally, a probabilistic active learning scheme interacts with a human expert to quickly reveal the list of true semantically relevant anomalous records.

From the previous 3 steps, here we focus in the last two, main details of the initial detection of candidate anomalies using a BN are described elsewhere [9]. Accordingly, the main contributions of this work are: i) To incorporate a subspace clustering technique that provides suitable subsets of variables and

micro clusters to search for relevant anomalies, ii) To develop a new active learning strategy that uses a Bayesian probabilistic approach to quickly discovering semantically relevant anomalous records, and iii) To implement and validate these ideas using synthetic and real datasets.

This paper is organized as follows. Section 2 reviews relevant previous work. Section 3 discusses our approach in detail. Section 4 shows the results of applying our methodology to synthetic and real datasets. Finally, Section 5 presents the main conclusions of this work.

2. Related Work

There is an extensive literature related to anomaly detection problems [20,26,45]. In general, Anomaly Detection methods can be classified in two main classes, unsupervised and supervised methods. Unsupervised methods detect anomalies as low density areas of the feature space. Supervised methods use labeled data to inductively learn about anomalous patterns. In both cases, it is possible to boost performance by adding an active learning scheme that incorporates direct feedback from a domain expert. Next, we review relevant works in each of these areas.

Among unsupervised techniques, statistical approaches are the earliest methods used for anomaly detection. These approaches detect anomalies as outliers that deviate markedly from most of the observations, therefore, they are located in low likelihood areas of the input data [18].

Many statistical methods models the data using mixture models, in particular, using Gaussian distributions [2,13]. The typical strategy consists of calculating a score and a threshold that are used to identify points that deviate from normal data. In [13] they propose an algorithm that fits mixture models using the Expectation Maximization (EM) algorithm. They fit models to normal and anomalous records assuming that each record has a prior probability λ to be anomalous. Then, they obtain an anomaly score that is based on measuring the variation of the normal distribution when a point is moved to the anomalous distribution. Statistical tests have also been used to detect anomalies. In [43] they use a Student's t-test. A normal sample N_1 is compared with a test sample N_2. If the test shows significant difference between them, N_2 is considered as an anomaly. Ye and Chen [47] use a χ^2 statistic to determine anomalies. The training phase assumes that normal data has a multivariate Gaussian distribution. Afterward, they label a new test record as an anomaly if its value presents a high deviation from its expected value under the distribution assigned to normal data. A related approach is to use a regression model to detect anomalies [5,11]. In this case, after fitting a model, anomalous records are detected as points with large residuals.

Some rule based methods are proposed for outlier detection [15,27]. The work proposed in [15] present an activity monitoring system for detecting fraudulent activity on news story monitoring, they use a classifier to learn rules that discriminates between normal and fraudulent activity. The learned rules creates profiles of user's behavior to model single entities like cell phone accounts, customers, users, etc. The behavior profiles related to a single activity feeds into a detector which combines them and generates an alarm if the deviation from the normal activity exceeds a threshold. Lane and Brodley [27] also introduced an activity monitoring system for fraud detection, they store and record activities of users on a computer system as sequences. Using case based learning techniques they compare command sequences issued by users against the stored sequences.

In the context of unsupervised methods, clustering techniques have been proposed to detect anomalous records. BIRCH [49] uses local clustering and data compression to incrementally cluster all data points in hierarchies. It uses a specialized tree structure designed to operate with large datasets inducing a good quality clustering from a single pass through data. To detect anomalies, BIRCH finds all nodes

having isolated points that are not merged with other nodes during the process. DBSCAN [14] is a density based clustering algorithm, finding clusters of arbitrary shapes. DBSCAN starts from points having more neighbors (core points) and adds points that can be reached moving through the neighbors of the neighbors and so on. Once the algorithm finish, all points non-reachable from core points are considered as anomalies. Most recent clustering algorithms proposed for anomaly detection are in the context of intrusion detection in networks, see [46,40]. Unfortunately, clustering algorithms suffer from the curse of dimensionality problem. In effect, in large dimensional spaces, typical distance metrics used to characterize similarity do not provide suitable clusters. Notably, subspace clustering algorithms have not been commonly use for anomaly detection. A notably exception is a recent work proposed in [38], they perform a subspace clustering algorithm to rank data points according to the size of the clusters and the number of dimensions on each subspace where points belong.

In the context of supervised methods, Nearest Neighbors techniques have been used to detect anomalies, the main intuition is that anomalies are records with less neighbors than normal points [36,25]. Breuning et al. [8] assign to each data instance an anomaly score called Local Outlier Factor (LOF). This score is given by the ratio between the local density of the point and the average local density of its k nearest neighbors. Local density is calculated using the radius of the smallest hyper-sphere that is centered at the data instance and contains k nearest neighbors. Papadimitriou et al. [32] propose a variant of the LOF called Multi Granularity Deviation Factor (MDEF). For a given record, its MDEF is calculated as the standard deviation among its local density and the local densities of its k nearest neighbors. They use the MDEFs to search for micro clusters of anomalous records. In the same lines, Jin et al. [23] propose another variant of LOF that improves efficiency by avoiding unnecesary calculations. They achieve this by calculating upper and lower bounds among the micro clusters detected.

In terms of parametric methods for supervised learning, many different anomaly detection algorithms have been proposed, such as decision trees [24,4] and neural networks [30,6]. Decision Trees algorithms fit the data focusing only on salient attributes, a desirable characteristic when dealing with high dimensional data. These algorithms works modeling all points corresponding to normal classes, then points having an erroneous or unexpected classification are considered as anomalies. Similarly, neural networks are used to model the unknown distribution of normal class points, training a feed forward network adjusting the weights and thresholds learning from the input data. Neural Networks works well when training sets are representative of the unseen data, unfortunately that not occurs for new instances that are out of the scope of the training set. Decision trees and Neural Networks are susceptible to over fitting when no stopping criteria are well determined.

In the context of active learning, the main focus has been on improving the accuracy of a classifier by actively deciding what instance to label [39,28,37,44]. The Query by Committee algorithm [39] queries instances that present the greatest disagreement among a set of classifiers. The Uncertainty Sampling algorithm [28] queries instance with greatest uncertainty (margin). A similar approach is presented by Tong and Koller in the context of support vector machines [44]. In [37], Roy and McCallum use a heuristic to gather information from the instance that minimizes the expected error in the classification of future observations. In a work more closely related to our approach, Zhang and Chen [48] propose a scheme that uses active learning to determine a relevant subspace to characterize each object in a dataset. The system asks an expert to annotate the important features for a given object. Probability vectors for non annotated objects are estimated using kernel regression, being more influenced by its annotated neighbors. The active learning scheme selects the element having more uncertainty about its probability vector as the next object to be annotated, the uncertainty is estimated with the entropy of the joint probability distribution of the object.

In terms of of anomaly detection, Pelleg and Moore [34] propose an active learning strategy that fits a mixture model to the data using a modified version of the EM-GMM algorithm. The algorithm is able to include labeled data in the maximization step. At each iteration, the active learning scheme asks a domain expert to label instances that have low likelihood values under their closest mixture component. Abe et al. [1] propose an algorithm that reduces the anomaly detection problem to a classification problem by obtaining a labeled dataset with artificially generated anomalies. They use active learning to selective query the label of instances from the original dataset and refine the classifier model. He and Carbonell [19] use a nearest neighbor approach to calculate local densities of data points. Afterwards, they use active learning to query the label of points with highest variation on it local densities.

3. Our Approach

This section describes the main steps of our approach. As we mentioned before, our algorithm is based on three main steps: 3.1) Unsupervised identification of initial set of candidate anomalies using a BN; 3.2) Unsupervised identification of subspaces and micro clusters using subspace and probabilistic clustering techniques; and 3.3) Semi-supervised identification of semantically relevant anomalies using a probabilistic active learning scheme. Next, we refer to each of these steps.

3.1. Unsupervised identification of initial set of candidate anomalies

As a first step of our algorithm, we fit a BN to the records in the input dataset [9]. The algorithm proposed in [9] uses a variant of the Sparse Candidate Algorithm [17], they search for different Bayes networks adding, deleting and reversing arcs for each node. Instead of using traditional Greedy Hill Climbing search, they shrink the search space by statistically selecting the most probable parents for each variable. The factorization of the joint pdf provided by the BN allows us to efficiently estimate the likelihood of each record $x = (x_1, \ldots, x_m)$ as $P(x) = \prod_i^m p(X_i = x_i | Pa^G(X_i))$, where G is the acyclic directed graph that defines the BN, $Pa^G(X_i)$ is the set of direct parents of X_i in G, and m is the total number of attributes in the dataset. This algorithm runs in time $O(mn^2)$, where n is the number of nodes considered in the network and m are the number of instances in the dataset.

If the training of the BN is successful, uncommon records appear as low probability objects. We use these probability values as indicators of the degree of rareness of each record in the dataset. Our experience indicates that the detection of anomalous records only based on this criterion often produces a great number of false positives. Therefore, we only use the BN as an initial step that helps us to filter the input dataset by identifying a set of candidate anomalous records given by the first τ records with lowest likelihood. Deciding the correct value of τ depends directly on the capacity of the BN to fit the data. In our experience, anomalous records usually fall within the 5 to 15% of the records with lowest probability under the BN model. In this work we use $\tau = 0.15$ to obtain the set of filtered records or filtered dataset (FDS).

3.2. Unsupervised identification of subspaces and micro clusters

As we mentioned before, one of our main assumptions is that interesting anomalies form micro clusters that are embedded in selected subspaces of the input data. Accordingly, the second step of our algorithm uses subspace and probabilistic clustering techniques to find those subspaces and micro clusters. Given that this process is carried out in the reduced set of candidate anomalies (FDS), we refer to the detected

patterns as micro clusters with respect to the original dataset. As we explain next, we divide this step in two main tasks: initial identification of relevant subspaces, and identification of micro clusters within these subspaces.

3.2.1. Identification of relevant subspaces

We use the CLIQUE algorithm [3] to identify relevant subspaces. CLIQUE finds subspaces by combining density and grid based clustering techniques in conjunction with a monotonicity principle on the density of the grid cells. As one of the main steps of its operation, the CLIQUE algorithm needs the setting of a parameter that indicates when a specific set of cells can be considered as dense. In general, the performance of the algorithm is very sensitive to the value of this parameter. Fortunately, in our case the setting of this parameter is less critical. This is because in a subsequent processing step the detected subspaces are efficiently filtered with the help of the active learning scheme. Consequently, we use a conservative value for the density threshold overestimating the selection of potentially relevant subspaces. If dense units exists in k dimensions, all of their projections in a subset of the k dimensions are also dense, they are $O(2^k)$ different combinations. The Clique algorithm is exponential in the highest dimensionality of any dense unit. For a dataset with m points and k variables, the running time of Clique is $O(c^k + mk)$ for a constant c [3].

3.2.2. Identification of micro clusters

After detecting relevant subspaces, we search for micro clusters by fitting a mixture model inside each subspace. At the beginning of this process, each record in the FDS receives a weight inversely proportional to the likelihood assigned to the point by the initial BN. This increases the relevance used by the mixture model to fit the most strange records according to the output of the BN.

Depending if records in the input dataset contain continuous or categorical values, we apply a different strategy to fit a mixture model. In case of continuous data, we fit a Gaussian mixture model (GMM) using an accelerated version of the expectation-maximization algorithm [42]. In case of categorical data, we fit clusters by using the implementation of the k-modes algorithm provided in [21]. In the categorical case, we also use a closest mode scheme to assign points to clusters. We use these assignations to estimate the mixing probabilities of the mixture model. Finally, in the continuous and categorial case, we initially overestimate the number of components in the mixture, then we iteratively learn a suitable value by using feedback from a domain expert through the active learning scheme that is described in detail in the next section.

3.3. Semi-supervised identification of semantically relevant anomalies

As a third step, our algorithm applies a probabilistic active learning scheme to detect semantically relevant anomalies. Our probabilistic approach is based on a hierarchical Bayesian generative model. We use this model to estimate the probability that a given record corresponds indeed to an interesting anomaly to a human user.

Our hierarchical generative model is based on the subspaces and micro clusters obtained by the process described in the previous Section. According to our model, the generation of an anomalous records consists of two main steps. First, one of the possible subspaces is selected according to a *Multinomial* distribution. Afterwards, within the selected subspace, a micro cluster is selected using the parameters of the mixture model used to model the micro clusters inside the subspace. Figure 1 depicts a diagram of this generative process, where there are n possible subspaces and, in each subspace, there are K_{S_j} possible clusters where we can obtain an anomaly.

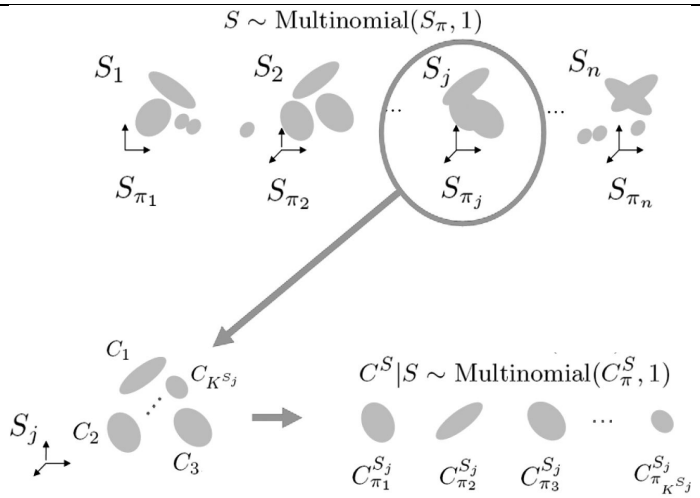

Fig. 1. Diagram of the two-step hierarchical model used to generate an anomaly.

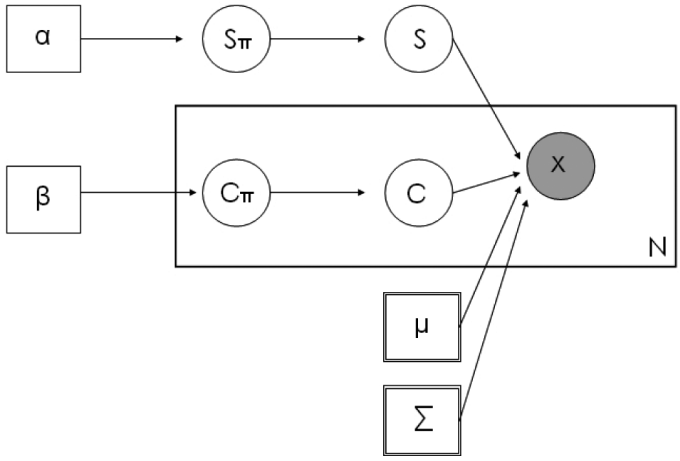

Fig. 2. Graphical model for the active learning process. Squares are model parameters, double line squares are fixed parameters, circles are latent variables, shadow circle correspond to the observed variable and the box represents replicated variables.

We move the previous hierarchical model to the Bayesian area by using *Dirichlet* distributions to model relevant priors distributions. The *Dirichlet* distribution is a multi-parameter generalization of the Beta distribution and defines a distribution over distributions, i.e. the result of sampling a *Dirichlet* is a distribution on some discrete probability space [7]. Figure 2 shows the resulting graphical model behind our Bayesian probabilistic approach. For a given data point $X \in$ FDS, variable $S \in [1 \dots n]$ represents the selected subspace where we are projecting X, in other words, the subspace from where we will search for anomalies. Variable $C^S \in [1 \dots K^S]$ represents the cluster of the mixture within the subspace S where we will sample data points. We use *Dirichlet* priors for variables $S_\pi \sim Dirichlet(\alpha)$ and $C^S_\pi \sim Dirichlet(\beta^S)$ to apply posterior updates using the conjugate prior properties between *Dirichlet* and *Multinomial* distributions. *Dirichlet* parameter $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ is initialized as a symmetrical *Dirichlet* prior ($\alpha_1 = \alpha_2 = \dots = \alpha_n$) and parameter β^S is initialized according to the weights determined in the clustering process for subspace S (section 3.2.2). Also the fixed set of parameter vectors μ^i_j and

Σ_j^i $i \in [1 \dots n]$ $j \in [1 \dots K^S]$ for each subspace are obtained from the mixture models estimated in the clustering process. Specifically, the generative process is given by:

- Choose $S_\pi \sim Dirichlet(\alpha)$
- For each $S \in [1 \dots n]$, choose $C_\pi^S \sim Dirichlet(\beta^S)$
- Generate each anomalous record x by:
  - Choose $S \sim$ Multinomial$(S_\pi, 1)$
  - Choose $C^S|S \sim$ Multinomial$(C_\pi^S, 1)$
  - Choose $x \sim Gaussian(\mu_{C^S}^S, \Sigma_{C^S}^S)$

Once a given subspace is selected, we sample an observation from the Gaussian mixture model relative to that subspace. Given that with continuous data we use Gaussian functions, it is possible that the sample from the Gaussian mixture does not correspond to the position of a real record inside the subspace. To solve this problem, we use Euclidean distance to select the record in the subspace that is closest to the sampled observation considering only the variables within the respective subspace. With categorical data we first sample the cluster using the mixture weights and then we take a sample from the selected cluster according to the initial likelihood provided by the BN.

3.4. Inference

To update the parameters of the *Dirichlet* distributions using the labels provided by an domain expert, we use the conjugacy property between *Dirichlet* and *Multinomial* distributions. Intuitively, *Dirichlet* distribution saves information about the occurrences of the possible states of a *Multinomial* distribution, in this step we define our states as the possible subspaces and our occurrences as the user feedback (anomaly or normal point). We maintain one distribution for anomaly detections (true positives) and one distribution for normal points (false positives). For anomaly detections, we consider an occurrence j of the variable S ($S = j$) if the user labels one record selected from the subspace j as an anomaly. In that case we update the *Dirichlet* parameter relative to the detections distribution. For cases where the user labels an instance as a normal point, we use a second *Dirichlet* distribution with parameter $\bar{\alpha}$ to save information about the "unsuccessful" subspaces, analogous, for this second distribution we consider an occurrence j of the variable S ($S = j$) if the user labels one record selected from the subspace j as a "normal" point.

After t iterations, the posterior parameters α and $\bar{\alpha}$ of the *Dirichlet* distribution and the second *Dirichlet* distribution is:

$$\alpha^{t+1} = (\alpha_1^t + \sum_{i=1}^{t} \delta(S = S_1), \dots, \alpha_n^t + \sum_{i=1}^{t} \delta(S = S_n)) \quad (1)$$

where $\sum_{i=1}^{t} \delta(S = S_i)$ is the number of anomalies detected in subspace S_i at iteration t.

$$\bar{\alpha}^{t+1} = (\bar{\alpha}_1^t + \sum_{i=1}^{t} \delta(S = S_1), \dots, \bar{\alpha}_n^t + \sum_{i=1}^{t} \delta(S = S_n)) \quad (2)$$

where for this case $\sum_{i=1}^{t} \delta(S = S_i)$ is the number of normal points detected in subspace S_i at iteration t.

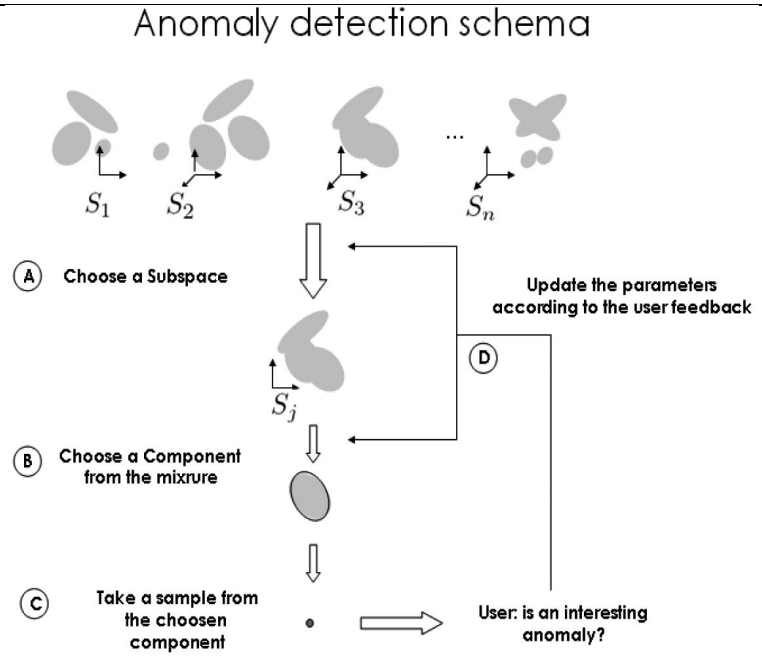

Fig. 3. Schema of the entire anomaly detection process. Different steps are detailed in sections 3.2, 3.3 and 3.4

Having updated the parameters that saves information about detections, the probability of selecting the subspace j given all the occurrences $S^1, \ldots, S^t$ is:

$$P(S^{t+1} = j | S^1, \ldots, S^t, \alpha) = \frac{\frac{\alpha_j}{\sum_{k=1}^{n} \alpha_k}}{\frac{\alpha_j}{\sum_{k=1}^{n} \alpha_k} + \frac{\bar{\alpha}_j}{\sum_{k=1}^{n} \bar{\alpha}_k}} \tag{3}$$

The update for *Dirichlet* variables C_π^S is analog.

The *Dirichlet* distribution controls the parameters of the *Multinomial* distribution over the set of subspaces or clusters. In this way, a positive feedback from the user increases the probability of selecting again a record from the same subspace or cluster.

Due to the fact that anomalies are a small fraction of the data, and we want to detect the more possible anomalies in less time, our active learning algorithm fits a probabilistic model in order to increase the likelihood of sample new interesting anomalies in future iterations. In that sense, we are finding a distribution according to user interest more than obtaining a blind distribution from the raw data without considering semantical information. In 3.4.1 we show that the 3-step sampling process converges to the user desired distribution after a finite number of iterations of the entire process.

Figure 3 shows a resume of the entire anomaly detection process. Steps A and B corresponds to the *Multinomial* selection of subspaces and clusters, step C is the data point sampling step from the selected cluster, step D corresponds to the inference depending on the user feedback.

3.4.1. Proof of convergence of the sampling process.

Let F be a *Multinomial* distribution over $G = \{g_1, g_2, \ldots, g_c\}$ possible clusters from where the algorithm will try to get interesting anomalies. $F \sim Multinomial(U, 1), U = \{u_1, u_2, \ldots, u_c\} \sum_{i=1}^{c} u_i =$

1. We want to rapidly discover the user desired distribution, giving more likelihood to clusters that user chooses as interesting. Suppose without loose of generality that interesting anomalies for the user are in cluster $g_j \quad j \in [1 \dots c]$, then we want that the distribution U converges to a *Multinomial* distribution with expected parameters $E(u_i), i \in [1 \dots c]$, such that $E(u_j) > E(u_i) \quad \forall i \neq j$.

We model U with a *Dirichlet* (α) distribution with $\alpha = \{\alpha_1, \dots, \alpha_c\}$, $\sum_{i=1}^{c} \alpha_i = 1$. The posterior probability distribution of U given $f_1, \dots, f_k$ iterations (*Multinomial* events) is:

$P(U|f_1, \dots, f_k) \sim \text{Dirichlet}(\alpha'_1, \dots, \alpha'_c)$, where $\alpha'_i = \alpha_i + \delta(f_k = i)$ and $\delta(f_k = i)$ indicates the number of occurrences for the cluster g_i.

Because the user prefers cluster g_j, exists one value for k such that there will be more occurrences for the elements in cluster g_j, then $\delta(f_k = j) > \delta(f_k = i), \forall i \in \{[1 \dots c] \setminus j\}$. Choosing that k as the minimal number of iterations we have that the expected value for u_i is $E(u_i) = \frac{\alpha_i}{\sum_{i=1}^{c} \alpha_i}$, then we have that $E(u_j) > E(u_i), \forall i \in \{[1 \dots c] \setminus j\}$.

4. Experiments

We test our algorithm under different conditions using synthetic and real datasets. Our main goals are to validate the main hipothesis behind our algorithm and to test its performance. Accordingly, we start by testing our method in a synthetic dataset that is built according to our main assumptions, i.e., anomalies belong to micro clusters located in key subspaces of the feature space. Our goal here is to use ground truth data to test if in a scenario that follows our assumptions, our method is able to achieve good performance. Unfortunately, it is not possible to count with this type if ground truth in real datasets. Afterwards, we evaluate the generality of the method using real datasets. We focus on the ability of the method to find relevant anomalies, while requesting minimum feedback from the user. Finally, we compare our active learning model against an alternative state-of-the art technique [10], showing that our approach offers several advantages for the case of semi-supervised anomaly detection problems.

4.1. Experiments on a synthetic dataset

We generate synthetic datasets by sampling from Gaussian Mixtures in some known fixed subspaces. In case of dimensions not included in the selected subspace, we generate uniformly distributed data. We add small clusters in low density areas, to simulate anomalous patterns, labeling some of them as containing interesting anomalies. Algorithm 1 shows the steps to generate artificial samples on v variables.

4.1.1. Subspace learning

To evaluate if our method effectively learns the correct subspaces, we generate a synthetic dataset following the algorithm in the previous section. We generate 100.000 records and 7 attributes, inserting 0.5% of anomalous records as small clusters in low density areas from two known subspaces with dimensions $\{1, 2\}$ and $\{4, 5, 6\}$, respectively. For subspace $\{1, 2\}$ we generate 13 components and for subspace $\{4, 5, 6\}$ 7 components. Anomalous clusters were concentrated in micro clusters with 25% of the variability presented in normal clusters (25% of the mean variability among normal clusters).

After applying the BN, all the synthetic anomalies appear among the first 10% of the elements with lowest likelihood values. In other words, all the anomalies are inside the FDS. Afterwards, by applying the CLIQUE algorithm, the method finds 4 main subspaces:

Algorithm 1 Synthetic Data generation

1: Generate n random non overlapping subspaces $S_i, i \in [1 \dots n]$ /* *Assume without loss of generality that S_i has i variables* */
2: **for** $i = 1$ to n **do**
3: Generate a random number of components $k \in [1 \dots h]$
4: **for** $c = 1$ to k **do** [/* *Generation of Gaussian Components*/]
5: Generate a random vector $\mu_c \in S_i$
6: Generate an orthonormal base of i vectors $b_1, \dots, b_i$ /* *Eigenvectors of* Σ^c */
7: Let E the matrix with columns $[b_1 b_2 \cdots b_i]$
8: Generate i random values $\lambda_1, \dots, \lambda_i$
9: Set D as a diagonal matrix where $d_{jj} = \lambda_j \quad j \in [1 \dots i]$
10: Compute a covariance matrix $\Sigma^c = EDE^T$
11: Generate a random membership vector $[w_1^i w_2^i \cdots w_c^i]$, $\sum_{j=1}^{c} w_j^i = 1$
12: **end for**
13: **for** $j = 1$ to m **do** [/* *Generation of m non anomalous samples*/]
14: Randomly select a component $b \in [1 \dots c]$ from discrete distribution $\{w_1^i, w_2^i, \dots, w_c^i\}$
15: Generate a sample s from $Gaussian(\mu_b, \Sigma^b)$
16: Set the values of x_j in the correspondent variables from S_i with the values of s
17: **end for** /* *Generate anomalies in subspace*/
18: Select a random number of anomalous clusters cf
19: **for** $j = 1$ to cf
20: Select the data point with lowest likelihood value under the GMM model
21: Set that point as a new mean vector μ_a
22: Create a new covariance matrix Σ_a following the same steps in 6–10.
23: Select a random number of anomalies z
24: Generate samples $A = \{a_1, \dots, a_z\}$ from $Gaussian(\mu_a, \Sigma_a)$
25: Insert the samples in A into the dataset and label them as anomalies
26: **end for**
27: **end for**
28: **for** $j = 1$ to v **do** [/* *Generate uniform data for the remaining variables*/]
29: **for all** generated data points x_t **do**
30: **if** $j \notin \{S_1, \dots, S_n\}$ **then**
31: Generate a uniform number u
32: Set $x_t(j) = u$
33: **end if**
34: **end for**
35: **end for**

$\{1, 2\}, \{4, 5, 6\}, \{3\}, \{7\}$. We can see that 2 of the detected subspaces match the real subspaces used to generate the anomalies. As we stated before, the active learning scheme complement the output of the subspace detection method by using feedback from a domain expert to filter irrelevant subspaces. Figure 4 shows this effect by displaying the number of records selected by our algorithm in each subspace as the number of queries to the user increases. We can see that subspaces containing relevant anomalies quickly become the most popular, while the rest of the subspaces becomes less important because true anomalies are not found there.

4.1.2. Detection of anomalous records

To evaluate the efficiency of our method to quickly detect relevant anomalies, we track the number of queries that the algorithm needs to ask to an expert in order to identify most of the anomalies. To illustrate the advantage of including our active learning step, we also consider a case without this step. In this case, the algorithm shows sequentially to the expert the records sorted in ascending order according to the likelihood values provided by the BN, not using the feedback provided from the expert to improve the model about relevance anomalies. Figure 5 shows the percentage of anomalies detected as the number

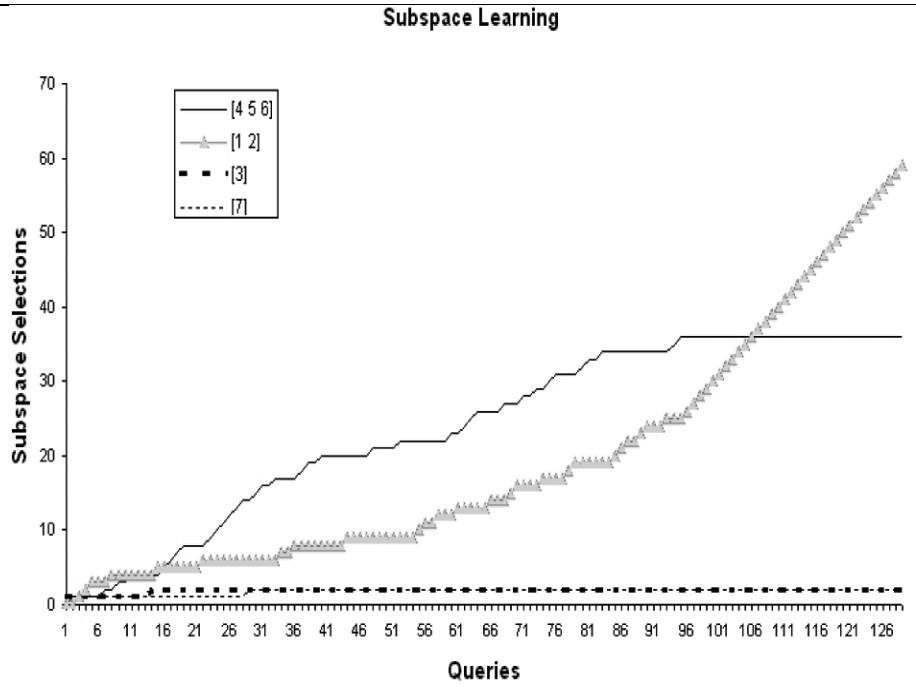

Fig. 4. Effect of active learning on subspace selection process. As the number of queries to the user increases, most of the records selected by the algorithm are coming from subspaces containing indeed relevant anomalies.

of data labeled increases for the case with and without active learning. The Figure shows that the active learning achieves its goal of accelerating the anomaly finding process, with an increment over the initial BN model of 30% in the first 2% of queries, 50% in the first 4% of queries, and 70% in the first 6% of queries.

As a further test, we also compare our approach against a recent active learning approach proposed by Cebron and Berthold [10], having good results in classification problems. This work uses an Exploration to Exploitation (EXR-EXT) approach. The exploration phase selects instances that are more representatives of data to be labeled by the user expert an included as the first set of training points for a K-NN classifier. They evaluate the representativeness with a potential function that corresponds to the sum of Gaussian Kernels between the instance and all the other remaining points in the dataset. Then the instances with highest potential values are included in the training set as more representative neighbors, they call these instances as prototypes. The Exploitation phase takes into account information of the current classifier in order to find new points that improves classification. They use weighted K-NN based on the labeled prototypes. For each new test instance they calculate a probability of membership to any of the classes depending on the distance to the prototypes and their labels. Elements with similar probability values among all classes are more uncertain about their class membership. They compute the uncertainty for a classifier as the sum of the class membership entropies among all data points, then they select data points having high uncertainty about their classification to be labeled an included as new prototypes.

To compare our approach with the active learning algorithm exposed above (EXR-EXT) we use a simulated dataset on three dimensional space generating points according to the algorithm 1. For this experiment we use four normal components an one micro cluster as anomalous (see Fig. 6). The

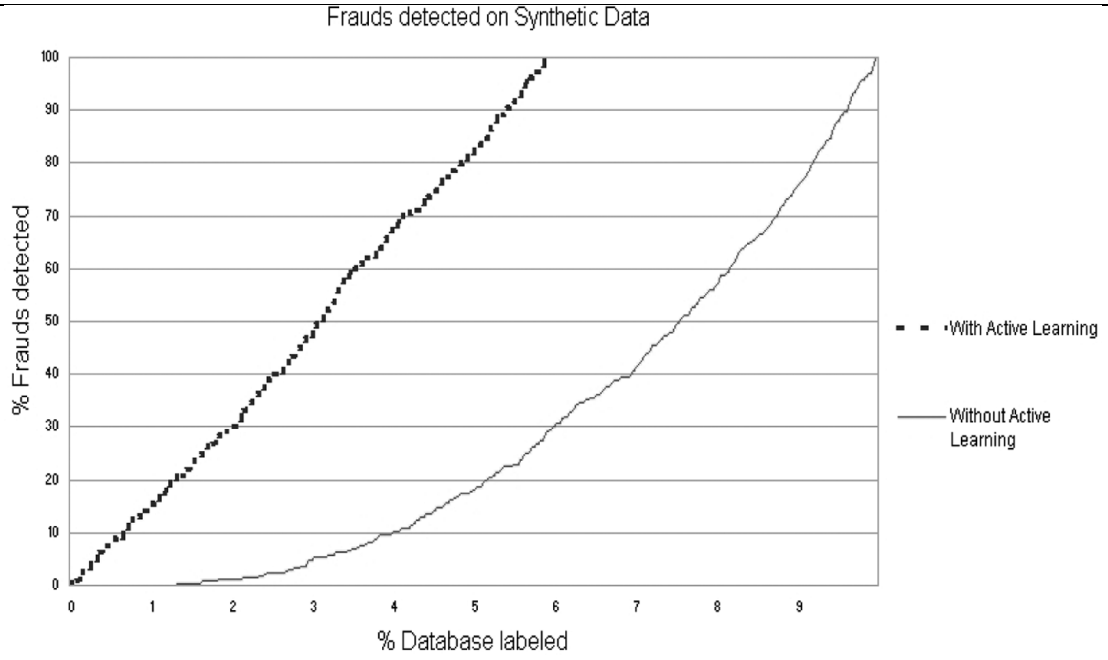

Fig. 5. Anomalies detected as percentage of data labeled increases. 100% of the anomalies are detected after labeling 6% of the total data.

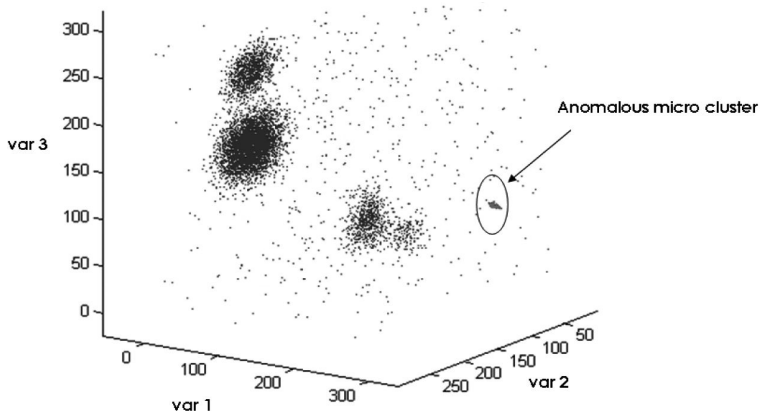

Fig. 6. Dataset used to compare the efficiency of two different active learning approaches.

main goal is to detect all the anomalies we can in less iterations of the algorithm. Figure 7 shows the classification accuracy while the number of iterations increases. We measure the accuracy as an indicator of the true positives rate penalized by the false positives rate. The EXR-EXT algorithm spend many time trying to learn the right label of the entire dataset, while our active learning scheme just refines its knowledge about the interesting points, outperforming EXR-EXT in the anomaly detection task. We can observe a drastic change of accuracy for the EXR-EXT algorithm about the iteration 17, that indicates the instance in where the algorithm performs its best fit for data, then it decrease in accuracy, because it starts to include more representative points than the algorithm needs to perform a good classification.

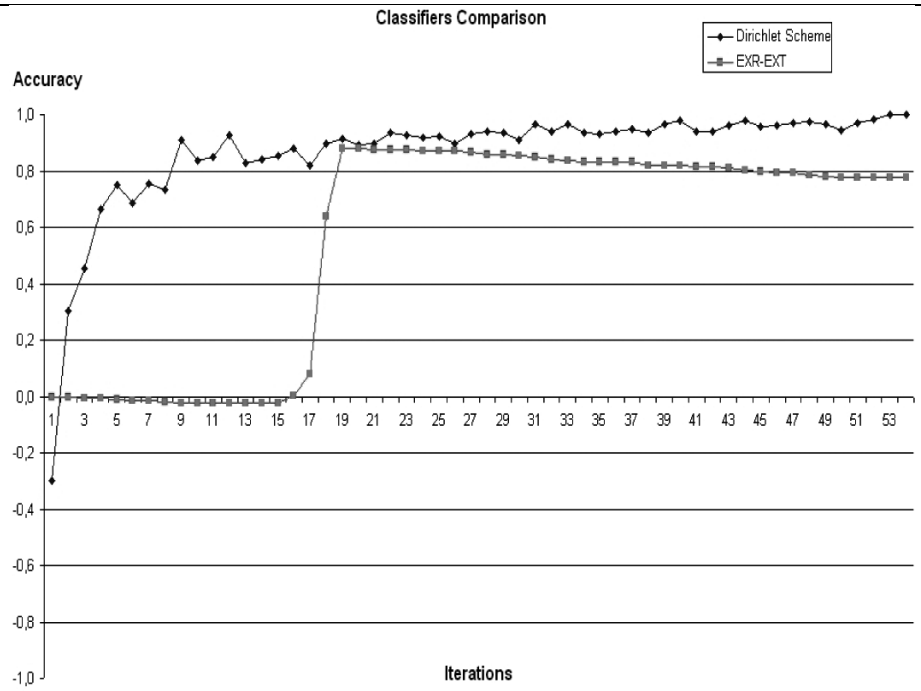

Fig. 7. Accuracy of *Dirichlet* Active Learning Scheme vs EXR-EXT algorithm. We observe that *Dirichlet* scheme starts learning in firsts iterations while EXR-EXT start increasing its accuracy sixteen iterations after *Dirichlet* model.

4.2. Experiments on real datasets

To evaluate the performance of our method in real scenarios, we test our algorithm with two real datasets. First, we use a fraud detection application using a dataset of automobile insurance company containing regular and fraudulent cases [35]. Afterward, we use a flaw detection application using a dataset containing visual attributes extracted from X-ray images of regular and faulty metallic pieces [29]. Both datasets are previously labeled by a domain expert, who adds a label indicating if a record corresponds to a regular or an anomalous case.

4.2.1. Automobile Insurance Dataset.

This dataset consists of 31 attributes and 14.400 records, where each record contains information about automobile insurance issues such as car accidents, car robbery, etc. In this dataset, there is a total of 100 records corresponding to fraudulent situations. As in the case of synthetic data, we evaluate the accuracy of the subspace learning process and the efficiency of our algorithm to identify relevant anomalies.

Subspace learning. Unfortunately in the case of real data, we do not have ground truth information about suitable subspaces to detect fraudulent records.

We run our algorithm finding 5 subspaces with high-level density clusters. Afterwards, the active learning step selects the subspace $\{19, 26, 30\}$ as the more popular, where most of the frauds are correctly detected. Figure 8 shows the result of projecting all data to the subspace $\{19, 26, 30\}$. The radius of each dot is proportional to the number of records with the corresponding values. It is possible to observe that in this subspace most of the frauds are highly atomized simplifying the identification task of the active learning approach.

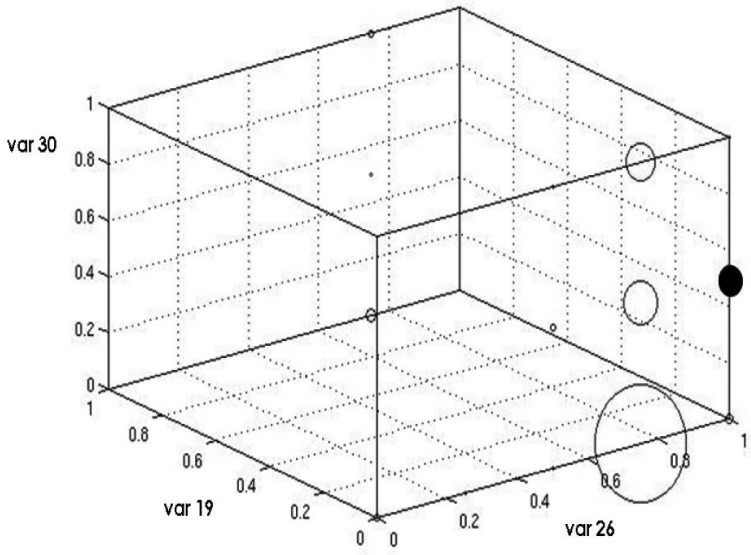

Fig. 8. Projection of the data over the most popular subspace, one of the clusters (black) is 97% composed by anomalous records.

To further investigate the relevance of detecting a suitable subspace to search for anomalies, we run the k-modes ($k = 7$) algorithm over the complete dataset under 2 conditions. First considering all the attributes in the dataset and then only the subspace detected with our active learning approach. Figure 9 shows the results. The first histogram corresponds to the case of using the entire feature space while the second shows clustering results over the subspace detected by our model. Gray columns represent percentage of anomalous points composing each cluster. The size of each column represents the percentage of the data belonging to that cluster. We use the Adjusted Rand Index (ARI) indicator [22] to evaluate the clustering quality among subspaces. Given that our goal is anomaly detection, we consider in the ARI formula only the anomalous points, avoiding problems related to the unbalanced number of instances with respect to normal data. We can see that anomalies are well concentrated in one cluster concerning the clustering process of the specific subspace detected by our model, on the other hand, running the clustering process for the same points considering all the attributes results in poor quality clusters (55% less in ARI), where anomalies are distributed over all clusters, making very hard to rapidly detect them.

Detection of frauds. We evaluate the efficiency of our method on fraud detection as the percent of data labels required to detect 100% of the frauds. Figure 10 shows percent of frauds detected as percent of data labeled increases. The initial BN filter let 100% of the anomalies in the first 13% of the elements with lowest likelihood values. We can see that the proposed algorithm is able to detect 100% of the frauds after labeling 8% of the data.

4.2.2. X-ray wheels Images Dataset.

This dataset contains 969 instances and 31 features, where each record has information about visual attributes of specific regions in metallic pieces. There a total of 100 records coming from faulty metallic pieces.

The subspace clustering algorithm found six subspaces, each one containing six dimensions. Two of the six subspaces (4 and 5) were selected with more frequency by the active learning algorithm, due to the fact that in those subspaces the algorithm founds more failures.

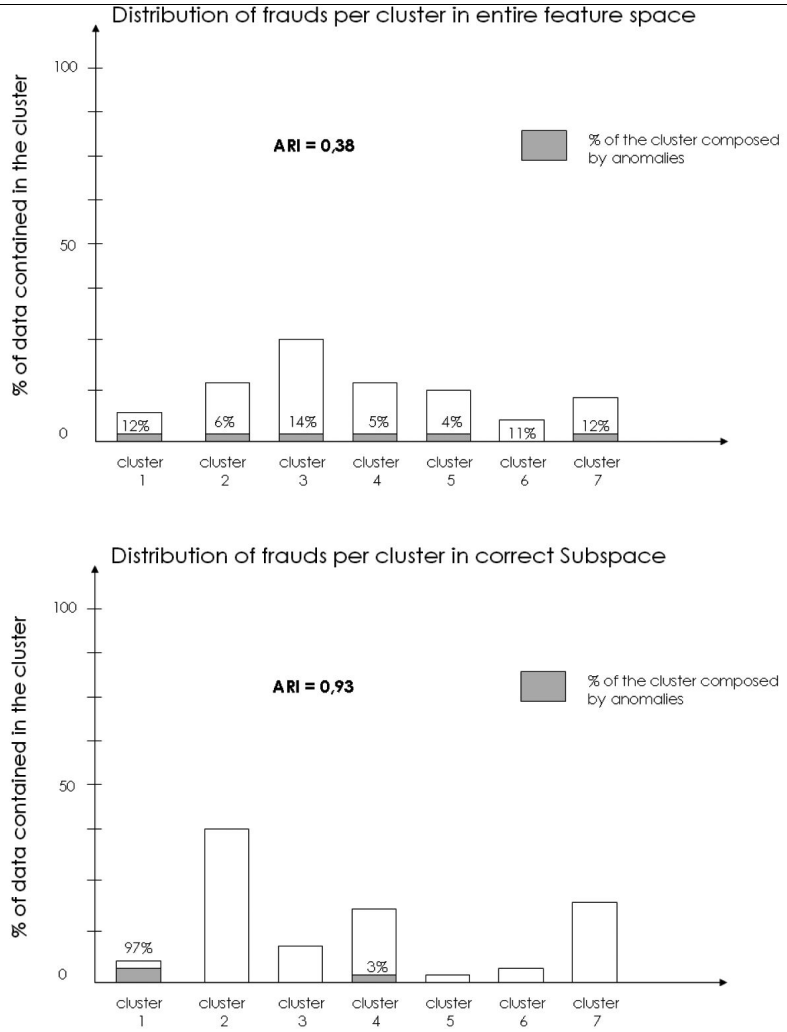

Fig. 9. Class distributions over all clusters. Anomalies are well concentrated in cluster 1 when the clustering process run over the correct subspace. When the clustering process run considering all variables, anomalies are dispersed over all clusters making very hard to detect them fast.

Detection of image failures. We evaluate the efficiency of our method on image failure detection as the percent of data labels required to detect 100% of the faults. Figure 11 shows percent of faults detected as percent of labeled data increases. The initial BN filter let 100% of the faults in the first 58% of the elements with lowest likelihood values. The active learning process detected 100% of the failures using 46% of data as labeling information, reducing in 12 percentual points the number of hints required. Is important to note that this dataset has less instances than the Automobile insurance dataset, then the percent of labeled data required is higher because is a relative indicator, in this case we need 46% of labeled instances, that corresponds to 440 instances to detect 100 anomalies, in the automobile case we detect 100 frauds using 1150 labeled instances, that corresponds only to 8% of the dataset. For the same reason in this case we use $\tau = 0.5$.

Importance of Subspace Clustering. We show the clusters composition on subspaces with higher levels of failure detection. Figure 12 shows the clusters composition for Subspaces 4 and 5 and their

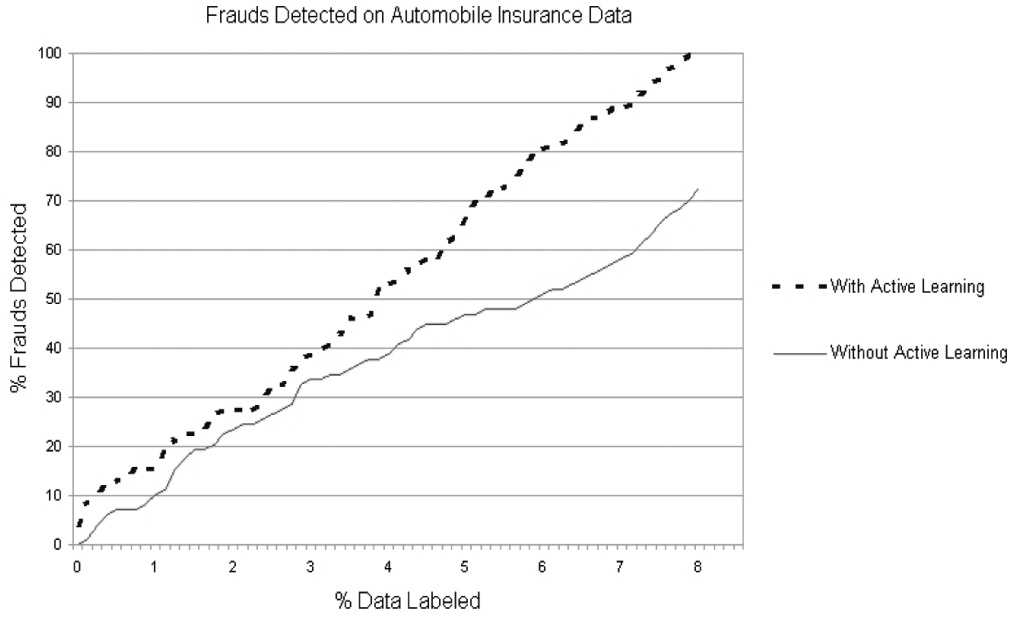

Fig. 10. Frauds detected on an automobile insurance dataset. 100% of frauds are detected after labeling 8% of the dataset.

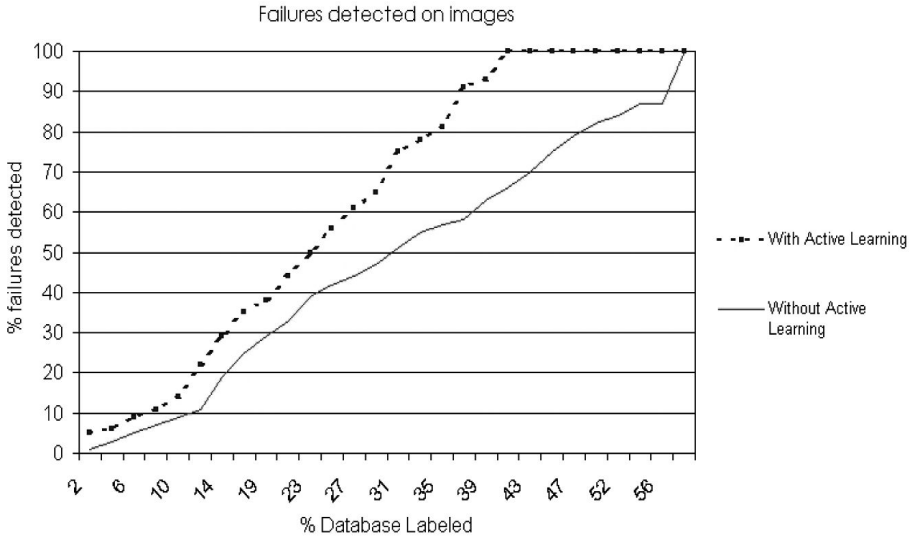

Fig. 11. Failures detected on x-ray wheel images dataset. The active learning schemme improves the BN detection in 12 percentual points.

respective ARI values. The size of columns represent the percent of the data (after the BN filter), the gray segment represents the percent of data for each cluster corresponding to image failures. All the failures are concentrated in just two clusters in subspace 5 and mostly in one cluster in subspace 4. That concentration plus the weighted clustering process improves detection accuracy.

Figure 13 shows the distribution of clusters and anomalies considering the entire feature space. With many unnecessary variables the clustering process result in more clusters with failure records dispersed

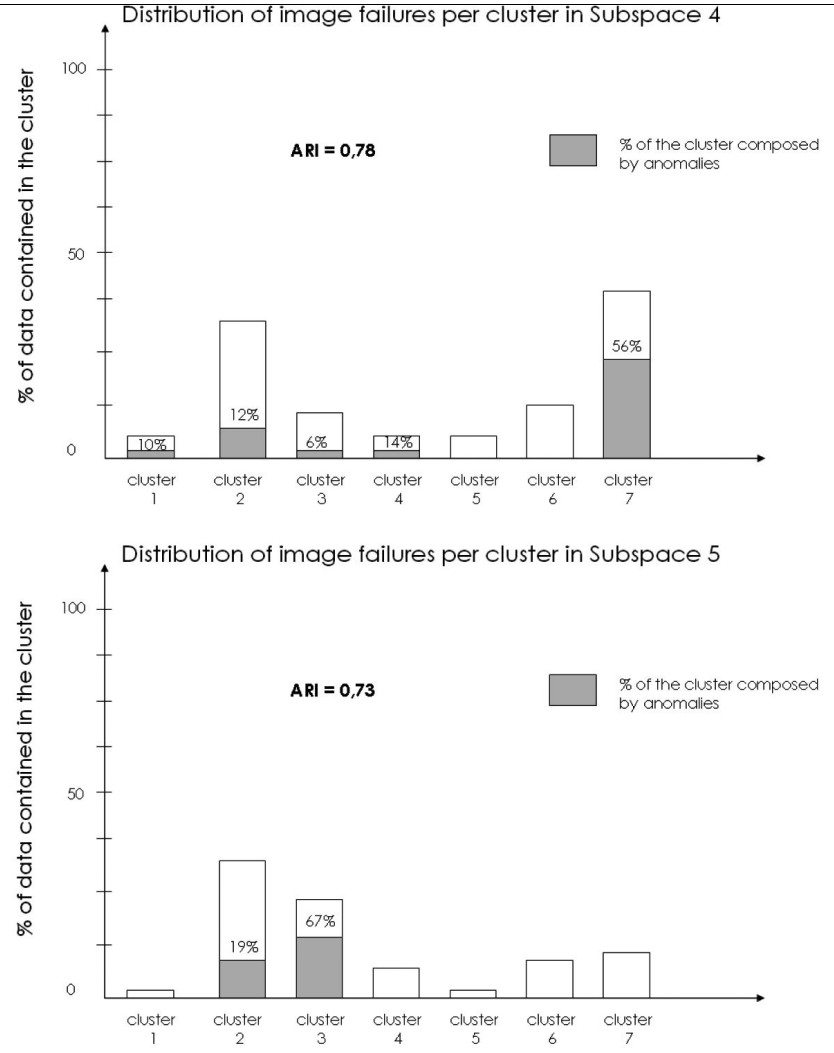

Fig. 12. Percent of failures over all clusters. The size of each column indicates the percent of total data included in that cluster. The gray part indicates for each cluster, which percent of the points belonging to it corresponds to anomalies. Anomalies are well concentrated in cluster 7 in Subspace 4 and clusters 2 and 3 in Subspace 5.

among many clusters, in this case the ARI indicator decreases in 51 percentual points with respect to subspace 4 and 46 percentual points with respect to subspace 5, making very difficult to detect the failures in an efficient way.

5. Discussion

This work contributes with an algorithm based on an active learning scheme that tackles the problem of detecting anomalous records in large datasets. Using an initial filtering stage provided by a BN, subspace clustering techniques, and properties of *Dirichlet* distributions, we are able to effectively use feedback from an expert to speed up the selection of relevant anomalies that exhibit regularities or

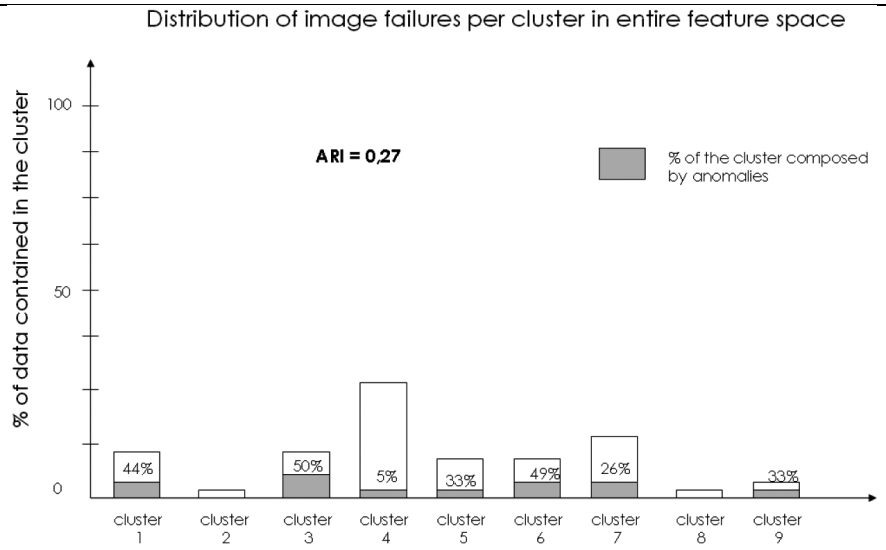

Fig. 13. When the clustering process run considering all variables, clusters and image failures are dispersed making very hard to detect them fast.

patterns in selective subspaces. The active learning scheme proposed in this work is different concerning the function we are trying to optimize. While classical active learning approaches maximizes the class boundaries improving a supervised classifier, we maximize the number of detections over the number of hints required from the user. In addition, this paper contributes with a new point of view of the anomaly detection problem, showing that although high dimensional datasets make more difficult detecting anomalies, searching in some selective subspaces and selective groups of data initially filtered by the BN model, we can find that anomalies are distributed in isolated micro clusters making possible to detect them. Our results show that our approach is able to identify relevant subspaces and also to significantly decreases the time to reach relevant anomalies. Moreover we show the relevance of work with some selective subspaces instead of the entire set of features to search for anomalies. Given that the expert time is usually the most valuable resource in user-computer interaction systems and that information is constantly changing, we believe that tools as the one presented here may be of great help as a filtering step to guide the search for anomalies in cases where an exhaustive analysis is not possible. Furthermore, by providing a set of specific attributes corresponding to the subspace used to detect an anomaly, the method proposed here can also provide an explanation of the main sources of a detected anomaly.

It is important to note that given the running time of the preprocessing steps of our method (like the BN step and subspace clustering step), our algorithm is not suitable for datasets with thousands of variables, like some biological cases, we are working in some methods to improve the running time of these steps to allow our method work with this kind of data.

References

[1] N. Abe, B. Zadrozny and J. Langford, Outlier detection by active learning, In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2006, pp. 504–509, ACM.

[2] D. Agarwal, An empirical bayes approach to detect anomalies in dynamic multidimensional arrays, in: *Proceedings of the 5th IEEE International Conference on Data Mining. IEEE Computer Society*, 2005, pp. 26–33.

[3] R. Agrawal, J. Gehrke, D. Gunopulos and P. Raghavan, Automatic subspace clustering of high dimensional data for data mining applications, in: *Proceedings ACM-SIGMOD International Conference Management of Data*, 1998, pp. 94–105.
[4] A. Arning, R. Agrawal and P. Raghavan, A linear method for deviation detection in large databases. in: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 164–169.
[5] A. Bianco, M. Ben, E. Martinez and V. Yohai, Outlier detection in regression models with arima errors using robust estimates, *Journal of Forecasting* **20**(8) (2001), 565–579.
[6] C.M. Bishop, Novelty detection and neural network validation, in: *Proceedings of IEE Conference on Vision, Image and Signal Processing*, 1994, pp. 217–222.
[7] D. Blackwell and J. MacQueen, Ferguson distribution via polya urn schemes, *The Annals of Statistics* **1**(2) (1973), 353–355.
[8] M. Breunig, H. Kriegel, R. Ng and J. Sander, Lof: identifying density-based local outliers, in: *Proceedings of 2000 ACM SIGMOD International Conference on Management of Data*, 2000, pp. 93–104.
[9] A. Cansado and A. Soto, Unsupervised anomaly detection in large databases using bayesian networks, *Applied Artificial Intelligence* **22**(4) (2008), 309–330.
[10] N. Cebron and M. Berthold, Active learning for object classification: from exploration to exploitation, *Data Mining and Knowledge Discovery* **18**(2) (2008), 283–299.
[11] D. Chen, X. Shao, B. Hu and Q. Su, Simultaneous wavelength selection and outlier detection in multivariate regression of near-infrared spectra, *Analytical Sciences* **21**(2) (2005), 161–167.
[12] S. Djorgovski, A. Mahabal, R. Brunner, R. Gal, S. Castro, R. de Carvalho and S. Odewahn, Searches for rare and new types of objects, in: *Virtual Observatories of the Future, ASP Conference Proceedings*, (Vol. 225), 2001, pp. 52–63.
[13] E. Eskin, Anomaly detection over noisy data using learned probability distributions, in: *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000, pp. 255–262.
[14] M. Ester, H. Kriegel and X. Xu, A density based algorithm for discovering clusters in large spatial databases with noise, in: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.
[15] T. Fawcett and F. Provost, in: *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999, p. 5362.
[16] T. Ferguson, A bayesian analysis of some nonparametric problems, *The Annals of Statistics* **1**(2) (1973), 209–230.
[17] N. Friedman, I. Nachman and D. Peér, Learning Bayesian network structure from massive datasets: The Sparse Candidate algorithm, in: *Proceedings of 15th Conference on Uncertainty in Artificial Intelligence*, 1999, pp. 206–215.
[18] Grubb and E. Frank, Procedures for detecting outlying observations in samples, *Technometrics* **11**(1) (1969), 1–21.
[19] J. He and J. Carbonell, Nearest-neighbor-based active learning for rare category detection, in: *Advances in Neural Information Processing Systems*, (vol. 20), 2007.
[20] V. Hodge and J. Austin, A survey of outlier detection methodologies, *Artificial Intelligence Review* **22**(2) (2004), 85–126.
[21] Z. Huang, Extensions to the k-means algorithm for clustering large data sets with categorical values, *Data Mining and Knowledge Discovery* **2** (1998), 283–304.
[22] L. Hubert and P. Arabie, Comparing partitions, *Journal of classification* **2**(1) (1985), 193–218.
[23] W. Jin, A. Tung and J. Han, Mining top-n local outliers in large databases, in: *Proceedings of the seventh ACM SIGKDD International Conference on Knowledge discovery and data mining*, 2001, pp. 293–298.
[24] G. John, Robust decision trees: Removing outliers from databases, in: *Proceedings of the first International Conference on Knowledge Discovery and Data Mining*, 1995, pp. 174–179.
[25] E. Knorr and R. Ng, Algorithms for mining distance-based outliers in large datasets, in: *Proceedings of the VLDB Conference, New York, USA*, 1998, pp. 392–403.
[26] Y. Kou, C. Lu, S. Sirwongwattana and Y. Huang, Survey of fraud detection techniques, in: *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, 2004, pp. 749–754.
[27] T. Lane and C. Brodley, Applications of machine learning to anomaly detection, in: *Applications of Artificial Intelligence in Engineering*, R. Adey, G. Rzevski and T. Teti, eds, Southampton, UK: Comput. Mech. Publications., 1997, p. 11314.
[28] D. Lewis and W. Gale. A sequential algorithm for training text classifiers, in: *Proceedings of 17th International Conference ACM SIGIR*, 1994, pp. 3–12.
[29] D. Mery and D. Filbert, Automated flaw detection in aluminum castings based on the tracking of potential defects in a radioscopic image sequence, *IEEE Transactions on Robotics and Automation* **18**(6) (2002), 890–901.
[30] A. Nairac, N. Towsend, R. Carr, S. King, P. Cowley and L. Tarassenko, A system for the analisys of jet system vibration data, *Integrated ComputerAided Engineering* **6**(1) (1999), 53–65.
[31] R. Neapolitan, *Learning Bayesian Networks*, Prentice Hall, 2004.
[32] S. Papadimitriou, H. Kitagawa, P. Gibbons and C. Faloutsos, Loci: Fast outlier detection using the local correlation integral. Technical Report IRP-TR-02-09, Intel Research Laboratory, Pittsburgh, PA, 2002.
[33] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, 1988.
[34] D. Pelleg and A. Moore, Active learning for anomaly and rare-category detection, in: *Proceedings of the 18th Conference on Advances in Neural Information Processing Systems, NIPS*, 2004.

[35] C. Phua, D. Alahakoon and V. Lee, Minority report in fraud detection: classification of skewed data, *SIGKDD Explor Newsl* **6**(1) (2004), 50–59.
[36] S. Ramaswamy, R. Rastogi and K. Shim, Efficient algorithms for mining outliers from large data sets, in: *Proceedings of the ACM SIGMOD Conference on Management of Data, Dallas, TX*, 2000, pp. 427–438.
[37] N. Roy and A. McCallum, Toward optimal active learning through sampling estimation of error reduction, in: *Proceedings of 18th International Conference on Machine Learning, ICML*, 2001, pp. 441–448.
[38] T. Seidl, E. Müller, I. Assent and U. Steinhausen, Outlier detection and ranking based on subspace clustering, in: *Uncertainty Management in Information Systems*, 2009.
[39] S. Seung, M. Opper and H. Sompolinski, Query by committee, in: *Proceedings of 5th Annual ACM Workshop on Computational Learning Theory*, 1992, pp. 287–294.
[40] Ch. Son, S. Cho and J. Yoo, Volume traffic anomaly detection using hierarchical clustering, in: *Management Enabling the Future Internet for Changing Business and New Computing Services*, (vol. 5787), Springer Berlin, Heidelberg, 2009, pp. 291–300.
[41] A. Soto, A. Cansado and F. Zavala, Detection of rare objects in massive astronomical datasets using innovative knowledge discovery technology, in: *Astronomical Data Analysis Software and Systems XIV ASP Conference Series*, (Vol. 347), 2005, pp. 66–71.
[42] A. Soto, F. Zavala and A. Araneda, An accelerated algorithm for density estimation in large databases using Gaussian mixtures, *Cybernetics and Systems* **38**(2) (2007), 123–139.
[43] C. Surace and K. Worden, A novelty detection method to diagnose damage in structures: an application to an off-shore platform, in: *Proceedings of Eighth International Conference of Off-shore and Polar Engineering*, 1998, pp. 64–70.
[44] S. Tong and D. Koller, Active learning for parameter estimation in bayesian networks, in: *Proceedings of the 13th Conference on Advances in Neural Information Processing Systems, NIPS*, 2001, pp. 647–653.
[45] Arindam Banerjee Varun Chandola and Vipin Kumar, Anomaly detection: A survey, *ACM Computing Surveys* **41**(3), 2009.
[46] H. Yang, F. Xie and Y. Lu, Clustering and classification based anomaly detection, in: *Fuzzy Systems and Knowledge Discovery*, (vol. 4223), 2006, pp. 1082–1091.
[47] N. Ye and Q. Chen, An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems, *Quality and Reliability Engineering International* **17** (2001), 105–112.
[48] C. Zhang and T. Chen, An active learning framework for content-based information retrieval, *IEEE Transactions on Multimedia* **4** (2002), 260–268.
[49] T. Zhang, R. Ramakrishnan and M. Livny, Birch: An efficient data clustering method for very large databases, in: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1996, pp. 103–114.