International Journal of Basic and Applied Sciences, 13 (2) (2024) 47-55

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS

Research paper

Model for detecting anomalies and suspicious transactions on financial and non-financial entries

Coulibaly Kpinna Tiekoura [1] *, Mambe Digrais Moïse [2], Diaby Moustapha [3]

[1] LASTIC, Ecole Supérieure Africaine des TIC (ESATIC) ; Côte d'Ivoire, 18bp 1501 Abidjan 18
[2] LMI, Université Nangui Abrogoa, Côte d'Ivoire, 02 BP 801 Abidjan 02
[3] LASTIC, Ecole Supérieure Africaine des TIC (ESATIC) ; Côte d'Ivoire, 18bp 1501 Abidjan 18
**Corresponding author E-mail: tiekoura77@yahoo.fr*

Abstract

Some conventional audit methods sometimes fall short in identifying subtle irregularities and emerging fraudulent schemes. Our proposal is therefore part of the aim of improving audit processes. At the end of the experiments on two sets of data, our model offers the best results compared to other models, particularly in terms of CPU execution time of the model and also in terms of performance in the detection of normal data with a better rate of false positives.

Keywords: *Anomaly; Audit; Dissimilarity; Distance; Transaction.*

1. Introduction

In the ever-changing business world, where financial transactions are increasing in number and complexity, the importance of anomaly detection in audit engagements is becoming increasingly crucial. Auditors play a vital role in ensuring the transparency, integrity and compliance of companies' financial information. From this perspective, the detection of anomalies and suspicious transactions takes on particular significance, ranging from preventing fraud to preserving data integrity. Indeed, the complexity of rapidly changing markets and technologies has resulted in a considerable increase in the number of transactions where auditors face time constraints, creating complex challenges for them. Conventional audit methods can sometimes be insufficient to identify subtle irregularities and emerging fraudulent schemes. It is in this context that the detection of anomalies and suspicious transactions, supported by advanced tools and methodologies, is positioned as an essential pillar for strengthening the effectiveness of audit missions.
Several research works have focused on the problem of anomaly detection in different domains. Each of the solutions proposed resulting from his work is characterized by the field of application, the type of data used, the execution time and especially the approach or method of detection and classification of anomalies, given that each method has its strengths and weaknesses.
In this paper, given the limitations observed in existing solutions, we propose a model for detecting anomalies and suspicious transactions on financial and non-financial entries to ensure the reliability of financial transactions. This is all the more important when we know that every day, companies, particularly financial companies, are faced with different types of risks leading to enormous financial losses when they are poorly managed [1].
Our work will be structured around four main parts. In the first part, we will provide a description of the state of the art and then we will address our problem in the second part. The third part concerns the conceptual study and implementation of our SDAOS system. The fourth part will be devoted to the discussion of the results obtained. We will end with a conclusion with research perspectives.

2. State of the art

An anomaly can be described as an observation whose deviation or dissimilarity from the rest of the observations suggests that it was produced by a different mechanism [2].
In the literature, we distinguish a range of anomaly detection methods, each characterized by its strengths and weaknesses.
These anomaly detection techniques are generally classified into seven (7) broad categories, namely (i) distance, (ii) depth, (iii) distribution (iv) density, (v) classification, (vi) grouping (or clustering) and (vii) spectral decomposition or projection of data.

- Techniques based on Distance ([3] [4] [5] [6]): Among these techniques, we can cite the Euclidean distance, the Mahalanobis distance as well as the SLOM, ROF, AnyOut, k-medoids methods, CBLOF, etc. In this approach, the authors use the distance between an observation and the entire data set to assess whether this observation is an anomaly or not. The thresholds are comparable to those of density-based methods. Since common estimates of mean and variance are heavily influenced by outliers, more robust estimates have been made so that distance-based methods are more reliable [7].

- Depth-based techniques: One of the techniques of this approach is KSD [8]. The data set is modeled by a set of convex hulls whose objective is to group the data relative to their proximity to the core of the data set. The convex hulls near the core of the data set contain the most reliable values while the outer hulls contain the anomalies. The calculation time becomes excessive when the dimensions of the data set exceed. 3 or 4 [9].
- Techniques based on Distribution [10] [11]: These are Gaussian mixture models, the Bayesian approach, kernel estimation, etc. These methods aim to model the distribution of data with a known or unknown function, that is to say a parametric or non-parametric approach. Parametric techniques include the ARIMA, VARMA, GMM, MVE, EVT, Z-Score, Grubb's and Likelhood methods. For non-parametric techniques, we can cite HBOS, SmartSifter, BoxPlot, EWMA, CUSUM, Histogram. The principle is that observations whose behavior is suspicious compared to established distributions are considered anomalies. In order to decide whether a value differs significantly from the distribution, thresholds are most often set. It is important to emphasize that parametric methods are criticized for the fact that the distribution of data is generally unknown in advance.
- Density-based techniques: Several anomaly detection techniques using this approach have been proposed in the literature. They start from the principle that the density of observations near a value containing an anomaly is significantly lower than the density of observations near a safe value. When the dataset is composed of subsets of varying densities, these techniques are generally preferable to distance-based methods. As with the distance approach, thresholds are set to define the proximity of an observation and the limit beyond which a value is considered abnormal. One of the best known among them is the Local Outlier Factor (LOF) method proposed by Breunig et al. (2000) [12]. It is an unsupervised method which gives a score representing the degree of aberration of the observation. According to this technique, any observation whose degree of anomaly is significantly greater than 1 is considered an abnormal observation. It takes as a parameter the number of nearest neighbors to consider.

There are several improvements to LOF in the literature, notably iLOF and MILOF:

- Incremental LOF "iLOF" [13] is adapted to data flows and nevertheless consumes enough memory in calculating the density of new incoming data.
- Memory Efficient ILOF "MILOF" [14] is an improvement of iLOF which reduces memory consumption while having precision similar to iLOF.

Furthermore, for large datasets, we have the Grid-LOF "GLOF" method [15] which divides the dataset into small regions called Grid before calculating the density.

- Techniques based on classification: In this category, we distinguish between methods based on the supervised approach and that on the unsupervised approach.

Supervised methods require a database containing a label that indicates whether an observation is normal or abnormal to train their model on points with a known label. Among these methods, we have One-Class Support Vector Machine (OC-SVM) ([16][17]) which is an anomaly detection method which applies SVM algorithms to the problem of One class classification (OCC) and the principle of which is the search for a hyperplane in a high-dimensional space which separates anomalies from normal data. As for unsupervised methods, they have a great advantage because they do not require data labeling. Thus, they manage to detect anomalies by isolating observations that turn out to be unusual compared to others. This makes it possible to detect new types of anomalies, unlike supervised learning algorithms which only identify anomalies that are consistent with the labeled data and the built predictive model. In this category, we can cite the Isolation Forest (or IForest) method ([18][19]) which is based on decision trees and random forests. The latter, considered one of the most recent and most widely used anomaly detection methods, uses the isolation of observations from the construction of several random trees. When from the root, a forest of random and independent trees collectively produces a short path to reach an observation, it is assigned a high probability of constituting an anomaly.

A new version of IForest, called Majority Voting IForest (MVIForest) [20] exists and is based on the different decisions of individual trees rather than a global decision of the forest. MVIForest has a shorter execution time than IForest, with almost the same performance for detection.

- Techniques based on Clustering: These techniques such as CLARANS and DBSCAN ([21][22]) seek to group observations into different subsets and consider that the outliers are the residuals of the process, that is to say the observations that are not attached to a subset. One of the limitations of these methods is the fact that their output strongly depends on the choice of the clustering algorithm. Furthermore, they are more suitable for grouping (or clustering) of data and not for detecting anomalies. These methods require prior calculation, and as a result, they are expensive in execution time.
- Techniques based on data projection or spectral decomposition ([23][24]): The principle of these methods consists of projecting observations onto a new subspace to facilitate the detection of anomalies. PCA, GWPCA, etc. are among these methods. The number of principal components to choose for decomposition remains a challenge and is always defined by a threshold.

3. Problematic

Most of the detection methods described above use manual thresholds for data filtering, that is to say the threshold beyond which an observation is considered abnormal. The choice of these thresholds is crucial because it can have a considerable impact on the detection of abnormal values. A high threshold would increase the false positive rate while a too low threshold would increase the false negative rate.

Furthermore, some anomaly detection methods are limited a priori by the distributions or the type of input data. This constitutes an obstacle for the development of general and automated tools and does not allow the use of these methods on multiple and diverse data.

In addition, certain techniques detect cases of fraud or anomalies by limiting themselves to selecting the most suspicious cases without clear and precise justification for these anomalies, something which often complicates the auditor's work.

Also, some approaches such as Euclidean distance assume that all variables are all important and independent, which may not always be true in real-world scenarios.

One of the important issues of an anomaly detection method lies in the relevance or performance of its classifier and the calculation time of its algorithm. Performance is relative to the rate of false positives and false negatives and is generally estimated using the area under the ROC curve and certain metrics such as Specificity and Recall ([25][26]).

Thus, in this paper, we attempt to provide an answer to the problem relating to performance, calculation time, but above all to the extent of the type of data taken into account. In other words, how can we create an efficient anomaly detection method with better calculation time and which takes into account multiple and diverse data?

4. Contribution

4.1. Description

To solve this problem, we propose a hybrid method using the Mahalanobis distance and the cosine similarity. Mahalanobis distance, unlike other distance techniques such as Euclidean distance or Manhattan distance, takes into account the covariance structure of the data, all of which makes it particularly useful when managing datasets where the Variables are correlated and have different scales. It thus captures the relationships between variables and provides a more precise measure of the dissimilarity between two points. Another benefit of choosing this metric for our model is its great ability to effectively standardize variables at different scales or units of measurement and allow meaningful comparisons. This is the case for a dataset including both income (in dollars) and temperature (in degrees Celsius). Finally, we use Mahalanobis distance for its effectiveness in cluster analysis by grouping similar data points and also for its robustness to outliers through the fact that it is less affected by those values that deviate from the global data model.
The Mahalanobis distance being more suited to quantitative data, in order to extend the field of data usable by our model, we associated the cosine similarity which takes into account categorical or textual data.
In addition, our anomaly detection model includes an alert system that triggers immediate notification if a suspicious transaction is detected. These alerts contain detailed information about the transaction in question, the associated risk score, and other relevant details.
Finally, to ensure data security and compliance, we have added encryption protocols and secure access to the system, ensuring that our solution complies with current data protection regulations.

4.2. Mathematical formalism of our model

4.2.1. Function for calculating dissimilarity between quantitative variables

X : Vector of quantitative observations composed of p variables
x_{ij} is the observation of the j th variable Xj on the i th individual
$X_i = (x_{i1}\ x_{i2}\ x_{i3} \dots x_{ij} \dots x_{ip})$: vector X observed on the i th individual
$X_j = \begin{pmatrix} x_{1j} \\ \dots \\ x_{ij} \\ \dots \\ x_{nj} \end{pmatrix}$: the j th variable observed on all n individuals
n : number of individuals
p : number of variables observed on the i^th individual
$\bar{X} = (\bar{X}_1\ \bar{X}_2 \dots . \bar{X}_j \dots \bar{X}_p)$: Vector of means of p variables or mean vector of observations or center of gravity G of the cloud of points
S : Covariance variance matrix or dispersion matrix
$S_{jk} = C_{X_j X_k}$: The covariance betweenX_j and X_k
$M = S^{-1}$: Diagonal matrix of inverse variances
$X = \begin{pmatrix} x_{11}\ x_{12} \dots x_{1j} \dots x_{1p} \\ x_{21}\ x_{22} \dots x_{2j} \dots x_{2p} \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ x_{i1}\ x_{i2} \dots x_{ij} \dots x_{ip} \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ x_{n1}\ x_{n2} \dots x_{nj} \dots x_{np} \end{pmatrix}$: observation matrix

The vector of means of the p variables is calculated as follows:

$$\bar{X} = \frac{1}{n}\left(\sum_{i=1}^{n} x_{i1} \sum_{i=1}^{n} x_{i2} \dots \sum_{i=1}^{n} x_{ip}\right) \quad (1)$$

The covariance between X_j and X_k :

$$S_{jk} = C_{X_j X_k} = \frac{1}{n}\sum_{i=1}^{n}(x_{ik} - \bar{X}_k)\left(x_{ij} - \bar{X}_j\right) \quad (2)$$

$$S_{jk} = \frac{1}{n}\sum_{i=1}^{n} x_{ik}\, x_{ij} - \bar{X}_k \bar{X}_j \quad (3)$$

The dispersion matrix or covariance variance matrix:

$$S = \begin{pmatrix} S_{11} & S_{1p} \\ S_{p1} & S_{pp} \end{pmatrix} \quad (4)$$

Diagonal matrix of inverse variances:

$$S^{-1} = \begin{bmatrix} 1/S_1^2 & \cdots 0 & 0 \\ \vdots & \ddots 0 & \vdots \\ 0 & \cdots 0 & 1/S_p^2 \end{bmatrix} = D_{\frac{1}{S^2}} \quad (5)$$

Mahalanobis distance between an individual X_i and the center of the point cloud $\bar{X}$

$$d^2_{\mathcal{M}}(X_i, \bar{X}) = (X_i - \bar{X})^T S^{-1} (X_i - \bar{X}) \quad (6)$$

The Mahalanobis distance between an individual X_i and the center of the point cloud $\bar{X}$ is a positive value that quantifies the dissimilarity between a data point and the mean of the entire data set. A smaller distance indicates that the data point is closer to the mean and is more similar to the data set as a whole. On the other hand, a greater distance means a greater dissimilarity.
For a better interpretation of the Mahalanobis distance, we opted for a statistically determined rejection threshold. Unlike most detection methods described in the state of the art which use manual thresholds, our approach is more objective and makes it possible to reduce the rate of false positives and false negatives.
The rejection threshold φ is calculated as follows:

$$\varphi = \frac{t_{\alpha/2}(n-1)}{\sqrt{n}\sqrt{n-2+t_{\alpha/2}{}^2}} \quad (7)$$

With $t_{\alpha/2}$ the critical value coming from the Student's Law table and n the sample size
If $d^2_{\mathcal{M}}(X_i, \bar{X}) > \varphi$, the value is abnormal data;
If $d^2_{\mathcal{M}}(X_i, \bar{X}) \leq \varphi$, the value is not abnormal data
i) Function for calculating the dissimilarity between qualitative variables
In the case of qualitative variables, we use the cosine distance to measure the degree of similarity of a given document in order to detect the anomaly. This distance works by transforming textual data into a vector of digital representations. These numerical values will be represented by the weights $W_{u_i,d}$ measuring the importance of the terms contained in a given document in relation to a corpus of documents. Thus, the similarity between two documents d_1 and d_2 is determined based on the angle between their corresponding vectors.
The following pseudo algorithm presents the process of calculating the similarity between documents with the cosine distance:

Algorithm 1: Calculate the similarity between two documents
1. Carry out text preprocessing by removing stop words, special characters and performing radicalization or lemmatization to normalize the text.
2. Calculate term frequency (TF) by counting the frequency of each term in the document.
3. Calculate Inverse Document Frequency (IDF) which measures the importance of each term in all documents to give higher weight to rare terms.
4. Calculate TF-IDF by combining TF and IDF to obtain the final digital representation of the documents.
5. Then calculate the cosine similarity using the TF-IDF vectors of the documents

$\vec{u}$: vector u of qualitative variables to analyze, comprising q terms
$\vec{v}$: vector v of qualitative variable playing the role of the reference vector and comprising q terms
u_i : i th term of the vector $\vec{u}$: $\vec{u}\ (u_i, \dots, u_q)$
v_i : i th term of the vector $\vec{v}$: $\vec{v}\ (v_i, \dots, v_q)$
df_{u_i}: number of documents containing the term u_i
$n_{u_i,d}$: Number of times the term u_i appears in the document d
$\sum_k n_{k,d}$: Number of terms in the document d
N : total number of documents
Frequency of a term u_i in a document d

$$tf_{u_i,d} = \frac{n_{u_i,d}}{\sum_k n_{k,d}} \quad (8)$$

The inverse frequency of the term term u_i in the document or importance of the term u_i in the document

$$idf_{u_i} = log_{10}\left(\frac{N}{df_{u_i}}\right) \quad (9)$$

The weight of the term u_i in document d relative to the corpus of documents.

$$W_{u_i,d} = tf_{u_i,d}.idf_{u_i} \quad (10)$$

$$W_{u_i,d} = \frac{n_{u_i,d}}{\sum_k n_{k,d}} X log_{10}\left(\frac{N}{df_{u_i}}\right) \quad (11)$$

The Cosine similarity of the angle Θ between the vectors $\vec{u}$ and $\vec{v}$ $is\ obtained\ by$:

$$SimCos(\vec{u}, \vec{v}) = \frac{\vec{u}.\vec{v}}{\|\vec{u}\|.\|\vec{v}\|} \quad (12)$$

$$SimCos(\vec{u}, \vec{v}) = \frac{\sum_{i=1}^{q} u_i v_i}{\sqrt{\sum_{i=1}^{q} u_i{}^2}\sqrt{\sum_{i=1}^{q} v_i{}^2}} \quad (13)$$

$SimCos(\vec{u}, \vec{v}) \in [-1, 1]$,

$$\text{If } \begin{cases} SimCos(\vec{u}, \vec{v}) = -1, \rightarrow \text{ the vectors } u \text{ and } v \text{ are opposite} \\ SimCos(\vec{u}, \vec{v}) = 0, \rightarrow \text{ the vectors } u \text{ and } v \text{ are not similar} \\ SimCos(\vec{u}, \vec{v}) = 1, \rightarrow \text{ the vectors } u \text{ and } v \text{ are similar} \\ SimCos(\vec{u}, \vec{v}) \in]-1,1[: \text{there is an intermediate similarity/dissimilarity between the 2 vectors} \end{cases}$$

ii) General function of our anomaly detection method

$$F(X_i, u_i, v_i, \xi, \lambda) = \xi d^2_{\mathcal{M}}(X_i, \bar{X}) + \lambda SimCos(\vec{u}, \vec{v}) \quad (14)$$

$$F(X_i, u_i, v_i, \xi, \lambda) = \xi (X_i - \bar{X})^T S^{-1} (X_i - \bar{X}) + \lambda \frac{\sum_{i=1}^{q} u_i v_i}{\sqrt{\sum_{i=1}^{q} {u_i}^2} \sqrt{\sum_{i=1}^{q} {v_i}^2}} \quad (15)$$

With $\begin{cases} \xi = 1 \; et \; \lambda = 0 \; if \; the \; data \; is \; quantitative \\ \xi = 0 \; et \; \lambda = 1 \; if \; the \; data \; is \; qualitative \end{cases}$

4.3. Functional architecture of our model

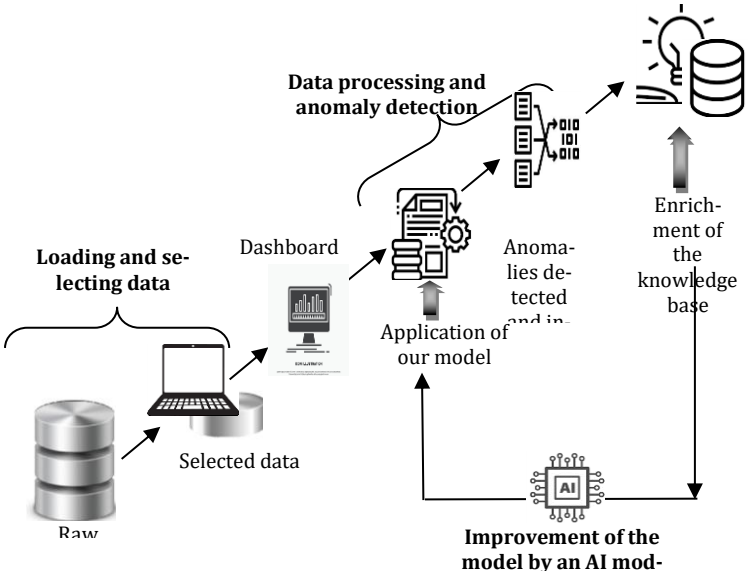

Fig. 1: Functional Architecture of Our System.

a) Loading and Selection of Data

The first step of the system concerns the loading and selection of data. It allows users to upload data files that are previously checked by the system to identify the data type, based on the file extension. If the extension is not supported such as pdf files, the file is rejected by the system and an error message is returned. Otherwise, it is accepted and the file is stored in memory, ready for use in subsequent steps of the process.

b) Dashboard

After data loading, the dashboard provides a comprehensive visualization of essential information. This includes relevant details about the data used. The dashboard provides a clear and concise perspective, making it easy to understand fundamental data characteristics and enable in-depth analysis.

c) Data processing and anomaly detection

The last phase of the system is based on the detection of potentially suspicious transactions.

At this level, our system will apply the anomaly search algorithm depending on the type of data (qualitative or quantitative). In the case of the financial transactions that concern us in this study, our algorithm also applies specific rules (Table 1), evaluates each transaction and assigns a score based on various criteria. This assessment then allows the system to identify abnormal transactions based on high scores. These different anomalies are then recorded in a knowledge base which will strengthen, thanks to an associated AI module, the ability of our system to effectively identify unusual patterns in future data to be analyzed.

It should be remembered that the integration of rules into our process for detecting anomalies and suspicious financial transactions plays a central role in evaluating the conformity of data and highlighting atypical patterns. These rules, defined as specific criteria, make it possible to determine what is considered abnormal within a data set by assigning pre-established scores.

The underlying method involves calculating an arithmetic average based on these varying scores. Observations that deviate significantly from the norm are likely to receive higher scores, thereby signaling an increased likelihood of an anomaly or suspicious operations. Ultimately, our approach provides a quantitative approach to assess deviation from defined rules, thus facilitating automatic and accurate detection of non-compliant behavior within the data.

Table 1: Anomaly Detection Rules and Suspicious Operations

RULES	SCORES
A transaction for an amount greater than the average of all transactions under the same transaction code (debit/credit)	3
Transactions to the credit of an account belonging to the category of a personnel account (personnel account class)	3
Self-validated operations (Auth ID = system)	4
Operations carried out on weekends and public holidays	4
Operation such as “credit/debit regulation, transfer, miscellaneous transfer by Gl, credit/debit cancellation, miscellaneous transfer” FDI, GLT, NUD, NUC, VDE, VDR	2
Duplicate operation	4

The scores assigned in the table above represent the degree of risk associated with each rule, scaled on several levels from 1 to 4. Rules with a level of 4 are those which present the highest level of vulnerability, thus indicating that they are likely to have a significant impact on the detection of anomalies.

4.4. Pseudo code

Algorithm 2: Pseudo code for detecting an anomaly

```
1.      BEGIN
2.      // Data Loading Module:
3.       * LoadFile(file) function:
3.1.    If file is authorized:
Read file
Store data in an appropriate data structure
Return data
3.2.    If not
Show an error message
3.3.    End If
4.      // Data Visualization Module:
5.       *CreateDashboard(data) function:
Create an interactive dashboard
View data statistics and graphs
View mission-relevant information
Flip the dashboard
6.      // Suspicious Transaction Detection Module:
7.       DetectAnomalies(data) function:
7.1.    For each transaction in the data:
Find the category (quantitative or qualitative) of the data
Apply the equivalent anomaly detection function
Apply anomaly detection rules and calculate the final score
Analyze the score for the transaction
If the score is above a threshold:
Mark transaction as suspicious
End If
7.2.    End For
7.3.    Return suspicious transactions
8.      // Main program
8.1.    File = GetFilePath()
8.2.    Data = LoadData.LoadFile(File)
8.3.    If Data is valid:
Dashboard = ViewData.CreateDashboard(Data)
Show (Dashboard)
SuspiciousTransactions = DetectSuspiciousTransactions.DetectAnomalies(Data)
Show (Suspicious Transactions)
IF Not
Show ("Error loading data.")
8.4.    End IF
9.      END
```

5. Experimental framework

For our experiment, we used two (2) datasets: on the one hand KDD-Cup99 HTTP[1] which was developed and published by Goldstein and Uchida 2016 [27], after some modifications to the dataset original data KDD-Cup99[2]. On the other hand, the Statlog Shuttle dataset [27] obtained by Goldstein after reducing the number of anomalies. These two datasets are widely used by the anomaly detection community as part of the comparative study of different methods.

KDD-Cup99 HTTP contains a standard set of data to audit, which includes a wide variety of simulated intrusions in a military network environment. It contains a total of 103,351 observations, 30 attributes and 176 anomalies (Observations). As for Statlog Shuttle, it includes 46,464 observations, 10 attributes and 878 anomalies. It can be used for supervised, semi-supervised and unsupervised anomaly detection methods.

We carried out our experiment on a Surface Pro 9, 64-bit computer with an Intel(R) Core (TM) i7-1065G7 CPU @ 1.30GHz 1.50 GHz, 16.0 GB RAM,

To evaluate our approach (SDAOS), we will compare it experimentally to three of the widely used anomaly detection methods, namely the LOF, iForest and MVIForest methods from different categories. To do this, we will use the same experimental data, namely the two data sets mentioned above.

6. Results and discussions

The results of the four (4) methods will be compared against the algorithm execution time and the following three metrics:

- The area under the ROC curve (ROC AUC) which is a standard in comparing the performance of anomaly detection methods.
- The Recall: allows you to know the proportion of real positive results that have been correctly identified. This is the metric to consider when the non-detection of an anomaly is important.

[1]). http://dx.doi.org/10.7910/DVN/OPQMVF

[2]). KDD Cup, 1999.KDD Cup. Data available: {http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html}; 1999.

$$Recall = \frac{TP}{TP+FN} \quad (16)$$

Specificity: is the rate of negative individuals correctly predicted by the model. It makes it possible to evaluate the performance of the method in detecting normal data.

$$Specificity = \frac{TN}{TN+FP} \quad (17)$$

With:

VP: True Positive (abnormal data) FP: False Positive (normal data)
VN: True Negative FN: False Negative

Positives represent abnormal data and negatives represent normal data.
Measuring the execution time of each algorithm will allow us to evaluate the degree of complexity of the algorithm.

Table 2 below summarizes the results of the 4 methods (LOF, iForest, MVIForest and our SDAOS proposal) on the two datasets considered (KDD-Cup99 http and Statlog Shuttle).

Table 2: Results of the 4 Methods with the KDD and Shuttle Datasets

DATASETS		ROC AUC	Recall	Specificity	TIME (S)CPU
LOF	KDD	0.82	0.91	0.69	37.90
	Shuttle	0.67	0.39	0.90	35.47
IForest	KDD	0.96	0.92	0.97	29.16
	Shuttle	0.97	0.97	0.95	9.43
MVIForest	KDD	0.98	0.92	0.98	25.02
	Shuttle	0.97	0,98	0.96	8.70
SDAOS	KDD	0.98	0.91	0.99	23.06
	Shuttle	0.96	0.96	0.97	8.09

Figures 2 and 3 show the comparison of the three (3) models to our model (SDAOS) according to the different metrics presented, using the KDD-Cup99 http dataset and the Shuttle dataset respectively.

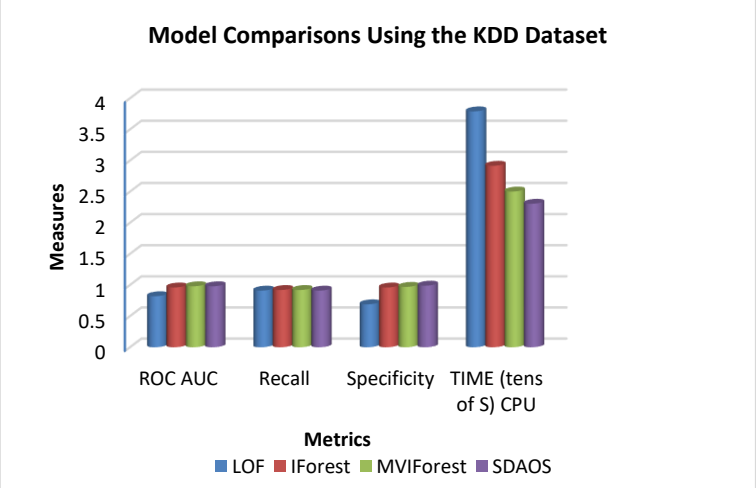

Fig. 2: Model Comparisons Using the KDD Dataset.

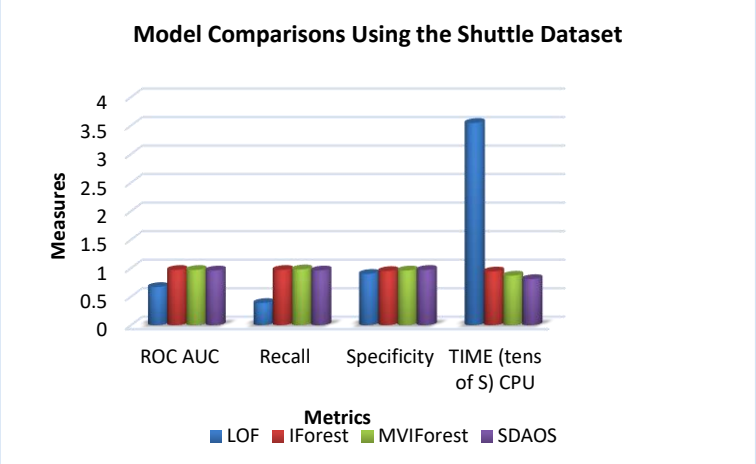

Fig. 3: Model Comparisons Using the Shuttle Dataset.

Observation of the results shows that for the area of the ROC curve as well as Recall, our model presents results almost similar to the IForest and MVIForest models, with a slight advantage of MVIForest in particular for Recall. This is explained by the fact that this model emphasizes the rate of well-classified abnormal data and considers the non-detection of an anomaly important.
Furthermore, with regard to Specificity and especially CPU execution time, our model offers the best results compared to other models. Indeed, this shows that on the one hand our model is more efficient in detecting normal data with a better rate of false positives. Furthermore, with both datasets, our model runs faster than the other models presented by offering the best response times. The effectiveness of our method is also explained by the fact that it has low complexity and also allows scalability thanks to the results obtained with the large dataset such as KDD-Cup99 http. Furthermore, unlike certain techniques which detect cases of fraud or anomalies by limiting themselves to selecting the most suspicious cases without clear and precise justification for these anomalies, our solution presents a report with some explanations on the different anomalies in order to facilitate the auditor's work.

7. Conclusion

In conclusion, this project offers an innovative approach to data management, particularly financial data, with a focus on detecting anomalies and suspicious transactions. The process begins with secure data loading, followed by a clear and detailed presentation through an informative dashboard. The application of specific rules by our algorithm then makes it possible to evaluate each transaction by assigning scores. At the end of this evaluation, our system identifies abnormal transactions which are then recorded in a knowledge base with the aim of strengthening the system's ability to effectively identify unusual patterns in future data to be analyzed.
The score-based approach provides visibility into transactions exhibiting deviant behavior, strengthening the ability to prevent fraud and ensure data integrity. This proposal represents a significant addition to audit tools, combining ease of use with good anomaly detection capability, paving the way for deeper analyzes and more informed decision-making.
The integration of artificial intelligence (AI) into the project offers a promising prospect by strengthening the capabilities of detecting anomalies and suspicious operations. Using machine learning techniques, AI can learn patterns from historical data and improve detection accuracy.
However, one of the limitations of our model, as of most anomaly detection algorithms, is the real-time processing of data streams at increasing flow rates and often without any prior knowledge of the data. In our next work, we will therefore focus on distributed detection algorithms in order to overcome this constraint.

References

[1] Hassid, Olivier. « Chapitre 1. Une histoire récente des risques en entreprise », Le management des risques et des crises. Dunod, 2011, pp. 7-36. https://doi.org/10.3917/dunod.hassi.2011.01.

[2] Hawkins, D.M. (1980). Multivariate outlier detection. In: Identification of Outliers. Monographs on Applied Probability and Statistics. Springer, Dordrecht. https://doi.org/10.1007/978-94-015-3994-4_8.

[3] P. C. Mahalanobis, « On the generalised distance in statistics », Proceedings of the National Institute of Sciences of India, vol. 2, no 1, 1936, p. 49–55

[4] Dang, T., Ngan, H. Y. T., & Liu, W. (2015). Distance-Based k -Nearest Neighbors Outlier Detection Method in Large-Scale Traffic Data, (February 2016). https://doi.org/10.1109/ICDSP.2015.7251924.

[5] Harris, P., Brunsdon, C., Charlton, M., Juggins, S., & Clarke, A. (2014). Multivariate Spatial Outlier Detection Using Robust Geographically Weighted Methods. Math Geosciences, 1–31. https://doi.org/10.1007/s11004-013-9491-0.

[6] Filzmoser, P., Ruiz-gazen, A., & Thomas-agnan, C. (2014). Identification of local multivariate outliers. https://doi.org/10.1007/s00362-013-0524-z.

[7] Rousseeuw, P. J., & Hubert, M. (2011). Robust statistics for outlier detection. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1(1), 73–79. https://doi.org/10.1002/widm.2.

[8] Chen, Y., Dang, X., Peng, H., & Bart, H. L. (2009). Outlier detection with the kernelized spatial depth function. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(2), 288–305. https://doi.org/10.1109/TPAMI.2008.72.

[9] Papadimitriou, S., Kitagawa, H., Gibbons, P. B., & Faloutsos, C. (2003). LOCI: fast outlier detection using the local correlation integral. Proceedings 19th International Conference on Data Engineering (Cat. No.03CH37405), 315–326. https://doi.org/10.1109/ICDE.2003.1260802.

[10] Chhabra, P., Scott, C., Kolaczyk, E. D., & Crovella, M. (2008). Distributed spatial anomaly detection. Proceedings – IEEE INFOCOM, 2378–2386. https://doi.org/10.1109/INFOCOM.2007.232.

[11] Ngan, H. Y. T., Lam, P., & Yung, N. H. C. (2016). Outlier Detection In Large-scale Traffic Data By Naïve Bayes Method and Gaussian Mixture Model Method, (February).

[12] Breunig, M. M., H.-P. Kriegel, R. T. Ng, et J. Sander (2000). Lof : identifying densitybased local outliers. In ACM sigmod record, Volume 29, pp. 93–104. ACM. https://doi.org/10.1145/335191.335388.

[13] Pokrajac, D., A. Lazarevic, et L. J. Latecki (2007). Incremental local outlier detection for data streams. In 2007 IEEE symposium on computational intelligence and data mining, pp. 504–515. IEEE. https://doi.org/10.1109/CIDM.2007.368917.

[14] Salehi, M., C. Leckie, J. C. Bezdek, T. Vaithianathan, et X. Zhang (2016). Fast memory efficient local outlier detection in data streams. IEEE Transactions on Knowledge and Data Engineering 28(12), 3246–3260. https://doi.org/10.1109/TKDE.2016.2597833.

[15] Lee, J. et N.-W. Cho (2016). Fast outlier detection using a grid-based algorithm. PloS one 11(11), e0165972. https://doi.org/10.1371/journal.pone.0165972.

[16] Schölkopf, B., J. C. Platt, J. Shawe-Taylor, A. J. Smola, et R. C. Williamson (2001). Estimating the support of a high-dimensional distribution. Neural computation 13(7), 1443–1471. https://doi.org/10.1162/089976601750264965.

[17] Schölkopf, B., R. C. Williamson, A. J. Smola, J. Shawe-Taylor, et J. C. Platt (2000). Support vector method for novelty detection. In Advances in neural information processing systems, pp. 582–588.

[18] Liu, F. T., K. M. Ting, et Z.-H. Zhou (2008). Isolation forest. In 2008 Eighth IEEE International Conference on Data Mining, pp. 413–422. IEEE. https://doi.org/10.1109/ICDM.2008.17.

[19] Liu, F. T., K. M. Ting, et Z.-H. Zhou (2012). Isolation-based anomaly detection. ACM Transactions on Knowledge Discovery from Data (TKDD) 6(1), 3. https://doi.org/10.1145/2133360.2133363.

[20] Yousra Chabchoub, Maurras Ulbricht TOGBE, Aliou Boly, Raja Chiky. An in-depth study and improvement of Isolation Forest. IEEE Access, 2022, vol. 10, p. 10219-10237. https://doi.org/10.1109/ACCESS.2022.3144425.

[21] Chawla, S., & Gionis, A. (2013). k -means– : A unified approach to clustering and outlier detection. https://doi.org/10.1137/1.9781611972832.21.

[22] Muller, E., Assent, I., Iglesias, P., Mulle, Y., & Bohm, K. (2012). Outlier Ranking via Subspace Analysis in Multiple Views of the Data. https://doi.org/10.1109/ICDM.2012.112.

[23] Filzmoser, P., Maronna, R., & Werner, M. (2007). Outlier identification in high dimensions. Filzmoser, P. Maronna, R. Werner, M. https://doi.org/10.1016/j.csda.2007.05.018.

[24] Harris, P., Brunsdon, C., Charlton, M., Juggins, S., & Clarke, A. (2014). Multivariate Spatial Outlier Detection Using Robust Geographically Weighted Methods. Math Geosciences, 1–31. https://doi.org/10.1007/s11004-013-9491-0.
[25] Kriegel, H., Kröger, P., Schubert, E., & Zimek, A. (2009). LoOP : Local Outlier Probabilities. Proceeding of the 18th ACM Conference on Information and Knowledge Management – CIKM '09, 1649. https://doi.org/10.1145/1645953.1646195.
[26] Zhang, Y., Meratnia, N., & Havinga, P. (2010). Outlier Detection Techniques for Wireless Sensor Networks: A Survey. IEEE Communications Surveys & Tutorials, 12(2), 1–12. https://doi.org/10.1109/SURV.2010.021510.00088.
[27] Goldstein, M. et S. Uchida (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. PloS one 11(4), e0152173. https://doi.org/10.1371/journal.pone.0152173.