Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection

Article *in* Information Sciences · May 2019

DOI: 10.1016/j.ins.2019.05.042

6 authors, including:

Fabrizio Carcillo
Université Libre de Bruxelles
7 PUBLICATIONS **1,243** CITATIONS
SEE PROFILE

Yann-Aël Le Borgne
Université Libre de Bruxelles
77 PUBLICATIONS **2,167** CITATIONS
SEE PROFILE

Olivier Caelen
Worldline
62 PUBLICATIONS **3,111** CITATIONS
SEE PROFILE

Gianluca Bontempi
Université Libre de Bruxelles
369 PUBLICATIONS **17,095** CITATIONS
SEE PROFILE

Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection

Fabrizio Carcillo[a], Yann-Aël Le Borgne[a], Olivier Caelen[b], Yacine Kessaci[b], Frédéric Oblé[b], Gianluca Bontempi[a]

[a]*Machine Learning Group, Computer Science Department, Faculty of Sciences ULB, Université Libre de Bruxelles, Brussels, Belgium.*
(email: {fcarcill, yleborgn, gbonte}@ulb.ac.be)
[b]*R&D Worldline, Worldline, France.*
(email: { olivier.caelen,yacine.kessaci,frederic.oble} @worldline.com).

Abstract

Supervised learning techniques are widely employed in credit card fraud detection, as they make use of the assumption that fraudulent patterns can be learned from an analysis of past transactions. The task becomes challenging, however, when it has to take account of changes in customer behavior and fraudsters' ability to invent novel fraud patterns. In this context, unsupervised learning techniques can help the fraud detection systems to find anomalies. In this paper we present a hybrid technique that combines supervised and unsupervised techniques to improve the fraud detection accuracy. Unsupervised outlier scores, computed at different levels of granularity, are compared and tested on a real, annotated, credit card fraud detection dataset. Experimental results show that the combination is efficient and does indeed improve the accuracy of the detection.

Keywords: Fraud Detection, Ensemble Learning, Outlier Detection, Semi-supervised Learning, Contextual Outlier Detection

1. Introduction

Credit card fraud detection aims to decide whether or not a transaction is fraudulent on the basis of historical data. The decision is notoriously difficult because of changes in customer spending behaviors, for example during holiday periods, and in fraudsters' own techniques, particularly those

that they use to adapt to fraud detection techniques. It is now known that machine learning techniques offer an effective approach to tackling problems like these [12].

A typical Fraud Detection System (FDS) includes multiple layers of control, each of which can either be automated or supervised by humans [6, 11]. Part of the automated layer embraces machine learning algorithms that build predictive models based on annotated transactions. In the last decade, intensive machine learning research for credit card fraud detection has led to the development of supervised, unsupervised, and semi-supervised techniques [27, 28]. In our previous works on credit card fraud detection, we investigated supervised [6, 11], unsupervised [8], and semi-supervised techniques [8, 7].

Supervised techniques rely on the set of past transactions for which the *label* (also referred to as *outcome* or *class*) of the transaction is known. In credit card fraud detection problems, the label is either *genuine* (the transaction was made by the cardholder) or *fraudulent* (the transaction was made by a fraudster). The label is usually known *a posteriori*, either because a customer complained or as a result of an investigation by the credit card company. Supervised techniques make use of labeled past transactions to learn a fraud prediction model, which returns, for any new transaction, the probability of it being a fraud. However, not all labels are available immediately [6, 13].

Unsupervised outlier detection techniques do not require knowledge of the label of transactions, and aim at characterizing the data distribution of transactions. They rely on the assumption that outliers of the transaction distribution are frauds; they can therefore be used to detect unseen types of frauds because they do not rely on transactions labeled fraudulent in the past. It is worth noting that their use also extends to clustering and compression algorithms [10]. Clustering allows the identification of separate data distributions for which different predictive models should be used, while compression reduces the dimensionality of the learning problem; both algorithms tend to improve the performances of supervised techniques.

The two approaches are complementary: supervised techniques learn from past fraudulent behaviors, while unsupervised techniques target the detection of new types of fraud. These two complementary approaches are combined in the semi-supervised techniques [8, 37] often used when there are many unlabeled data points and few labeled ones. They aim to perform better than a supervised model that uses only the dataset of the few available labeled

data points, or an unsupervised model that does not profit from the few labels.

This paper concerns the integration of unsupervised techniques with supervised credit card fraud detection classifiers.

In particular we present a number of criteria to compute outlier scores at different levels of granularity (from *high granular* card specific to *low granular* global outlier scores) and we assess their added value in terms of accuracy once integrated as features in a supervised learning strategy. As discussed in Section 2 the combination of unsupervised and supervised learning is not new in literature. In particular, our global approach, as we refer to it, is inspired by the *best-of-both-worlds* principle proposed by Michenkova et al. in [24]. What is original in this paper is the adoption of this principle in a credit card fraud detection setting and specifically the design and assessment of several outlier scores adapted to the specific nature of our problem. Section 3 introduces the standard unsupervised outlier scores used in the experimental section (Section 3.1), three original approaches to consider different levels of granularity when computing the outlier scores (Section 3.2), and the metrics used to compare the different approaches (Section 3.3). The experimental comparison is performed in Section 4, while the discussion and conclusion are presented in Section 5 and Section 6.

2. The state-of-the-art

The use of ensemble learning is popular in the supervised learning community for such techniques as boosting [15], bagging [4]; it is also common in unsupervised outlier detection, where ensemble strategies improve the estimation of the outlier scores [25, 38]. Sequential [26] and parallel [32] ensemble strategies have been also proposed to combine supervised and unsupervised outlier-detection algorithms.

The integration of supervised and unsupervised techniques has already been discussed in the literature of fraud detection. In [32], Veeramachaneni et al. introduce the AI^2 system that concatenates results from the anomaly detection approach with those from the supervised learning approach. The [32] analysis begins with the concurrent use concurrently of a supervised model (Random Forest) and an ensemble of unsupervised models. The results from these models are then merged by selecting the top $\frac{n}{2}$ results from the supervised model and the top $\frac{n}{2}$ results from the unsupervised ensemble. Note that this method requires a strategy to combine the scores deriving

from different outlier detection methods and to manage the observations common to both subsets ($\frac{n}{2}$ unsupervised and $\frac{n}{2}$ supervised outputs). To tackle this issue, the authors propose to project the different scores in the same space, for example by normalizing the scores in the $[0, 1]$ interval.

In [35], Yamanishi and Takeuchi developed a two-stage online outlier detection algorithm based on unsupervised learning. In the first step, the algorithm trains a Gaussian mixture model to score an unsupervised dataset and imputes it by giving positive labels to highly scored data. In the second step, the labeled dataset is used to learn a supervised outlier detector.

The *best-of-both-worlds* principle is a sequential approach proposed by Michenkova et al. in [24]. This team applied multiple unsupervised outlier detection algorithms to transform an initial dataset using a collection of outlier scores. This sequential approach includes unsupervised learning in the first stage and supervised learning in the second. The outlier score vector s^o, obtained by the unsupervised model over the original dataset DS, is used to augment DS: $DS^{'} = (DS, s^o)$. Next, the team compared the results in terms of AUC-ROC using a logistic regression model in three different settings: *original dataset alone*, *outlier scores alone*, and *original dataset + outlier scores*. Using two datasets, they showed that the classifier improves its accuracy when it uses outlier scores in addition to standard features. The goal of adding multiple outlier scores to standard features is to highlight the different aspects of feature space outlierness. The key advantage of this approach is that through it, we do not need to normalize or combine scores generated by heterogeneous methods. Furthermore, the supervised method is expected to automatically extract information from these scores.

Recently, a class of outlier detection algorithms emerged called *contextual outlier detection* [10, 22, 29]. This class of algorithms aims to find outliers given a *context*. A *context* is a subset of the original dataset, and it is usually identified by one or more *contextual attributes*. Besides *contextual attributes*, there are *behavioral attributes* which are used to identify the outlier score for each instance. Two instances with exactly the same *behavioral attributes* but that are defined in two different *contexts* may be identified as outlier and inlier instances, respectively.

3. Our Approach

Given the nature of the fraud detection problem, particularly the *one-to-many* relationship between cards and transactions [8, 7], we propose an

extension of the *best-of-both-worlds* principle introduced by Michenkova et al. in [24]. This extension consists of the definition of a number of outlier scores (Section 3.1) with consideration to different levels of granularity (Section 3.2) and their integration into the supervised approach.

3.1. Outlier scores

An outlier score vector can be generated using different unsupervised techniques. In this section, we introduce the outlier scores used in our experiments: *Z-score*, *PC-1*, *PCA-RE-1*, *IF*, and *GM-1*.

Given a dataset X of f features and N observations the multivariate *Z-score* of a vector $x \in \Re^f$ is

$$\sum_{i=1}^{f} \left(\frac{x_i - \hat{\mu}_i}{\hat{\sigma}_i} \right)^2$$

where $\hat{\mu}_i$ and $\hat{\sigma}_i$ are the sample mean and standard deviation of the ith feature, respectively.

Principal Component Analysis (PCA) is another well known technique for outlier detection [19, 34] which works by transforming the original dataset X (after normalization) into a set $T = XW$ of f linearly uncorrelated variables called principal components. The matrix W is squared and its ith column W_i is the ith eigenvector of $X^T X$. We consider two scores of a vector $x \in \Re^f$ based on PCA: the first is the value of the first component

$$\textit{PC-1} = W_1^T x$$

and the second

$$\textit{PCA-RE-1} = \|x - W_1 W_1^T x\|$$

is the reconstruction error obtained by using the first principal component.

Variations of these two scores are denoted by changing the suffix of the score name. So, *PC-2* will denote the second principal component, and *PCA-RE-2* will be the reconstruction error in the case of values reconstructed using the first two principal components together.

The score *IF* is based on Isolation Forest [23], and uses the length of the path between the root and the leaves of a random forest [5] as an indicator of outlierness. Finally, *GM-m* is given by the density in x of a Gaussian Mixture (GM) model fit to the dataset, where the suffix m denotes the number of mixtures.

3.2. Global, local, and cluster granularity

The approach described by Michenkova *et al.* in [24] for augmenting the dataset DS takes outlier scores computed on the whole DS set into consideration. As cardholder behaviors are very diverse, computing outlier scores in a global fashion may be a sub-optimal solution. Based on *contextual* outlier detection, this section proposes three main approaches to defining *contexts* and computing outlier scores at different levels of granularity:

1. Global granularity: All transactions are considered to be samples of a unique global distribution for which outlier scores can be computed. A transaction is considered anomalous if it lies outside the overall multivariate pattern of the entire set of transactions. This approach is the closest to the one discussed in [24], since no specificity relating to the credit-card problem (e.g., the fact that transactions belong to different customers) is taken into account.
2. Local granularity: The computation of the outlier scores is completed in a card-based manner, and a transaction is considered anomalous only if it abnormally differs from past transactions carried out by the same card.
3. Cluster granularity: This is a compromise between the two previous approaches. This method's rationale is that both the global and the card-based approaches have intrinsic limitations. It is not realistic to think that all genuine cardholders behave in the same manner. In statistical terms, this means that a global approach leads to a biased estimator of the distribution. At the same time, however, few historical examples are available at the card level with negative impact on the quality (i.e. large variance) of the related estimation. The cluster approach aims to select the optimal aggregation level at which a reasonable trade-off between the bias and the variance of the two extreme approaches can be reached. Clustering is completed at the card level, and it is based on a set of features which describe customer behavior, such as the amount of money spent over the last 24 hours.

Figure 1 is an illustrative example demonstrating the three approaches. In this example, we consider the amount of money involved in a transaction to calculate the outlier score. Our goal is to detect the most suspicious transaction or the most extreme transaction by considering only the amount of money spent in each transaction.

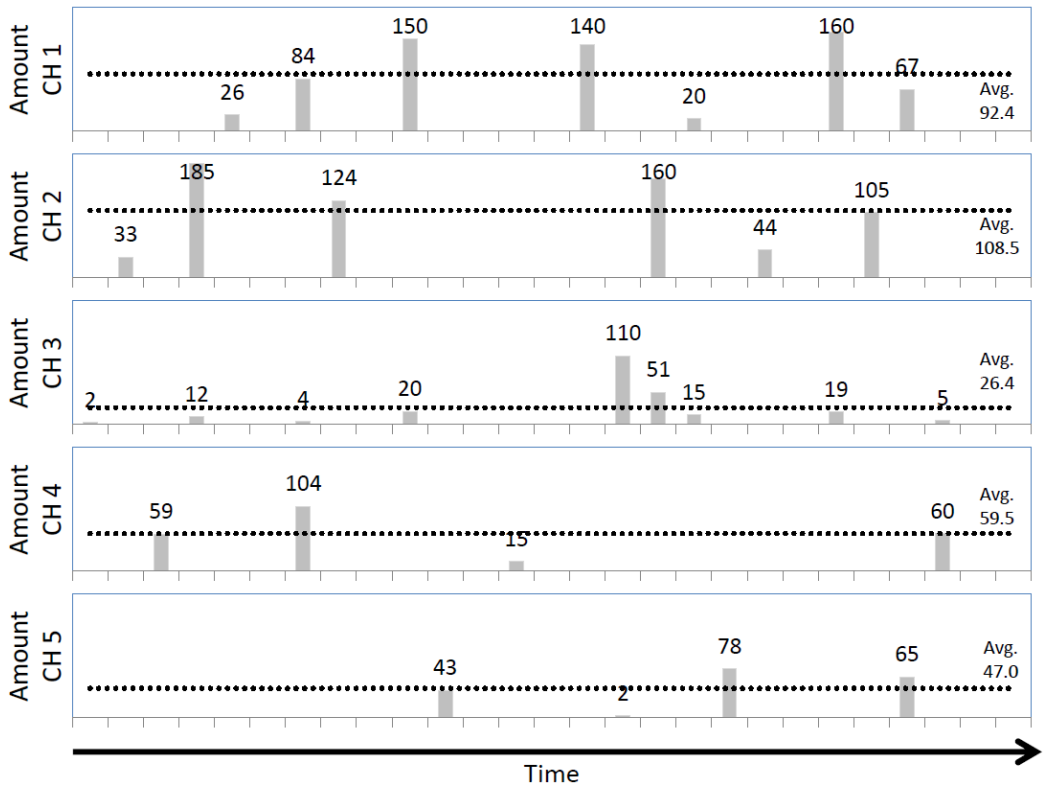


Figure 1: This illustrative example presents the behaviors of five cardholders (CHi, $i = 1, \ldots, 5$) over a fixed period of time. The dotted line represents average expenditure, and the bars represent the amount of money spent in each transaction.

From a global perspective, to detect the most suspicious transaction in a transaction history, we have to consider the average amount of money that is spent by the cardholders throughout their transactions (65.4 in the example).

The highest value recorded for the cardholder CH2 (185) is the most divergent value with respect to the average amount this cardholder spends. Accordingly, the cardholder CH2 receives an alert regarding this transaction.

In the local approach case, a cardholder's most suspicious transaction is determined according to the difference between individual transaction values and the average amount of money spent by a cardholder in their transaction history. In this example, cardholder CH3 is alerted, since of all the cardholders, the greatest difference is detected between one of their transaction values (110) and their average transaction amount (26.4).

Let us now cluster the cardholders in two groups based on their average spending amounts. The first cluster will contain cardholders with high average expenditures (CH1 and CH2), and the second will contain cardholders with low average expenditures (CH3, CH4, and CH5). The average transaction amounts are 99.8 in the high expenditure group and 39.1 in the low expenditure group. In this case, the cardholder CH2 is again alerted, since his transaction in the amount of 185 diverges the most from his cluster's average amount (99.8).

While running the global and local approaches is fairly straightforward, the cluster approach requires that some elements, such as the clustering algorithm and the features used in the cluster metric, be set. We choose to use the *k-means* algorithm [18], as it is simple to interpret, it runs quickly on large datasets, and it offers us the ability to decide a priori an arbitrary number of clusters to be identified (allowing us to easily control the aggregation level).

The matter of choosing the features on which clustering will be performed (i.e. the *contextual attributes*) is not trivial; this decision can have a major impact on the final accuracy. We considered two different sets of features: the first describes the cardholder's behavior and the second summarizes the cardholder's personal data. In the first case, we examined cardholders' average transaction expenditures and their total numbers of transactions over the last 24 hours. In the second case we consider the age, the nationality[1], and the gender of the cardholder. Since considering cardholder behavior leads to better accuracy, we only present clusters created according to cardholder behavior features in the experimental part. The principal hyper-parameter of the *k-means* algorithm is the k number of clusters to be created. This hyper-parameter is also important for our case study, as it defines the outlier score's level of granularity. In our experiments, we allow the hyper-parameter to vary from a minimum low granularity of 10 clusters to a maximum high granularity of 5000 clusters.

Algorithm 1 defines the process of outlier score construction at different levels of granularity. The function SCORE (row 1) receives a training set, a test set, and a list of outlier models to be computed as input. The output is the training and test sets augmented with related outlier scores. When

[1]Note that nationality is not encoded as a categorical variable but as a continuous variable given by the a priori risk of fraud associated to the nationality, estimated by the conditional frequency of the training set.

Algorithm 1 Outlier scores at different levels of granularity

Require: gr ▷ granularity: global, local, or cluster
Require: k ▷ number of clusters, if gr == cluster
Require: $cardUsage$ ▷ statistics on the card usage, if gr == cluster
Require: ot ▷ outlier models
Require: D_{tr} ▷ training set
Require: D_{te} ▷ test set

function SCORE($subD_{tr}, subD_{te}, ot$)
    $subOutD_{tr} \leftarrow subD_{tr}$
    $subOutD_{te} \leftarrow subD_{te}$
    for t **in** ot **do**
        $outlierModel \leftarrow$ fit t to $subD_{tr}$
        $trainingScore \leftarrow$ get score of $subD_{tr}$ using $outlierModel$
        $testScore \leftarrow$ get score of $subD_{te}$ using $outlierModel$
        $subOutD_{tr} \leftarrow$ append $trainingScore$ to $subOutD_{tr}$
        $subOutD_{te} \leftarrow$ append $testScroe$ to $subOutD_{te}$
    end for
    return $subOutD_{tr}$, $subOutD_{te}$
end function

if ($gr ==$ "$global$") **then** ▷ global granularity
    $(DOut_{tr}, DOut_{te}) \leftarrow$ SCORE(D_{tr}, D_{te}, ot)
end if
if ($gr ==$ "$local$") **then** ▷ local granularity
    $DOut_{tr}, DOut_{te} \leftarrow$ empty datasets
    for $card$ **in** D_{tr} **do**
        $(subD_{tr}, subD_{te}) \leftarrow$ SCORE($D_{tr}[cardID == card], D_{te}[cardID == card], ot$)
        $DOut_{tr} \leftarrow \{DOut_{tr}, subD_{tr}\}$
        $DOut_{te} \leftarrow \{DOut_{te}, subD_{te}\}$
    end for
end if
if ($gr ==$ "$cluster$") **then** ▷ cluster granularity
    $DOut_{tr}, DOut_{te} \leftarrow$ empty datasets
    $clusterLabel \leftarrow k\text{-}means(cardUsage, k)$
    $D_{tr} \leftarrow \{D_{tr}, clusterLabel\}$
    $D_{te} \leftarrow \{D_{te}, clusterLabel\}$
    for i **from** 1 **to** k **do**
        $(subD_{tr}, subD_{te}) \leftarrow$ SCORE($D_{tr}[cluster == i], D_{te}[cluster == i], ot$)
        $DOut_{tr} \leftarrow \{DOut_{tr}, subD_{tr}\}$
        $DOut_{te} \leftarrow \{DOut_{te}, subD_{te}\}$
    end for
end if

$DOut_{tr}, DOut_{te}$ ▷ augmented training and test set

global granularity is considered (row 13),the entire training and test sets are passed directly to the function SCORE; in the other two cases, only a portion (i.e. the one corresponding to the specific card or cluster) is passed to the function SCORE. In the local approach, this split is completed at the card level (row 18). In the cluster approach, the split is completed at the level of the cluster computed in row 26.

3.3. Metrics

Several metrics have been proposed in the literature to measure the quality of the detection. These include i) the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) [3, 9, 13, 30], ii) the Area Under the Precision-Recall Curve (AUC-PR) [9, 21], iii) the F-measure [1, 3, 16, 31], iv) specificity [3, 30], v) the Recall [3, 30, 31], and vi) the Precision [3, 13, 31, 33]. In this work, we will focus on analyzing the metrics that are considered the most relevant in matters of fraud detection[2]: Top$n$ Precision (i.e. the Precision associated to the n highest risk transactions returned by the algorithm) and AUC-PR. As the ultimate goal of the fraud detection process is to provide the maximum possible number of true positives within the alerts issued for fraud investigators, the most pertinent metric is Top$n$ Precision. This variant of the Precision metric refers to the ratio of the number of true positive alerts to the number of total alerts. We set $n = 100$ in accordance with our industrial partner, as it is possible for a group of investigators to analyze this number of suspicious credit cards in a day. The dependency of Top$n$ Precision on the n value has been studied in [3, 33], who showed that by decreasing n, Precision increases at the cost of a reduced Recall. The metrics that refer to the entire test set (and not simply to alerts) are the AUC-ROC and the AUC-PR. The Area Under the Curve (AUC) is a value in [0,1] that summarizes the relation between two metrics: the Recall and the False Positive Rate in the case of AUC-ROC and Precision and Recall in the case of AUC-PR. The AUC-ROC is equivalent to the probability that a randomly chosen fraudulent transaction has a score higher than that of a randomly chosen genuine transaction [17]. Though the two metrics may look similar, it has been shown that AUC-PR is more effective in cases of high class imbalance [14]. Furthermore, it is known that optimization of the AUC-ROC does not guarantee optimization of the AUC-PR (and vice versa) [14].

[2]This choice was done in agreement with our industrial partner Worldline.

4. Experiments

To compute a consistent outlier score for the local approach, we decided to consider only those cards with histories of at least 10 transactions in the training set. This threshold was set to preserve statistical accuracy in this approach, since the computation of a local outlier score (even the simplest) using fewer transactions would inevitably be affected by large variance, which would undermine the overall accuracy of the approach. This limitation is not present in the cluster and global approaches. To ensure a fair comparison between the three approaches, we also used the same set of cards in the global and cluster approaches. This is beneficial for the cluster approach, since it requires a minimum number of transactions to track customer behavior. Furthermore, we excluded the cards in the test set that were not present in the training set, since it would not be possible in this case to pre-compute outlier scores in the training set.

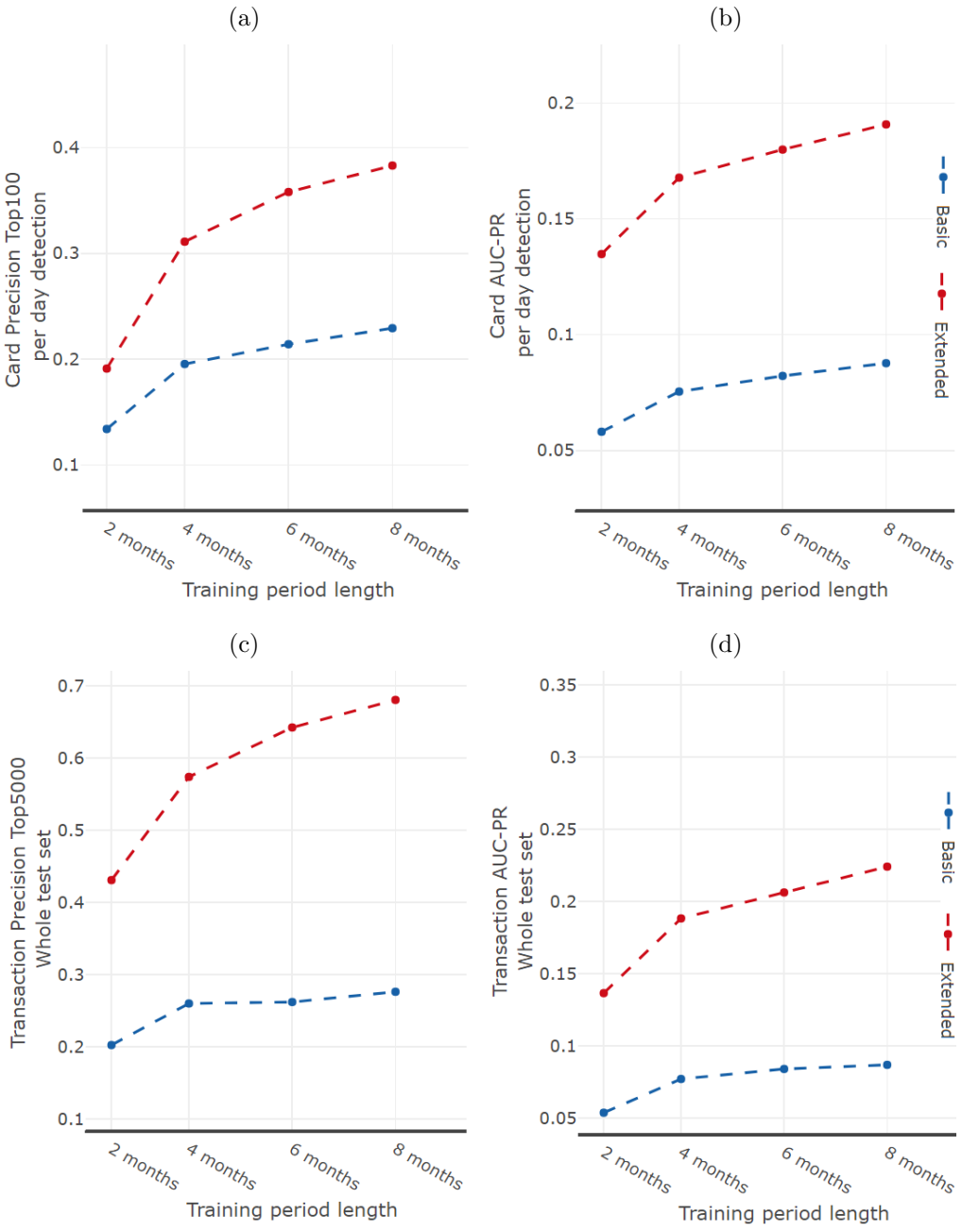
In accordance with the literature, we adopted a random forest model as a baseline, given its superiority in credit card fraud detection [1, 3, 30, 36]. We used a particular implementation of the random forest, Balanced Random Forest (BRF) [6], in which each tree is shaped on a balanced subset of the original sample (and undersampling is used to balance the two classes). A single model is trained over the whole training set and tested on 54 days' worth of data (*Static* approach [13]). The testing interval begins one week after the end of the training set period in order to emulate verification latency [6].

The dataset used for the experiments includes information from 334 days of transactions recorded from February 2 to December 31, 2016. This set was provided to us by our industrial partner, Worldline, a leading company in transactional services, and includes 76 million transactions. The percentage of fraudulent transactions in this dataset is 0.36%. We use transactions until September 30 for the training data, while those that occurred between October 8 and December 31 are used as test set data. The week of October 1 to October 7, 2016 represents the verification latency period, and data from this period is not used for training or testing purposes.

4.1. Baseline

In this first group of experiments, we trained classifiers using datasets taken from 2, 4, 6, and 8 months prior to the cut-off date (October 1, 2016) while using the original set of features alone (in a basic set) or with aggregated

Figure 2: Average daily accuracy in a 54-day test set and of different training lengths: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for fraudulent card detection, (c) the Top5000 Precision for fraudulent transaction detection, and (d) the AUC-PR for fraudulent transaction detection.



features (in an extended set). While the basic set includes the *raw* features obtained by our industrial partner, Worldline, the extended set includes the features acquired through feature engineering [2]. Examples of engineered features include the total sums of money spent by the cardholder and the number of transactions executed by a cardholder over the last 24 hours, as we mentioned previously.

We considered two metrics: the Top$n$ Precision and the AUC-PR. In Figure 2a, the Top100 Precision is plotted, and the Top5000 Precision is reported in 2c. There is a major conceptual difference between these two metrics: the Top100 Precision is computed every day and then averaged over the number of days in the test period, while the Top5000 Precision is computed according to the whole test set without considering the daily label. The same difference exists in the case of the AUC-PR of Figure 2b and Figure 2d.

A second difference is that the detection approach in Figure 2a and Figure 2b aims to detect fraudulent cards, while that of Figure 2c and Figure 2d aims to detect fraudulent transactions. This second metric is typically considered less relevant, as once a fraudulent transaction is detected, the related card is typically blocked, and no further transactions from that card can be considered.According to Figure 2, it appears that, independently of the considered metric, the larger the training set, the better the accuracy [20]. Furthermore, an extended set of features leads to a greater accuracy than that of a basic set.

4.2. Global approach

In this section, we use the best performing configuration of the baseline model (i.e., the configuration trained over an eight-month period using the extended feature set; see Figure 2). This configuration is compared to a model trained on the same set but with the additional features derived from the outlier scores.

In Figure 3, we can see that, in comparison to the baseline model, the use of global outlier scores does not significantly improve fraud detection. In the worst case (i.e., "All Outliers," where we use only the outlier scores and do not consider the baseline feature space), we observe a strong deterioration in accuracy. The outlier scores do not provide additional information, even if they are used in combination with baseline features (baseline + "All Outliers"). In this case, additional features increase the risk of overfitting.

The incapacity of global outliers to contribute useful information to the predictive model may be related to the fact that such scores are highly general and therefore unable to capture the specificity of a given fraud behavior. It is worth recalling that the aforementioned "All Outliers" and "Baseline All Outliers" correspond respectively to the "proposed" and "proposed+" modalities presented by Michenkova et al. in [24].

4.3. Local approach

Figure 4 reports the results of the experiments based on local outlier scores. Regardless of the metric in question, the use of local outlier scores is detrimental to fraud detection. Among all the outliers, the IF local outlier score is the best performing; the PCA-RE outlier score performs as well as the baseline model in terms of Precision Top5000 scores computed using the entire test set. Our interpretation is that while global outlier scores refer to sets that are too large(which introduces large bias), local outlier scores are likely computed on a set of transactions that is too restricted to be useful (introducing large variance). As for the global outlier score approach, the use of solely outlier scores (without considering the baseline feature space) produces a very low accuracy.

4.4. Cluster approach

In the experiments based on cluster outlier scores, we allowed the number of clusters in *k-means* to vary from 10 to 5000. In the first analysis, all unsupervised scores presented in Section 3.1 are used to augment the original dataset (Figure 5). Unfortunately, it appears that this approach is not beneficial to fraud detection and that its accuracy is even lower than that of the global case. If we consider the Top100 Precision metric, we can see that the case with 10 clusters performs better than other cases, even if it does not perform better than the baseline.

The second analysis concerns a study on the relevance of outlier scores with respect to the original features. Several methods exist to assess a feature's relevance. One of the fastest ways is to rely on the relevance returned by random forests. In Figure 6, we show the features used by this classifier, ordered by importance (outlier scores are shown in red). We note that the first two features have a significant impact on the model: they refer to the risk of the shop receiving the payment (*Shop Risk*) and the risk of the shop that received the previous payment (*Last Shop Risk*).Many of the outlier

Figure 3: Accuracy obtained on a test set of data from 54 days while using **global outlier scores** as additional features: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for card detection, (c) Top5000 Transaction Precision for the entire test set, and (d) AUC-PR for the entire test set.

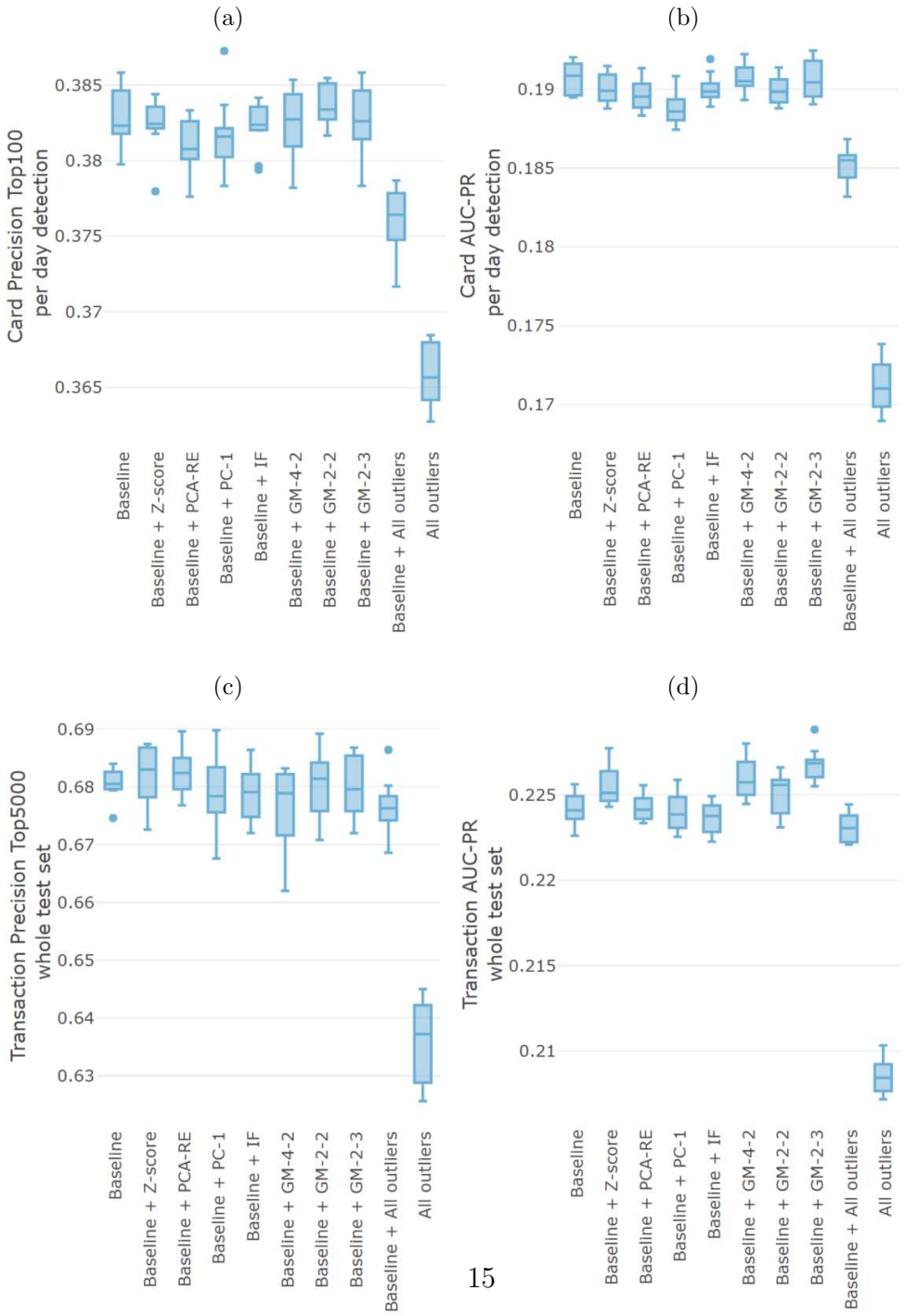


Figure 4: Accuracy obtained on a test set of data from 54 days while using **local outlier scores** as additional features: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for card detection, (c) Top5000 Transaction Precision for the entire test set, and (d) AUC-PR for the entire test set.

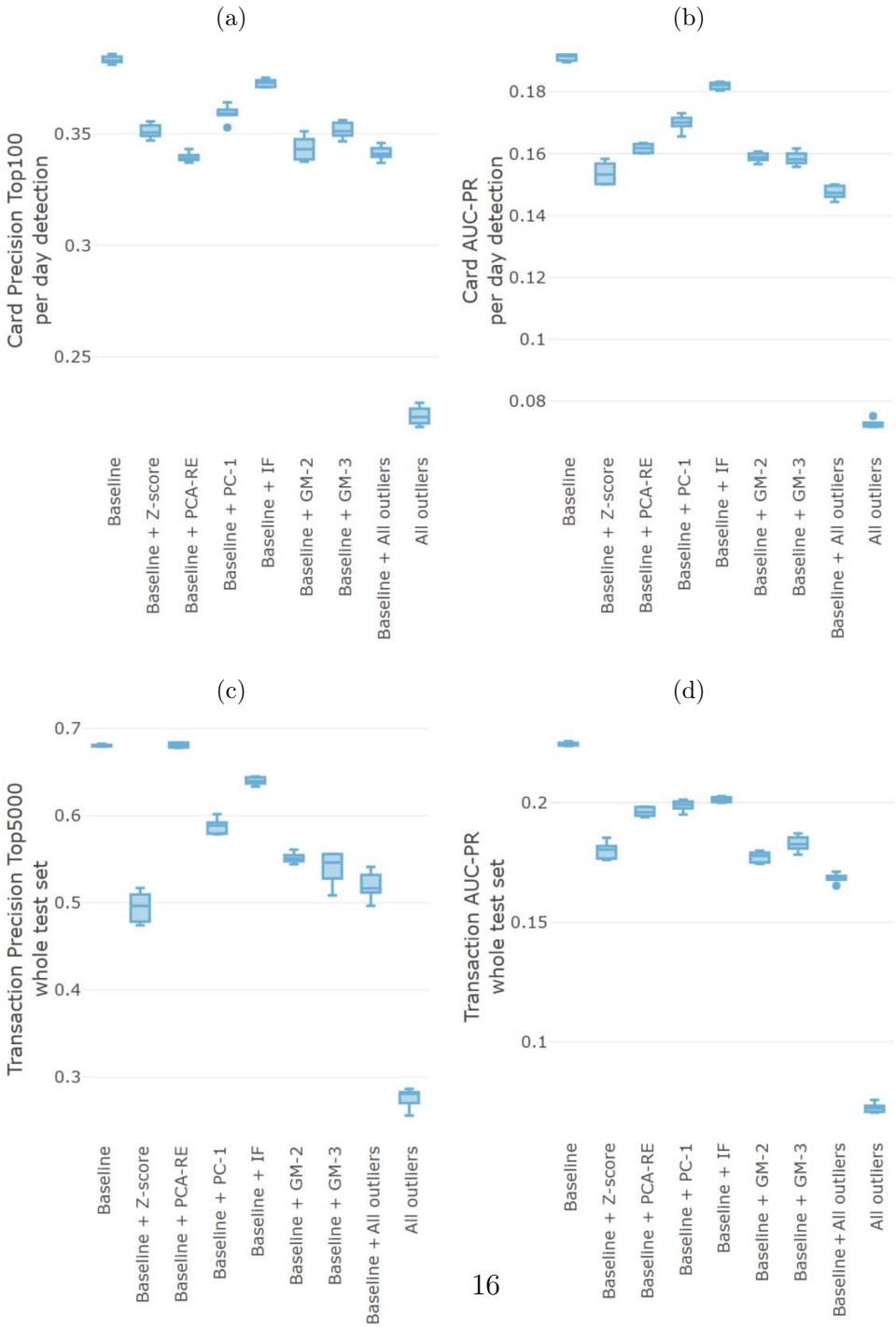
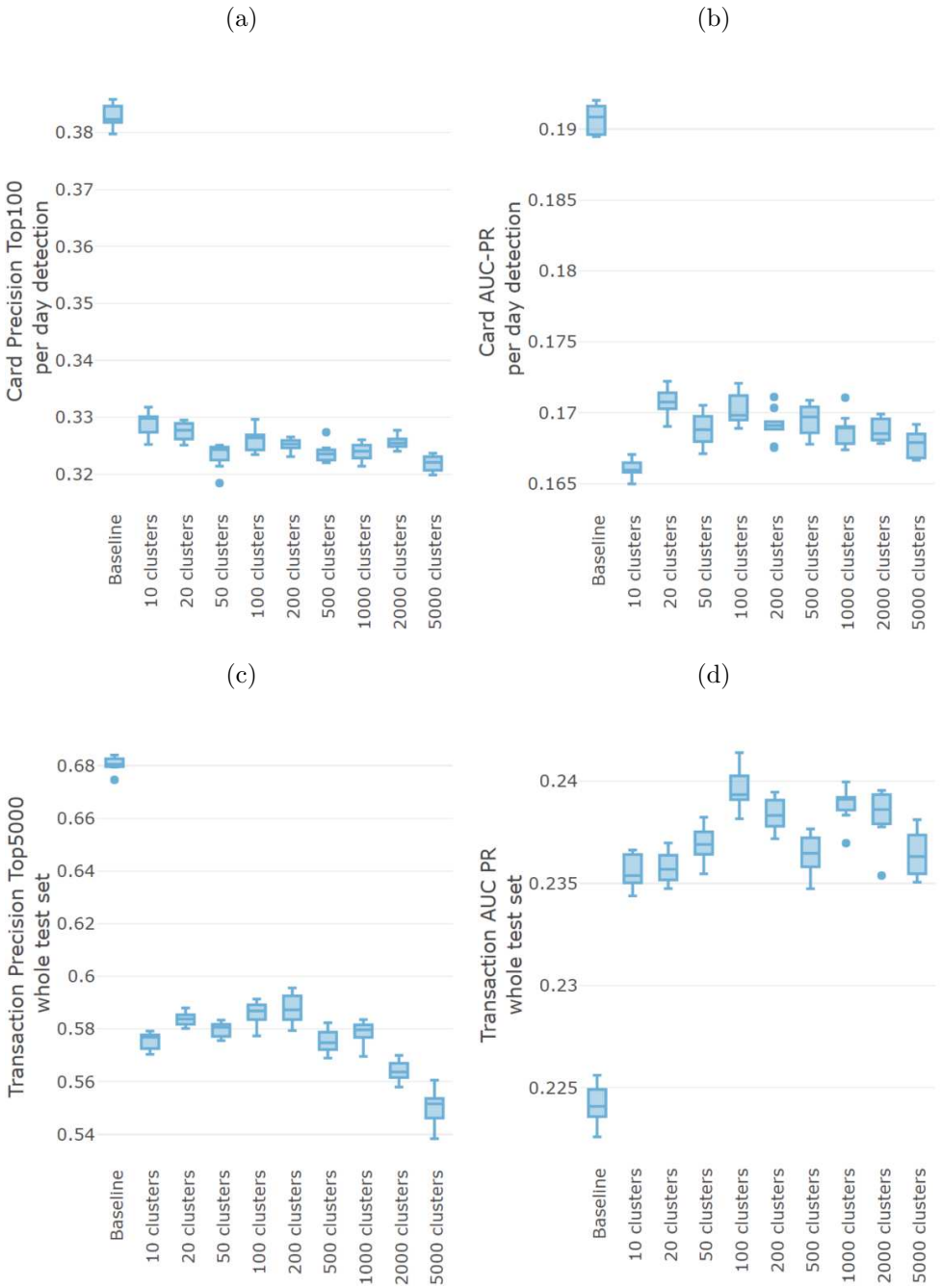


Figure 5: Accuracy obtained on data from a test set of 54 days while using **cluster outlier scores** as additional features: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for card detection, (c) Top5000 Transaction Precision on the entire test set, and (d) AUC-PR on the entire test set.



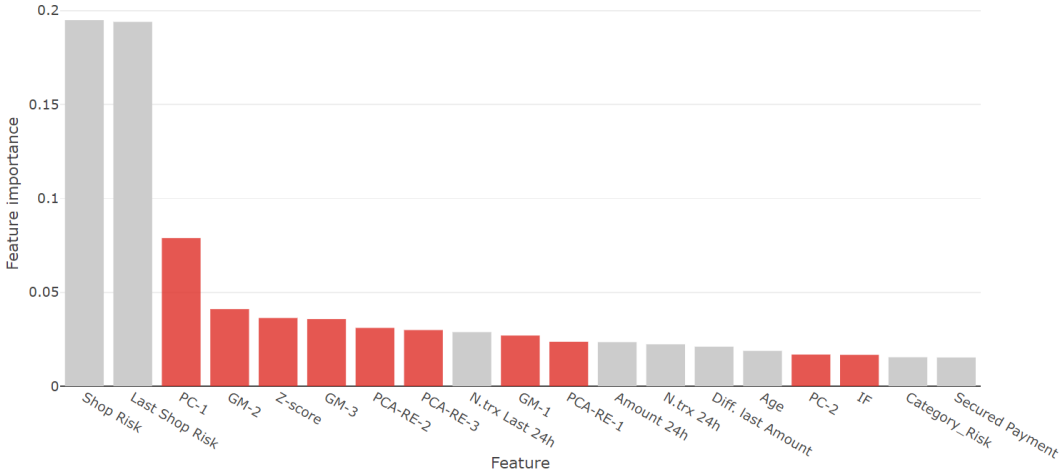


Figure 6: Features ranked by importance, obtained from the random forest classifier of Figure 5 (a 10-cluster case). The red bars refer to the scores returned by an outlier score technique.

scores are ranked just after these two important features, indicating that outlier scores could potentially play a key role in risk prediction.

For this reason, the third analysis focuses only on augmenting the original dataset with a single outlier. We consider the highest ranked outlier score (*PC-1* in Figure 7) and the second highest one (*GM-2* in Figure 8). The interest of *GM-2* is also witnessed by the fact that this score appears among those with the highest Top100 Precision in the global perspective (see Figure 3a).

While no improvement is seen in Top100 Card Precision (Figures 7a and 8a), a significant improvement is visible in terms of Card AUC-PR and Transaction AUC-PR. In the case of *GM-2*, the Top5000 Transaction Precision is also higher.

5. Discussion

Several inconsistencies in the behaviors of the Top100 Precision and the AUC-PR metrics used to assess the cluster approach in comparison to the baseline model warrant deeper analysis. For this reason, we report all Precision-Recall curves in the baseline and the 10-cluster local approaches based on a

Figure 7: Accuracy obtained on a test set of data from 54 days while using **cluster-based PC-1 outlier scores** as additional features: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for card detection, (c) Top5000 Transaction Precision on the entire test set, and (d) AUC-PR on the entire test set.

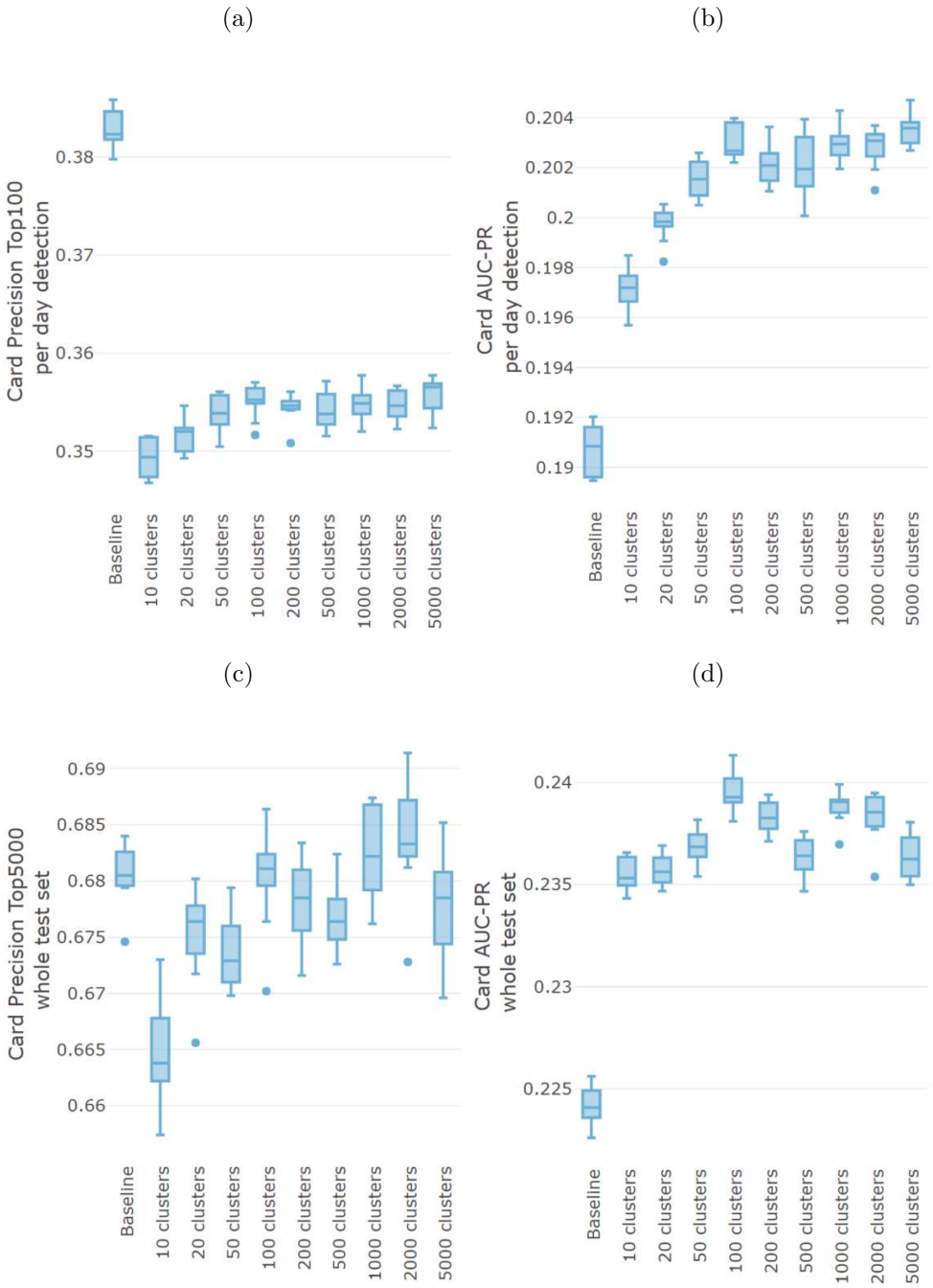
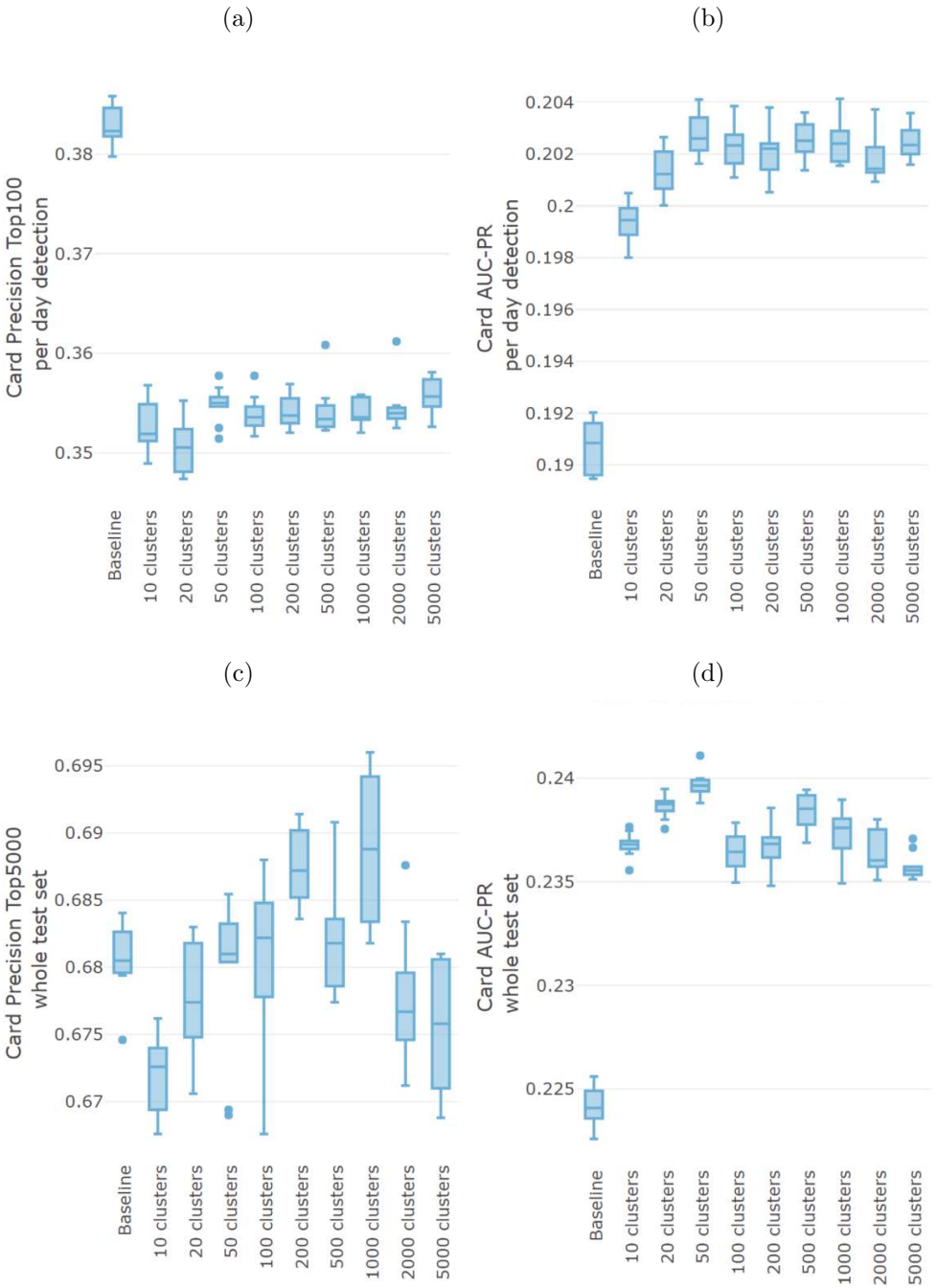


Figure 8: Accuracy obtained on a test set of data from 54 days while using **cluster-based GM-2 outlier scores** as additional features: (a) average daily Top100 Card Precision, (b) average daily AUC-PR for card detection, (c) Top5000 Transaction Precision on the entire test set, and (d) AUC-PR on the entire test set.



combination of baseline features and GM-2 outlier scores (shown in blue and red, respectively, in Figure 9a).

First, we note that the red curve is higher than the Baseline curve for Recall values ranging from 0.1 to 0.3. This is in line with the AUC-PR accuracy observed in Figures 8b and 8d, in which the cluster approach outperforms the baseline approach.

Figure 9b shows a closeup of Figure 9a, focusing on the Recall interval between 0 and 0.01. We observe here that the baseline PR curve often rises above the 10-cluster PR curve. This is consistent with the Top100 Card Precision (related to a low Recall configuration) illustrated in Figure 8a, where the baseline approach outperforms the "baseline + GM-2 outlier score" approach.

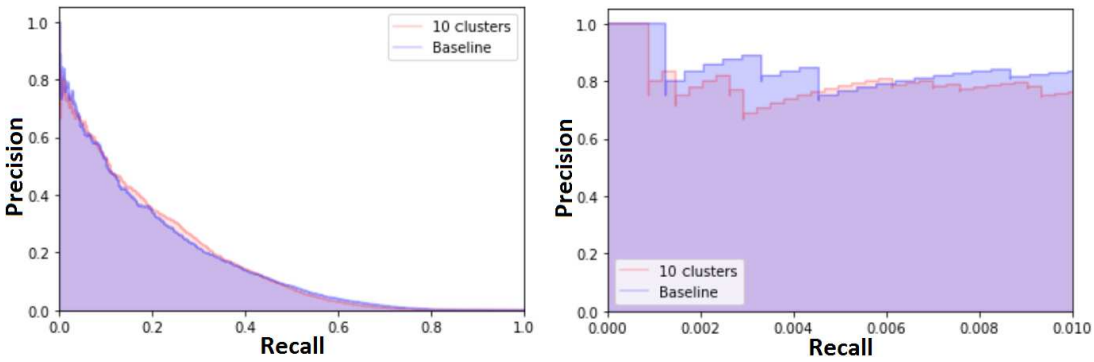
The cluster approach is shown to be the most promising method; however, it also has several technical limitations. First of all, it requires the choice of hyper-parameters. Selecting the appropriate *contextual attributes* is not a trivial matter (see Section 3.2), and the adoption of *k-means* requires the setting of a number of clusters k, which has an impact on the outlier score's level of granularity.

A further disadvantage of this method relates to the minimum number of transactions necessary for a card to be analyzed. As was mentioned in Section 4, in the local approach, we were restricted to considering cards in the training set with histories of more than 10 transactions. Note that the use of such a filter could make some fraudulent patterns non-observable.

6. Conclusion

This article proposes the implementation of a hybrid approach that makes use of unsupervised outlier scores to extend the feature set of a fraud detection classifier. The novelty of the contribution, beyond its applications in real and sizeable datasets of credit card transactions, is the implementation and assessment of different levels of granularity for the definition of an outlier score. The granularity in question ranges from the card level to the global level, considering intermediate levels of card aggregation through clustering.

The results are not convincing in terms of the global and local approaches. Our interpretation is that both approaches do not have the right level of granularity necessary to take advantage of unsupervised information. A more promising outcome is obtained through the cluster approach (notably in terms of AUC-PR), though it appears that augmenting data sets with too



(a) (b)

Figure 9: Precision-Recall curve obtained from a single day of test set data using only the baseline features and a combination of **cluster-based GM-2 outlier scores** and baseline features (10 clusters): (a) Precision-Recall curve for card detection and (b) zoomed Precision-Recall curve for card detection.

many scores could be detrimental due to overfitting and variance issues. The obtained results open the way for several potential research directions:

- The fact that the *best-of-both-worlds* method provides improvements in terms of AUC-PR but not in terms of Top$n$ Precision indicates that the added value of unsupervised measures may depend on the adopted accuracy criterion.
- Additional work (in terms of different clustering algorithms and different sets of features) with the clustering metric could shed further light on the relevance of this approach.
- Though many outlier scores seem to provide information about fraud risk (see the relevance plot in Figure 6), using many of them at the same time is detrimental to the approach's final accuracy.
- The impact of granularity on the approach's accuracy indicates the importance of analyzing datasets in a stratified manner, not only in an unsupervised manner but also in a supervised manner (e.g., by introducing some notion of locality).

Acknowledgement

The authors FC, YLB, and GB acknowledge the funding of the BruFence and DefeatFraud projects; both are supported by INNOVIRIS (Brussels Institute for the Encouragement of Scientific Research and Innovation).

References

[1] Bahnsen, A. C., Aouada, D., and Ottersten, B. (2015). Example-dependent cost-sensitive decision trees. *Expert Systems with Applications*, 42(19):6609–6619.

[2] Bahnsen, A. C., Aouada, D., Stojanovic, A., and Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. *Expert Systems with Applications*, 51:134–142.

[3] Bhattacharyya, S., Jha, S., Tharakunnel, K., and Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems*, 50(3):602–613.

[4] Bühlmann, P., Yu, B., et al. (2002). Analyzing bagging. *The Annals of Statistics*, 30(4):927–961.

[5] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32. Springer

[6] Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.-A., Caelen, O., Mazzer, Y., and Bontempi, G. (2018a). Scarff: a scalable framework for streaming credit card fraud detection with spark. *Information fusion*, 41:182–194.

[7] Carcillo, F., Le Borgne, Y.-A., Caelen, O., and Bontempi, G. (2017). An assessment of streaming active learning strategies for real-life credit card fraud detection. In *Data Science and Advanced Analytics (DSAA), 2017 IEEE International Conference on*, pages 631–639. IEEE.

[8] Carcillo, F., Le Borgne, Y.-A., Caelen, O., and Bontempi, G. (2018b). Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization. *International Journal of Data Science and Analytics*, pages 1–16.

[9] Carneiro, N., Figueira, G., and Costa, M. (2017). A data mining based system for credit-card fraud detection in e-tail. *Decision Support Systems*, 95:91–101.

[10] Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15.

[11] Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., and Bontempi, G. (2017). Credit card fraud detection: a realistic modeling and a novel learning strategy. *IEEE transactions on neural networks and learning systems*.

[12] Dal Pozzolo, A., Caelen, O., and Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 200–215. Springer.

[13] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., and Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert systems with applications*, 41(10):4915–4928.

[14] Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240. ACM.

[15] Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612.

[16] Fu, K., Cheng, D., Tu, Y., and Zhang, L. (2016). Credit card fraud detection using convolutional neural networks. In *International Conference on Neural Information Processing*, pages 483–490. Springer.

[17] Hanley, J. A. and McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36.

[18] Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

[19] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of educational psychology)*, 24(6):417. Warwick & York.

[20] Junqué de Fortuny, E., Martens, D., and Provost, F. (2013). Predictive modeling with big data: is bigger really better? *Big Data*, 1(4):215–226.

[21] Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.-E., He-Guelton, L., and Caelen, O. (2018). Sequence classification for credit-card fraud detection. *Expert Systems with Applications*.

[22] Liang, J. and Parthasarathy, S. (2016). Robust contextual outlier detection: Where context meets sparsity. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 2167–2172. ACM.

[23] Liu, F.T. and Ting, K.M. and Zhou, Z. (2008). Isolation forest In *Eighth IEEE International Conference on Data Mining*, pages 413–422. IEEE

[24] Micenková, B., McWilliams, B., and Assent, I. (2014). Learning outlier ensembles: The best of both worldssupervised and unsupervised. In *Proceedings of the ACM SIGKDD 2014 Workshop on Outlier Detection and Description under Data Diversity (ODD2). New York, NY, USA*, pages 51–54.

[25] Nguyen, H. V., Ang, H. H., and Gopalkrishnan, V. (2010). Mining outliers with ensemble of heterogeneous detectors on random subspaces. In *International Conference on Database Systems for Advanced Applications*, pages 368–383. Springer.

[26] Rayana, S., Zhong, W., and Akoglu, L. (2016). Sequential ensemble learning for outlier detection: A bias-variance perspective. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 1167–1172. IEEE.

[27] Sethi, N. and Gera, A. (2014). A revived survey of various credit card fraud detection techniques. *International Journal of Computer Science and Mobile Computing*, 3(4):780–791.

[28] Shimpi, P. R. and Kadroli, V. (2015). Survey on credit card fraud detection techniques. *International Journal Of Engineering And Computer Science*, 4(11):15010–15015.

[29] Song, X., Wu, M., Jermaine, C., and Ranka, S. (2007). Conditional anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 19(5):631–645.

[30] Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., and Baesens, B. (2015a). Apate: A novel approach for automated credit card transaction fraud detection using network-based extensions. *Decision Support Systems*, 75:38–48.

[31] Van Vlasselaer, V., Eliassi-Rad, T., Akoglu, L., Snoeck, M., and Baesens, B. (2015b). Afraid: fraud detection via active inference in time-evolving social networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2015 IEEE/ACM International Conference on*, pages 659–666. IEEE.

[32] Veeramachaneni, K., Arnaldo, I., Korrapati, V., Bassias, C., and Li, K. (2016). Aiˆ 2: Training a big data machine to defend. In *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*, pages 49–54. IEEE.

[33] Wei, W., Li, J., Cao, L., Ou, Y., and Chen, J. (2013). Effective detection of sophisticated online banking fraud on extremely imbalanced data. *World Wide Web*, 16(4):449–475.

[34] Wold, S. and Esbensen, K. and Geladi, P. (1987). Principal component analysis. In *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52. Elsevier.

[35] Yamanishi, K. and Takeuchi, J.-i. (2001). Discovering outlier filtering rules from unlabeled data: combining a supervised learner with an unsupervised learner. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 389–394. ACM.

[36] Zareapoor, M. and Shamsolmoali, P. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia Computer Science*, 48:679–685.

[37] Zhu, X. (2005). Semi-supervised learning literature survey. Technical report, Computer Sciences TR 1530, University of Wisconsin Madison.

[38] Zimek, A., Campello, R. J., and Sander, J. (2014). Ensembles for unsupervised outlier detection: challenges and research questions a position paper. *Acm Sigkdd Explorations Newsletter*, 15(1):11–22.