Algorithms For Anomaly Detection on Time Series: A Use Case on Banking Data

Hugo Alatrista-Salas, Jeymi Fabiola Arias Hancco, Luis Espinoza-Villalobos
Escuela de Posgrado Newman, Tacna, Peru
E-mail: hugo.alatrista@epnewman.edu.pe, jeymifabiola.arias@epnewman.edu.pe,
uisenrique.espinoza@epnewman.edu.pe

Keywords: Anomaly detection, data mining, banking data

Received: May 30, 2024

The present research aims to present an overview of methods for automatically detecting anomalies in data representing time series. A time series is a sequence of qualitative values obtained at successive times, generally measured with equal intervals. Time series can represent different real-life phenomena, such as the behaviour of the stock market, variations in temperature and other meteorological data, the behaviour of banking credit/debit card consumption, among others. In addition, this work presents a 4-step methodology for preprocessing data and detecting anomalies on a time series dataset representing the spending of debit and credit card customers. A synthetic anomaly injection technique was applied to validate the models. Results can be used to monitor banking behaviour and trigger alarms in case of possible fraud or rare events.

Povzetek: Opisan je pregled metod za samodejno zaznavanje anomalij v časovnih vrstah s poudarkom na uporabi tehnik rudarjenja podatkov. Predstavljena je 4-stopenjska metodologija za predobdelavo podatkov in zaznavanje anomalij na naboru podatkov o bančnih transakcijah. Rezultati se lahko uporabijo za spremljanje bančnega vedenja in sprožitev alarmov v primeru morebitne goljufije ali redkih dogodkov.

1 Introduction

Anomaly detection is a data mining task aimed at identifying rare items, events or observations that are significantly different from the majority of the data and do not conform to some notion of normal behaviour.

A time series is a sequence of qualitative values obtained at successive points in time, usually measured at equal intervals. Time series can represent different real-world phenomena, including the behaviour of the stock market, variations in temperature data or other meteorological variables, the credit/debit card consumption of rights holders of a financial institution, the imports and exports of a country in a given period, and so forth.

This paper presents a literature review of methods for identifying anomalies in data. This review examines existing anomaly detection techniques for time series analysis. Then, this paper demonstrates the efficacy of these algorithms in analysing the spending of debit and credit card customers, represented by bank transactions made in 2016, which comprise approximately one million records representing the total spending per customer in various businesses in Lima, Peru. The data are grouped by date, representing a time series composed of a set of points or a sequence of points.

Subsequently, an anomaly detection model was constructed using data mining algorithms, which are varied and various strategies have been proposed. In this paper, four different techniques were compared: a) Isolation Forest, The following four techniques were evaluated for their effectiveness in detecting anomalies: IFO, LOF, kNN, and SVM. All four techniques demonstrated the ability to identify anomalies with varying degrees of success. However, each employs distinct methodologies. For instance, IFO, kNN, and SVM are supervised machine learning algorithms, whereas LOF is an unsupervised algorithm.

Although this application work is oriented to show a proof of concept, the applications of these techniques are vast and have been applied to different areas of knowledge, such as medicine [20], in the legal domain [13], in banking [88], in data associated with video surveillance [105], in fraud detection in financial institutions [77], among others.

This paper is structured as follows. Section 2 describes the problem surrounding anomaly detection on data representing time series. Section 3 presents the state of the art, while Section 4 details the experiments and results obtained for the use case selected in this work. The paper ends with the conclusions and future work presented in Section 5.

2 The problem of time-series anomaly detection.

Anomaly detection is a set of techniques and methods for the identification of rare or anomalous events. In other words, it is the search for patterns that do not follow expected behaviour. Frequently, anomalies and outliers have

been confused when performing anomaly detection tasks. The authors in [11] show the difference between outliers and anomalies. An outlier is associated with a single piece of data, whereas an anomaly is associated with events. To illustrate this distinction, consider a patient's age is 140 years, then it is an outlier. Conversely, if a patient systematically receives drugs that are not appropriate for them, this would be considered an anomalous event. Although the definition is generally clear and concise, it is contingent upon the structure of the data and the specific problem being addressed. The authors of [38] propose a categorisation of the data into three categories:

1. Metric data is the most common form of representation. In this type of data, the objects under study are described by their attributes, and they can be operationalised using distance and proximity.

2. Evolutionary data represent objects that are described in different time stamps. Within this type of data, discrete sequences, time series and multidimensional data streams can be identified.

3. Unstructured and semi-structured data, which do not have a rigid structure (e.g. textual data). In order to cope with these types of data, a pre-processing stage is necessary. This can be done using techniques such as the *bag-of-words* approach [1], TF, TF-IDF [102], TF-IGM [17], kf-idf [97], LSI [103], *Word2vec, Doc2vec* [22, 1], among others.

In this paper, we concentrate on evolutionary data or time series. Time series can be of different types. One of the most common classifications is to store the data continuously (data stored to the second, for one day) or discrete (monthly data pooling) [25].

Specifically, many algorithms for time series anomaly detection have been proposed in the literature. The choice of one algorithm depends mainly on the data to be analysed. However, some studies in the literature compare the most important algorithms [44, 91, 36, 65]. The paper [73] is one of the most interesting one, comparing various time-series anomaly detection techniques. This comparison includes algorithms such as the Isolation Forest [55] based on binary tree partitioning, the Local Outlier Factor [14] based on the distance of nearest neighbours, and the Histogram-based Outlier Score algorithm. Besides, the authors include in the comparison a method based on the calculation of the frequency histogram [28], the Matrix Profile method [99], which uses the distance of closed neighbours, the NORMA algorithm [12] based on data segmentation, the Principal Component Analysis (PCA) technique, an algorithm using autoencoders for the detection of anomalous events [81], the Polynomial Approximation technique [52] based on polynomial filters and the *One-class Support Vector Machine* technique [82] based on supervised learning using data containing a single class or label. Finally, the authors also include deep learning-based algorithms such as the Long Short Term Memory Networks Anomaly Detection LSTM-AD [61] and the convolutional neural networks [32, 66, 34].

Other algorithms that have also been used and compared in different studies are those based on the prediction algorithm or ARIMA *forecasting* [98, 63, 21]. All these algorithms were successfully used to tackle various problems, *e.g.*, fraud detection [41], detection of suspicious websites [70], DNA analysis and matching [86, 90], analysis of ECG signals [60], detection of suspicious transactions [56], in the analysis of court rulings [13], among others.

In addition to the algorithms mentioned above, there are comprehensive frameworks for time series analysis and anomaly detection, such as EGADS [43], FuxEv [47], Robusttad [26], or the package called Ostad [35].

In contrast to previous works, we compared classic anomaly detection algorithms to show their efficacy in mining time series data that represent the consumption of debit/credit card owners. Indeed, few papers use real banking data combined with simple but effective algorithms to identify anomalies that represent possible changes in the common behavior of banking account owners. In addition, we propose a pipeline to mine this kind of data and a deep discussion on validating the results using the injecting synthetic anomalies methodology. Finally, we provide a complete and comprehensive literature review on anomaly detection in time series data problems.

3 Literature review

Anomaly detection is a topic that has been studied for a long time. One of the first articles to mention anomaly detection using statistical techniques is described in [85]. Anomaly detection has been successfully applied in different domains, such as medicine [20], in precision agriculture [2], the legal domain [13], in banking [88], video surveillance [105], fraud detection [77] and many others.

The literature identifies different forms of anomalies: 1) anomalies consisting of a single observation or point; 2) collective anomalies consisting of a set of points or a sequence of points; 3) contextual anomalies referring to domain or context anomalies [67]. In addition, anomalies can be found in phenomena described by different types or sources of data, such as images [78], textual data [42], graphs [94], geolocated data [96], Markov chains [48], time series [11], among others. Regarding time series anomaly detection, anomalies can be detected from univariate time series [39, 104, 72] and multivariate time series [49, 9, 29]. This literature review will focus on univariate time series anomaly detection techniques.

Many strategies for time series anomaly detection have been implemented worldwide. Several authors have focused their efforts on presenting surveys and benchmarks of anomaly detection techniques and outliers in the financial sector, considering the use of supervised learning techniques, unsupervised learning techniques, statistical tech-

niques, deep learning, among others [3, 62, 68, 71]. In [92], for example, the authors classify these strategies into five main groups: Techniques based on statistical methods, techniques based on clustering and techniques based on deviation, those based on distance and those based on density. On the other hand, the authors in [16] mention the previous techniques, but add the methods based on supervised classification.

3.1 Machine learning techniques

The objective of *machine learning* techniques is to create a model that, from a set of data describing a set of objects, creates a function that allows us to know the classes or categories to which an object belongs [10]. For example, suppose we have a dataset associated with the problem of fraudulent customers and is composed of a group of individuals (objects) and the characteristics that describe them (education, age, balance, etc.). In such a case, machine learning algorithms will determine whether or not the individuals in question belong to a particular class (*i.e.*, fraudulent customers) based on their attributes. Machine learning algorithms are divided into two main groups: supervised learning or classification and unsupervised learning, clustering or segmentation [80].

Unsupervised learning techniques, also known as clustering are techniques in which the data does not have a defined class or category. That is, the dataset has the characteristics of the objects, but the class to which the data belongs is unknown. Then, clustering algorithms aim to group similar elements or objects into clusters. For this purpose, the algorithms compare all the features of the objects with each other in order to put together objects with similar features [4]. In the present research work context, many unsupervised learning techniques have been used to detect anomalies in time series. For example, in [51], the authors utilise an extension of the k-means algorithm - which uses the partitioning strategy - to detect anomalies in financial data. Conversely, other *clustering* algorithms employ the density strategy. In this context, in [59], the authors use the Optics algorithm to assess the creditworthiness of quarterly financial reports of a Vietnamese company. Finally, there are other so-called hierarchical clustering methods. In [5], the authors use these techniques to detect outliers and anomalies from financial data.

In contrast to unsupervised algorithms, supervised learning algorithms utilise data categorised into a specific class or category. In that sense, the algorithms build a function to predict the categories or classes of future individuals with no class [37]. Many supervised learning algorithms have been used to detect anomalies. For example, decision trees and random forest algorithms were used in the context of anomaly detection in financial data. In the same context, the authors in [6] use algorithms based on the Bayesian theorem to detect fraudulent activities. Similarly, the authors in [33] use the Support Vector Machine (SVM) technique and different kernels on bank transaction data from a financial institution in Indonesia. Conversely, several authors employ neural networks to detect anomalies [64, 79]. The latter concentrates on using perceptrons, the basic techniques of neural networks. However, in recent years, neural networks have been improved and have given rise to techniques known as deep learning or deep learning.

In the context of the application of deep learning techniques for anomaly detection, several works were proposed in the literature and used different algorithms, each time with better results because deep learning models directly learn feature representations from the original data, such as images and texts, without requiring the [57] attribute engineering stage.

3.2 Deep learning techniques

In recent years, deep learning techniques have revolutionised the field of time series anomaly detection. Early on, the authors in [74] combine *auto encoders* and PCA to tackle the problem of money laundering from data provided by the Brazilian Federal Tax Secretariat. Similarly, the authors of [15] propose a method that combines a one-dimensional convolutional neural network-based algorithm and a *clustering* anomaly identification algorithm on a dataset called Tennesse Eastman Benchmark. Similarly, the authors of [101] propose a method called Multi-Scale Convolutional Recurrent Encoder-Decoder (MSCRED) to identify anomalies in multivariate time series. The authors used a Convolutional Long-Short Term Memory (ConvLSTM) for the anomaly detection task.

In the same vein, transformer-based models have emerged as an effective technique for anomaly detection. These models use self-attention mechanisms to capture long-range dependencies and complex temporal patterns. The Temporal Fusion Transformer (TFT) is a notable model that combines high accuracy and interpretability [53, 76]. Recent studies have improved the original TFT by incorporating dynamic spatio-temporal attention mechanisms, enabling more accurate anomaly detection in multivariate time series data [89].

Graph Neural Networks (GNNs) have been increasingly applied to anomaly detection in time series data, especially when the data exhibit relational or network characteristics. Techniques such as Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs) have been adapted to time series analysis, for example to tackle the problem of fraud detection [8]. A more recent innovation is the Spatio-Temporal Graph Neural Network (STGNN), which models both spatial and temporal dependencies simultaneously and has proved highly effective in detecting anomalies in traffic and IoT sensor networks [23, 93].

Other authors also used algorithms that are typically used for synthetic data generation. For example, in [27], the authors used a *Generative Adversarial Network* (GAN) to generate synthetic data from real data and thereby identify potential anomalies in the data. The authors used the F1-score measure to evaluate their results, which were com-

pared with eight state-of-the-art algorithms. Similarly, the authors in [46] used a MAD-GAN architecture to discover anomalies in two real databases: a water treatment and distribution system and cyber-attack data. More recently, TimeGAN [31] and AnomalyGAN [58] have been proposed in the literature. The aforementioned models generate synthetic time series data that closely resemble the normal data distribution. Anomalies are detected by comparing real data points with the synthetic data and identifying those that deviate significantly.

Other works in the literature have used the VAEs for anomaly detection. Indeed, the Variational Autoencoders (VAEs) have gained popularity due to their ability to model complex data distributions. Extensions such as Temporal Convolutional VAEs have been developed to capture temporal dependencies in time series data. These models learn a probabilistic latent space, which can then be used to identify anomalies as deviations from the learned normal patterns. Recent improvements have focused on increasing the robustness and scalability of VAEs for multivariate time series datasets [95, 100].

Hybrid models that combine several deep learning techniques have also emerged as powerful tools for anomaly detection. For example, integrating GNNs with LSTM networks [106] or combining VAEs with transformers [18] can leverage the strengths of each approach. These hybrid models have shown superior performance on complex and high-dimensional time series datasets.

In addition, efforts have also been made to address the big data problem, in particular the problem of data streaming. For example, in [84], the authors use a combination of techniques, including Maximum Likelihood Estimation (MLE), to identify anomalies in data from 50 different companies and over 120 million continuously measured time-series metrics.

Finally, other authors focused not only on identifying anomalies in time series but also on visualising them. For example, in [40], the authors use the matrix growth technique to visualise anomalies in time series associated with financial data. Similarly, the authors in [54] propose the Viz-Tree and Diff-tree tools, which allow the visualisation of patterns in time series and their possible anomalies using structures known as *suffix-trees*. Likewise, the article [45] describes the tool called *DeepTimeAnomalyViz* (DeTAVIZ), which collates the outcomes of the anomaly detection algorithms (TABL and ResNet) and in post-processing visualises the results in a web environment.

Table 1 summarizes papers covered in this literature review (see Section 3). Studies were describeb by the type of the study (survey, application or both), the datased used in experimentations were performed, the methods used for detecting anomalies and metrics to the performance evaluation. Some acronyms used in this table are the follows: SVM Support Vector Machine, DT Decision Trees, AUC Area under the curve, EER Equal Error Rate, A Accuracy, P Precision, R Recall, FS F1-score, ML Machine Learning, RFGB Relational Functional Gradient Boosting, and MLN Markov Logic Networks, HMM Hidden Markov Model, DR Detection Rate, FAR False Alarm Rate, FCM Fuzzy C-Means, PSO Particle Swarm Optimization, WQ Weighted Q, SI Silhouette coefficient, DBI Davies-Bouldin Index, SD Standard Deviation, MAE Mean Absolute Error, RMSE Root Mean Squared Error, MSE Mean Squared Error, MAPE Mean Absolute Percentage Error, FID Frechet Inception Distance, LPIPS Learned Perceptual Image Patch Similarity, QL Quartile losses.

4 Use case and results

Several papers in the literature have attacked the anomaly detection problem from financial data. Specifically, the authors in [7] use bank (ATM) transaction data and compare three different algorithms (*t-side, kNN* and *HMM*). In this paper, we will analyse bank transaction data to detect anomalies using four algorithms widely used in the literature. PyCaret (`https://pycaret.org`) was used to build the model. PyCaret is a machine learning *open-source* Python library that focuses on automating machine learning workflows. To this aim, a 4-step methodology was proposed in this paper (see Figure 1). The steps are described below.

4.1 Banking dataset

The provider of the bank transaction data is a Peruvian financial institution that plays an important role in the Peruvian market. It is important to stress that this database is private and was obtained thanks to an academic-industry research project.

The banking transaction dataset is composed of records generated each time a rightful customer makes a banking transaction, *i.e.*, uses his or her debit or credit card to purchase an establishment or an electronic purchase. For this study, a table with 18 characteristics has been created. Table 2 describes the characteristics of the dataset.

Each record in the dataset is identified by an ID, which represents a unique bank transaction or credit/debit card payment (represented by the debit_type attribute). In addition, it consists of a unique customer ID that uniquely identifies a cardholder. This ID has been generated using anonymisation algorithms to maintain cardholders' privacy. On the customer side, we also have age and gender. In addition, each transaction ID is associated with a date, which represents the date the transaction took place. Furthermore, each bank transaction is associated with a merchant ID, indicating the merchant where the payment was made. Additionally, the address of the merchant, the Merchant Category Code (MCC), the amount of the transaction in soles and USD, the spatial characteristics of both the merchant and the bank branch where the cardholder opened their account, and the merchant's location on a map are all included. Finally, the geo-reference data allows merchants to be located on a map.

Table 1: List of the most representative papers covered in the present study.

Ref.	Study Type	Dataset	Techniques	Validation
[85]	Survey	-	-	-
[20]	Application	Genomics datasets	FRaC, JL proj, SVM, DT	AUC
[2]	Application	Greenhouse environment sensors	Autoencoders	A
[13]	Application	Criminal incidents dataset	Zero Modified Poisson	Anomaly injection
[88]	Application	Banking dataset	Egonet model	AUC
[105]	Application	Video surveillance	Anomalynet model	AUC, EER, P, R
[77]	Application/Survey	Real and synthetic datasets	Graph-based methods	AUC, P, R, FS
[67]	Application/Survey	22 datasets	29 ML models	AUC, P, R, FS
[78]	Application	Various datasets	NeuTraL AD	AUC, FS
[42]	Application	Flight incidents and textual datasets	RFGB, MLN	Weighted AUC
[94]	Application	CIDDS-001 and CICIDS-2017 datasets	Random Forest	P, R, FS
[96]	Application	Safecast and Kurama datasets	CVRAD and DBSCAN	Improved accuracy
[48]	Application	Various datasets	HMM-based approach	A, P, R, FS
[11]	Survey	-	-	-
[39]	Application	Numenta Benchmark + synthetic	Neural Networks	P, R, FS
[104]	Application	SMAP, MSL, TSA	Graph Attention Networks	P, R, FS
[72]	Application	Various datasets	Gaussian Process Regression	DR, FAR
[49]	Application	Various real and synthetic datasets	FCM, PSO	A
[9]	Application	Various datasets	UnSupervised Anomaly Detection (USAD)	P, R, FS
[29]	Application	10 datasets	19 unsupervised algorithms	AUC
[3]	Survey	-	-	-
[68]	Survey	-	-	-
[71]	Survey	-	-	-
[92]	Application	Various datasets	Graph Neural Networks	A
[16]	Survey	-	-	-
[68]	Survey	-	-	-
[4]	Application	Iris and Tennis datasets	Various techniques	A, Error rate
[51]	Application/Survey	10 financial datasets	k-Means combined Gaussian Mixture Models	WQ, SI, DBI
[59]	Application	Various datasets	Mahalanobis distance	Mean and SD
[5]	Application	Various datasets	Hierarchical Cluster	-
[37]	Application	Health datasets	Various techniques	A, P, R, AUC
[6]	Application	Financial datasets	Various techniques	A, P, R, FS
[33]	Application	Financial datasets	One-Class Support Vector Machine	A, P, R, FS
[33]	Application	Financial datasets	One-Class Support Vector Machine	A
[65]	Application	Various datasets	DeepAnT	FS, AUC
[79]	Application	Credit card transactions	Multi-layer perceptron	MAE, RMSE, A
[57]	Survey	-	-	-
[74]	Application	Financial transactions	AutoEncoder	MSE
[15]	Application	Tennessee Eastman Process	Clustering-augmented convolutional autoencoder	FS
[101]	Application	Synthetic and real datasets	MSCRED	FS
[53]	Application	Various datasets	Temporal Fusion Transformer	P50 and P90 QL
[76]	Application	Various datasets	Temporal Context Fusion Transformer	P, R, FS
[89]	Application	Trafic datasets	Spatiotemporal Fusion Transformer	MAE, RMSE, MAPE
[7]	Application	ATM datasets	Three sequence-based methods	FS
[23]	Application	Traffic datasets	ST-HGCN	A, FS
[93]	Application	Various datasets	Graph Neural Networks	A, Error rate
[27]	Application	Various datasets	TadGAN	FS
[46]	Application	Various datasets	Generative Adversarial Networks + LSTM	P, R, FS
[58]	Application	Image dataset	Generative Adversarial Networks	FID, LPIPS
[95]	Application	3 public datasets	MSCRVAE	P, R, FS
[100]	Application	4 public datasets	MSCVAE	P, R, FS
[106]	Application	5 real datasets	Graph Attention Network	P, R, FS
[84]	Application	Financial datasets	ARIMA + Holt-Winters + HMM	-
[40]	Visualisation	Financial datasets	Growth Matrix visualisation	-
[54]	Visualisation	AURSAD dataset	VizTree	FS
[45]	Visualisation	Telemetry dataset	DeepTimeAnomalyViz	-

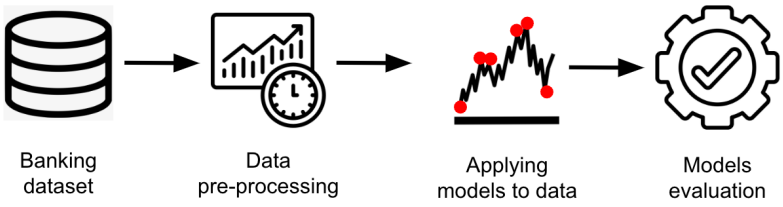

Figure 1: Methodology for detecting anomalies in time series dataset

Table 2: Description of bank transaction data set attributes

#	Name	Description
1	id	Transaction ID
2	client_id	Client ID
3	client_sex	Client sex
4	client_age	Client's age
5	date	Date of transaction
6	country_code	Country code
7	amount_sol	Amount in soles spent in the transaction
8	debit_type	Card type (debit, credit)
9	agency_departement	Agency department where the client opened their account
10	agency_province	Province of the agency where the client opened their account
11	agency_district	Agency district where the client opened their account
12	merchant_id	Merchant ID
13	merchant_departement	Department of the business where the purchase was made
14	merchant_province	Province of the business where the purchase was made
15	merchant_district	District of the business where the purchase was made
16	merchant_lon	Lambert longitude of the store where the purchase was made
17	merchant_lat	Lambert latitude of the store where the purchase was made
18	merchant_index	Merchant index

The dataset was collected between June 2016 and May 2017 and contains approximately 1.5 million customers, 55,000 different merchants, and 116.8 million transactions for both credit and debit cards across Peru.

In the dataset at our disposal, the provinces of Lima and Callao account for 46% of the total data (around 10 million transactions, see Figure 2). This distribution is consistent with the findings of the National Survey on the Demand for Financial Services and Level of Culture in Peru, conducted by the Superintendency of Banking and Insurance in 2016 [1]. The survey revealed that the majority of cards are concentrated in Lima and Callao, which is why this study was limited to these two provinces.

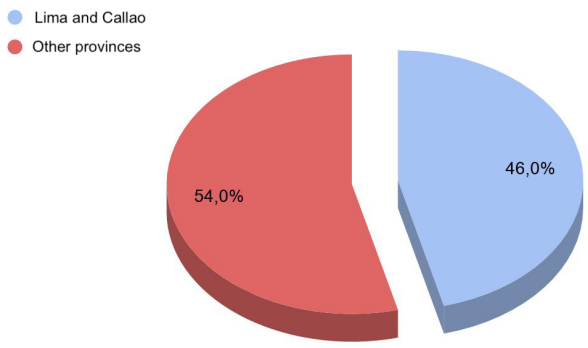

Figure 2: Proportion of banking transactions between Lima and Callao and the rest of the Peruvian territory

4.2 Data preprocessing

It should be noted that the bank transaction data described in Section 4.1 of this paper was utilised. The data set comprises over 10 million records (transactions), representing transactions carried out in Lima and Callao. These provinces of the department of Lima - capital of Peru - account for the most significant amount of consumption (approximately 46% of the total data), and are related to the results obtained by the National Survey on the Demand for

[1] https://www.sbs.gob.pe/Portals/0/jer/ESTUDIOS-SOBRE-INCLUSIÓN-FINANCIERA/Informe-de-Resultados.pdf

Financial Services and Level of Culture in Peru. The data were grouped to obtain approximately one million records, representing the sum of spending per customer and per day in Lima and Callao. These data were used to meet the objective of this research, namely to demonstrate a use case by implementing an algorithmic model that automatically identifies anomalies in data (bank transactions) that change over time (average transactions per day).

Table 3: Example of the dataset (time series)

Data	Amount_Soles
2016-11-01 05:04:17+00:00	126.22
2016-11-01 05:34:37+00:00	12.66
2016-11-01 05:35:38+00:00	74.50
2016-11-01 06:15:13+00:00	19.90
2016-11-01 06:18:33+00:00	383.60

Columns were then added to the data to control the time units, allowing us to represent the data value (bank transactions) from different time units. Table 4 shows these new columns.

From the data shown in the table, we are left with the attributes *Date* and *Amount_Soles*. These two attributes represent the time evolution of the average hourly expenditure of the financial institution's customers. Next, we will use four anomaly identification algorithms from the data at our disposal.

4.3 Applying models to data

This section shows the results obtained by running four supervised and unsupervised learning algorithms in our time series. In order to enhance clarity, the data for a single month (November) were presented graphically. However, the anomalies were identified using all the data available.

4.3.1 Isolation Forest - IFO

Isolation Forest is a data anomaly detection algorithm that detects anomalies in data using isolation [55]. The idea behind this algorithm is that it measures the distance separating an element or data from the rest of the data and, in this way, identifies an anomaly. To achieve this, IFO assumes that anomalous data exhibit two essential characteristics: they are few in number, *i.e.*, they are in the minority, and they are different, *i.e.*, they are markedly distinct from typical cases. By leveraging these two properties, IFO strives to isolate them. To this end, IFO employs multiple binary trees (Forest) and selects the optimal option from all the results. Regarding the parameters, the number of base estimators in the ensemble was fixed to 100, the contamination, *i.e.*, the proportion of outliers in the dataset, and the number of features were set to 1.0. The parameter bootstrap was set to true, *i.e.*, individual trees are fit on random subsets of the training data sampled with replacement. The rest of the parameters were set to default.

The Table 5 shows the results obtained by running the IFO algorithm on the data at our disposal. It illustrates the creation of two new columns by the method. The *Anomaly* column assigns a value of 1 if the data is anomalous and 0 if it is not. The *Anomaly value* column assigns a value to each record or data point. Values are considered anomalous if the value is close to 1 (positive). Conversely, the data is not anomalous as long as the value is smaller (negative). It is important to note that the values of the attribute *Anomaly* are assigned, taking into account the results obtained in the attribute *Anomaly value*. The results can be visualised using a line diagram. Figure 3 shows, in blue, the non-anomalous values and in red, the anomalous values.

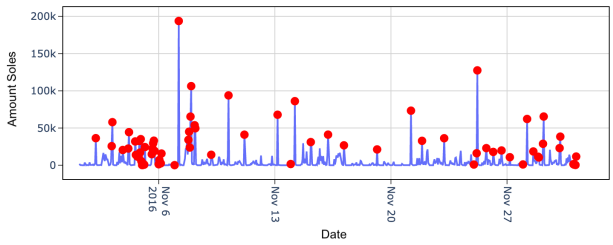

Figure 3: Visualisation of anomalies detected by the IFO algorithm for the month of November 2016

4.3.2 Local Outlier Factor - LOF

The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method. It calculates the deviation of the local density of a given element or data with respect to its nearest neighbours in the dataset. Once the deviation is calculated, the method considers samples that have a significantly lower density than their neighbours (with the help of a threshold) as outliers. As in the previous algorithm, the application of LOF on the study data adds two columns to the original data. The parameters used in KNN are as follows. The contamination value was fixed to 1.0. The leaf size was equal to 30, and it controls the minimum number of points in a given node. The distance computation used is the "Minkowski" distance, and the number of neighbours was fixed to 30. The value of p - the parameter of the "Minkowski" distance - was fixed to 2. Table 6 presents a subset of the results yielded by this algorithm.

In this case, the values in the *Anomaly value* column have different values from the previous results due to how the algorithm works. As seen in the Table 6, values that tend to zero (0) are considered anomalies, while larger values are considered average data.

Similar to the previous algorithm, we can visualise the results. This visualisation is crucial as it clearly shows normal values in blue and outliers in red (see Figure 4), helping us to differentiate between the two.

Table 4: Example of the dataset with new features

Date	Amount Soles	Day number	Day Day	Year day	Year week	Hour Hour	It is day of week?
2016-11-01 05:00:00+00:00	213.38	1	Tuesday	306	44	5	2
2016-11-01 06:00:00+00:00	1000.39	1	Tuesday	306	44	6	2
2016-11-01 07:00:00+00:00	179.80	1	Tuesday	306	44	7	2
2016-11-01 08:00:00+00:00	179.31	1	Tuesday	306	44	8	2
2016-11-01 09:00:00+00:00	18.90	1	Tuesday	306	44	9	2

Table 5: Example of results obtained by applying the IFO algorithm to the data

Date	Amount Soles	Anomaly	Anomaly value
2016-11-01 05:00:00+00:00	126.22	0	-0.128179
2016-11-01 06:00:00+00:00	12.66	0	-0.149509
2016-11-01 07:00:00+00:00	74.50	0	-0.129021
2016-11-01 08:00:00+00:00	19.90	0	-0.145932
2016-11-01 13:00:00+00:00	2334.68	1	0.033672

Table 6: Example of results obtained by applying the LOF algorithm to the data

Date	Amount Soles	Anomaly	Anomaly value
2016-11-01 05:00:00+00:00	126.22	0	1.073639
2016-11-01 06::00:00+00:00	12.66	0	1.544038
2016-11-01 07:00:00+00:00	74.50	0	0.991482
2016-11-01 08:00:00+00:00	19.90	0	177429.33
2016-11-01 13:00:00+00:00	150.99	1	1.071e+06

4.3.3 K-Nearest Neighbors Detector - kNN

Another supervised learning method is the k-Nearest Neighbours (kNN) algorithm. This algorithm was proposed in the 1960s and has since become one of the most widely used algorithms. This paper utilises a kNN-based algorithm to detect anomalies, which is called the K-Nearest Neighbours Detector. This algorithm has been used in real problems in different knowledge domains. Although kNN is a supervised algorithm, it employs an unsupervised approach, specifically in the context of anomaly detection. In kNN, the data are considered as points in a multidimensional (multiple features) space. The algorithm then evaluates the distance between points by considering their features. That is, the distance between them determines whether the points are similar or not. If the distance between the data point and its k nearest neighbours is sufficiently large, then the data point is considered anomalous. In this technique, k is an integer that generally takes odd values from 3 to 9. Similar to the previous algorithms, most of the default parameters were used. The k Value is fixed to 5 (as default), and the leaf size is fixed to 30. The "Minkowski" distance was used in this algorithm. Table 7 shows the results obtained with this algorithm.

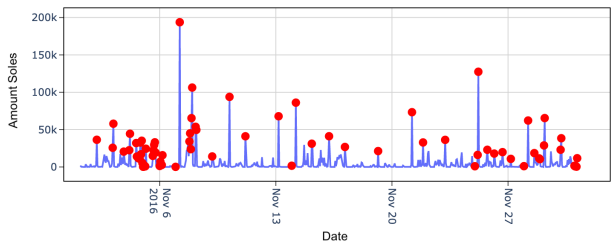

Figure 4: Visualisation of anomalies detected by the LOF algorithm for the month of November 2016

Table 7: Example of results obtained by applying the kNN algorithm to the data

Date	Amount Soles	Anomaly	Anomaly value
2016-11-01 05:00:00+00:00	126.22	0	0.000197
2016-11-01 06:00:00+00:00	12.66	0	0.000032
2016-11-01 07:00:00+00:00	74.50	0	0.000101
2016-11-01 08:00:00+00:00	19.90	0	0.000020
2016-11-01 20:00:00+00:00	3565.50	1	0.009267

We see in the Table 7 above that anomalies and non-anomalous data have similar values in the *Anomaly value* column. However, the algorithm defines a threshold σ and flags as anomalous data all those with values *Anomaly value*$> \sigma$. Finally, the results are shown visually in Figure 5.

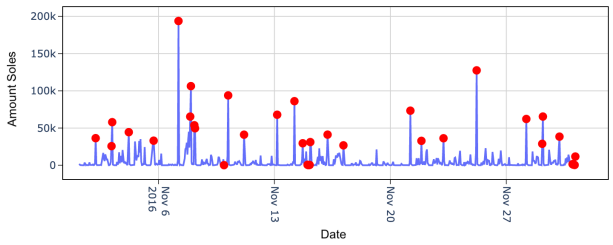

Figure 5: Visualisation of anomalies detected by the kNN algorithm for the month of November 2016

4.3.4 One-class SVM detector - SVM

The supervised machine learning algorithm, known as the Support Vector Machine (SVM), has been widely employed for the purpose of automatic anomaly detection [50]. In

this study, a version of the SVM algorithm is used, which has been optimised for the detection of anomalies, which we will refer to as the One-class SVM detector. In contrast to the classical SVM algorithm, the one-class SVM does not utilise target labels for model training. Indeed, the one-class SVM learns the boundary of the normal data points and identifies data outside this boundary from them and marks it as anomalous data. Regarding the algorithm parameters, the kernel cache size (in MB) was fixed to 200, and the coefficient was 0 (independent term in kernel function). The contamination was fixed to 1.0, the degree of the polynomial kernel function was fixed to 3, and the "rbf" was used as a kernel. The rest of the parameters were set to default.

Table 8: Example of results obtained by applying the SVM algorithm to the data

Date	Amount Soles	Anomaly	Anomaly value
2016-11-01 05:00:00+00:00	126.22	0	-5.005450
2016-11-01 06:00:00+00:00	12.66	0	2.731025
2016-11-01 07:00:00+00:00	74.50	0	-1.684504
2016-11-01 08:00:00+00:00	19.90	0	2.189123
2016-11-01 20:00:00+00:00	3565.50	1	453.284534

Table 8 presents the outcomes of the One-class SVM detector algorithm. As illustrated, the anomalous data correspond to instances that meet the condition *Anomaly Value* > 0 and normal values when the opposite is true. The results can be visually observed in Figure 6.

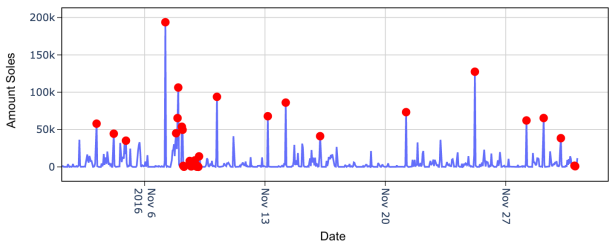

Figure 6: Visualisation of anomalies detected by the SVM algorithm for the month of November 2016

It is crucial to highlight that all figures present only a subset of the analysed data (the month of November 2016) in order to identify the anomalous data found by the algorithms visually. Indeed, since the data set lacks a class (anomalous/non-anomalous), the identification of anomalous values is conducted without the benefit of an *a-priori* hypothesis or prior knowledge of what is anomalous and what is not, as is the case in classical classification algorithms, where metrics such as accuracy, recall, and precision are employed. In this context, visual validation is essential.

4.4 Models evaluation

Validating anomaly detection results without a ground-truth dataset is challenging. Some documents in the literature use different strategies to validate anomalies found by unsupervised machine learning algorithms [29, 69, 24]. Metrics such us accuracy, precision, recall, F1, or the AUC-ROC (Area under the ROC Curve) are typically used in supervised learning with labelled data. However, for unsupervised anomaly detection, we can benefit from these metrics by creating pseudo-labels or by evaluating the method on synthetic datasets with injected known anomalies.

Specifically, in [69] the authors use this strategy. First, the authors create a synthetic dataset based on our real data and inject anomalies into it. As discussed in [30], three types of injection can be applied: random, contextual and collective. This first step provides a ground truth against which to validate. It is worth noting that the three types of anomaly injection have been tested in this work, and they show similar results. Once the ground truth was established, the unsupervised anomaly detection models were applied to the real and synthetic datasets. Anomaly scores were then calculated by assigning anomaly scores to the real data points. In this way, the synthetic data represents the ground truth labels, and for the real data, pseudo labels are generated by defining a threshold on the anomaly scores. Finally, quality metrics were calculated using the scores and the labels (or pseudo-labels).

Table 9: Quality measures obtained after applying the algorithms on the banking dataset

	SVM	KNN	LOF	IFO
Accuracy	0.8417	0.8333	0.8306	0.8083
Precision	0.3056	0.2222	0.1944	0.2361
Recall	0.1100	0.0800	0.0700	0.1700
F1-measure	0.3056	0.2222	0.1944	0.2361
AUC-ROC	0.5348	0.5174	0.5116	0.5406

Table 9 summarizes the results obtained by applying the Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms. Upon analyzing the results, we can see that accuracy values were relatively satisfactory, hovering around 0.8 (see Figure 7). The SVM is the algorithm with the highest accuracy value (0.8417). On the contrary, the precision, recall, F1 and AUC-ROC measurements were notably low, as depicted in Figures 8, 9, 10 and 11. In the following paragraphs, we discuss the underlying causes of the low values of these metrics.

A key factor contributing to the low precision, recall and F1 scores is the significant class imbalance inherent in anomaly detection tasks. The imbalance, where the number of anomalies is significantly smaller than the number of normal instances, leads to a skewed class distribution. This imbalance often results in a high number of false positives (normal points misclassified as anomalies) and false negatives (anomalies not detected), negatively affecting precision and recall.

The methodology for injecting synthetic anomalies also

plays a key role in the performance of the algorithms. Synthetic anomalies must accurately represent the actual anomalies in the real data. If they do not, the algorithms may not be able to detect them effectively.

Another important point is the diversity of the algorithms used in this study. Each has different recall, precision, F1 and AUC-ROC measures (w.r.t., accuracy). For example, the one-class SVM is sensitive to the choice of kernel, which affects its ability to detect anomalies. KNN relies on the density of data points and may underperform if anomalies are not clearly isolated. The Isolation Forest assumes that anomalies are few and distinct, which may not be true in all scenarios. Finally, the LOF algorithm is highly dependent on the choice of neighbourhood size, which affects its effectiveness. In this context, the performance of these algorithms is highly dependent on the parameters chosen.

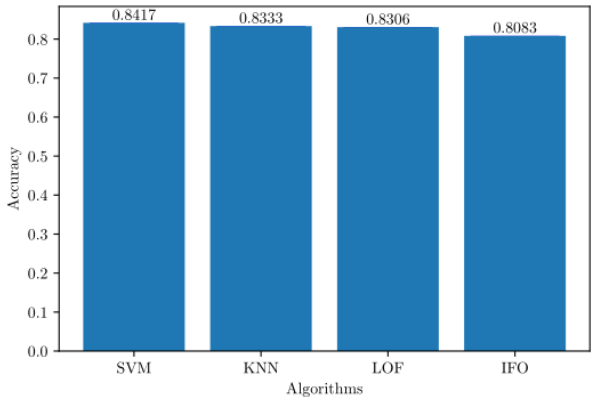

Figure 7: Accuracy for Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms

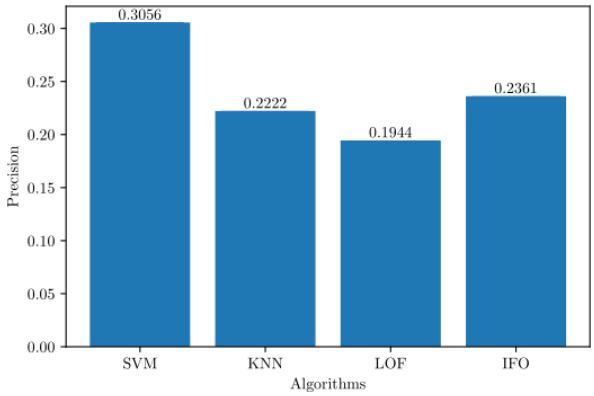

Figure 8: Precision for Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms

The problems discussed above can be addressed in several ways. First, a thorough optimisation of the parameters

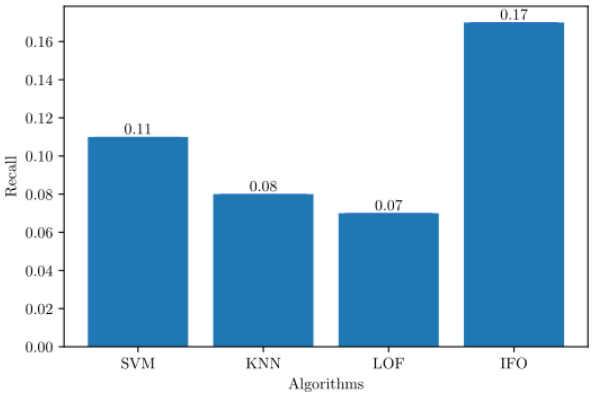

Figure 9: Recall for Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms

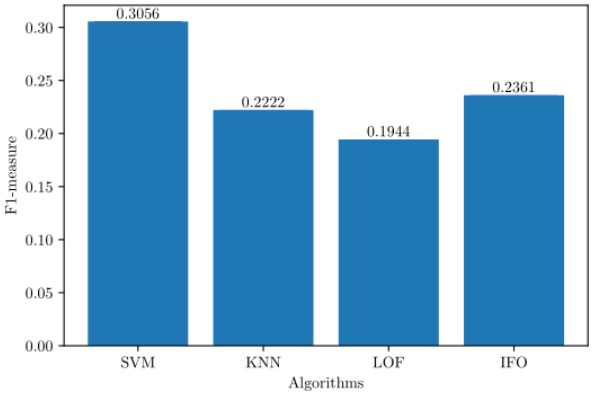

Figure 10: F1-measure for Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms

for each algorithm is crucial. Techniques such as grid or random search should be used to find the optimal set of parameters to ensure that each algorithm performs to the best of its ability. In addition, it is essential to address the class imbalance inherent in anomaly detection tasks. Using methods such as the Synthetic Minority Over-sampling Technique (SMOTE) [75] can help create a more balanced dataset. In addition, combining the outputs of multiple algorithms through ensemble methods such as Random Forest or Boosting can improve overall performance. Finally, incorporating domain-specific knowledge into the process is another effective strategy. This knowledge can guide the injection of realistic anomalies and help fine-tune the algorithms, making them more adept at identifying relevant anomalies.

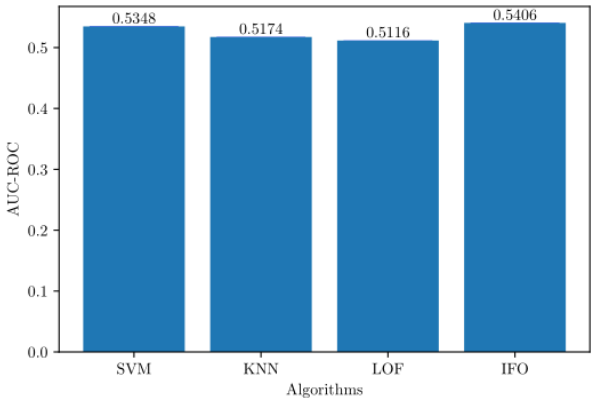

Figure 11: Area under the ROC curve (AUC-ROC) for Support Vector Machine - SVM, k Nearest Neighbors - kNN, the Local Outlier Factor - LOF and the Isolation Forest - IFO algorithms

4.5 Discussion

The detection of anomalies in banking transactions is paramount for enhancing the security and integrity of financial systems. Algorithms for anomaly detection, including both supervised and unsupervised methods, play a pivotal role in this process, as evidenced in [73]. Supervised algorithms, trained on labelled data, are effective at identifying known patterns of fraudulent activity [83]. In contrast, unsupervised algorithms excel at detecting new or emerging anomalies by recognising deviations from typical transaction behaviour [19]. These advanced techniques help banks proactively prevent fraud, ensuring that customer assets are protected and operational efficiency is improved through automation.

The importance of anomaly detection extends to risk management and regulatory compliance. By identifying unusual transaction patterns, banks can reduce risks associated with money laundering, terrorist financing, and other illegal activities [87]. Automated systems enable real-time monitoring, allowing banks to respond quickly to potential threats and maintain regulatory compliance. From a public policy perspective, implementing advanced anomaly detection systems in financial institutions is crucial for enhancing the security of the financial sector. Policymakers can create standardised guidelines for using these technologies, ensuring consistent and effective practices across the industry. Furthermore, addressing data privacy and ethical concerns is essential for maintaining public trust, requiring policies that ensure transparency and compliance with data protection laws.

Besides, public policies can promote research and innovation in anomaly detection technologies, driving the development of more sophisticated and accurate algorithms, reducing false positives and improving detection rates. These measures contribute to a more secure and stable financial system, benefiting both the industry and its customers.

Finally, the validation step of both supervised and unsupervised anomaly detection algorithms is crucial to ensure their effectiveness and reliability. In this work, unsupervised algorithms were used, which operate without predefined labels (*w.r.t.*, supervised algorithms). To take advantage of the quality metrics of supervised machine learning algorithms, a synthetic data series with injected anomalies was created and used as ground truth. This strategy allows the calculation of accuracy, precision, recall, F1 and area under the ROC curve. It is important to note that the last four metrics did not show satisfactory results due to the nature of the synthetic creation of unbalanced anomalies.

5 Conclusion and future directions

In this effort, we attack the problem of anomaly identification in time series. To this end, we present a comprehensive overview of the state of the art, including a detailed examination of various papers. Furthermore, we illustrate the efficacy of these techniques through the implementation of a use case. To this end, we utilise a dataset comprising over one million records of debit and credit card transactions. In the context of anomaly detection, we compare four techniques belonging to supervised and unsupervised machine learning methods. The effectiveness of all four techniques is demonstrated by calculating quality metrics such as accuracy, precision-recall, F1 and AUC-ROC using injection anomaly injection. However, the technique also has some limitations due to the creation of synthetic anomalies, which are usually few compared to the non-anomalous values.

In future work, it would be beneficial to compare the methods used in this project against those based on deep learning, such as convolutional neural networks. Furthermore, it would be advantageous to utilise other datasets to analyse the performance of these algorithms in other application areas, such as medicine or law.

References

[1] Haisal Dauda Abubakar, Mahmood Umar, and Muhammad Abdullahi Bakale. “Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec”. In: *SLU Journal of Science and Technology* 4.1&2 (2022), pp. 27–33.

[2] Mary Adkisson et al. “Autoencoder-based Anomaly Detection in Smart Farming Ecosystem”. In: *2021 IEEE International Conference on Big Data (Big Data)*. IEEE. 2021, pp. 3390–3399. https://doi.org/10.1109/BigData52589.2021.9671613.

[3] Mohiuddin Ahmed, Abdun Naser Mahmood, and Md Rafiqul Islam. "A survey of anomaly detection techniques in financial domain". In: *Future Generation Computer Systems* 55 (2016), pp. 278–288. https://doi.org/10.1016/j.future.2015.01.001.

[4] Ravinder Ahuja et al. "Classification and clustering algorithms of machine learning with their applications". In: *Nature-inspired computation in data mining and machine learning*. Springer, 2020, pp. 225–248. https://doi.org/10.1007/978-3-030-28553-1_11.

[5] JAS Almeida et al. "Improving hierarchical cluster analysis: A new method with outlier detection and automatic clustering". In: *Chemometrics and Intelligent Laboratory Systems* 87.2 (2007), pp. 208–217. https://doi.org/10.1016/j.chemolab.2007.01.005.

[6] Thushara Amarasinghe, Achala Aponso, and Naomi Krishnarajah. "Critical analysis of machine learning based approaches for fraud detection in financial transactions". In: *Proceedings of the 2018 International Conference on Machine Learning Technologies*. 2018, pp. 12–17. https://doi.org/10.1145/3231884.3231894.

[7] Maik Anderka et al. "Automatic ATM Fraud Detection as a Sequence-based Anomaly Detection Problem." In: *ICPRAM*. 2014, pp. 759–764. https://doi.org/10.5220/0004922307590764.

[8] Jay FK Au Yeung et al. "Jump detection in financial time series using machine learning algorithms". In: *Soft Computing* 24.3 (2020), pp. 1789–1801. https://doi.org/10.1007/s00500-019-04006-2.

[9] Julien Audibert et al. "Usad: Unsupervised anomaly detection on multivariate time series". In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020, pp. 3395–3404. https://doi.org/10.1145/3394486.3403392.

[10] Yalin Baştanlar and Mustafa Özuysal. "Introduction to machine learning". In: *miRNomics: MicroRNA biology and computational analysis* (2014), pp. 105–128. https://doi.org/10.1007/978-1-62703-748-8_7.

[11] Ane Blázquez-García et al. "A review on outlier/anomaly detection in time series data". In: *ACM Computing Surveys (CSUR)* 54.3 (2021), pp. 1–33. https://doi.org/10.1145/3444690.

[12] Paul Boniol et al. "Unsupervised and scalable subsequence anomaly detection in large data series". In: *The VLDB Journal* 30.6 (2021), pp. 909–931. https://doi.org/10.1007/s00778-021-00678-1.

[13] Jeffrey T Bordogna, Donald E Brown, and James H Conklin. "Design and implementation of an automated anomaly detection system for crime". In: *2007 IEEE Systems and Information Engineering Design Symposium*. IEEE. 2007, pp. 1–6. https://doi.org/10.1109/SIEDS.2007.4374005.

[14] Markus M Breunig et al. "LOF: identifying density-based local outliers". In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 93–104. https://doi.org/10.1145/335191.335388.

[15] Gavneet Singh Chadha et al. "Deep Convolutional Clustering-Based Time Series Anomaly Detection". In: *Sensors* 21.16 (2021), p. 5488. https://doi.org/10.3390/s21165488.

[16] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3 (2009), pp. 1–58. https://doi.org/10.1145/1541880.1541882.

[17] Kewen Chen et al. "Turning from TF-IDF to TF-IGM for term weighting in text classification". In: *Expert Systems with Applications* 66 (2016), pp. 245–260. https://doi.org/10.1016/j.eswa.2016.09.009.

[18] Ningjiang Chen et al. "Semisupervised anomaly detection of multivariate time series based on a variational autoencoder". In: *Applied Intelligence* 53.5 (2023), pp. 6074–6098. https://doi.org/10.1007/s10489-022-03829-1.

[19] Christos Cholevas et al. "Anomaly Detection in Blockchain Networks Using Unsupervised Learning: A Survey". In: *Algorithms* 17.5 (2024), p. 201. https://doi.org/10.3390/a17050201.

[20] Cyrus Cousins, Chirstopher M Pietras, and Donna K Slonim. "Scalable frac variants: Anomaly detection for precision medicine". In: *2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*. IEEE. 2017, pp. 253–262. https://doi.org/10.1109/IPDPSW.2017.148.

[21] Marco De Nadai and Maarten Van Someren. "Short-term anomaly detection in gas consumption through ARIMA and Artificial Neural Network forecast". In: *2015 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS) Proceedings*. IEEE. 2015, pp. 250–255. https://doi.org/10.1109/EESMS.2015.7175886.

[22] Roman Egger. "Text Representations and Word Embeddings". In: *Applied Data Science in Tourism*. Springer, 2022, pp. 335–361. https://doi.org/10.1007/978-3-030-88389-8_16.

[23] Jiangtao Feng et al. “Traffic Anomaly Detection base on spatio-temporal hypergraph convolution neural networks”. In: *Physica A: Statistical Mechanics and its Applications* (2024), p. 129891. https://doi.org/10.1016/j.physa.2024.129891.

[24] Antonino Ferraro et al. “Unsupervised Anomaly Detection in Predictive Maintenance Using Sound Data.” In: *SEBD*. 2023, pp. 449–458.

[25] Wayne A Fuller. *Introduction to statistical time series*. John Wiley & Sons, 2009.

[26] Jingkun Gao et al. “Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks”. In: *arXiv preprint arXiv:2002.09545* (2020). https://doi.org/10.48550/arXiv.2002.09545.

[27] Alexander Geiger et al. “TadGAN: Time series anomaly detection using generative adversarial networks”. In: *2020 IEEE International Conference on Big Data (Big Data)*. IEEE. 2020, pp. 33–43. https://doi.org/10.1109/BigData50022.2020.9378139.

[28] Markus Goldstein and Andreas Dengel. “Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm”. In: *KI-2012: poster and demo track* 9 (2012).

[29] Markus Goldstein and Seiichi Uchida. “A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data”. In: *PloS one* 11.4 (2016), e0152173. https://doi.org/10.1371/journal.pone.0152173.

[30] Nico Görnitz. *One-class Classification in the presence of Point, Collective, and Contextual Anomalies*. Technische Universitaet Berlin (Germany), 2019. https://doi.org/10.14279/depositonce-7897.

[31] Weixin Han et al. “Time series anomaly detection based on TimeGAN and LSTM neural network”. In: *2023 Eleventh International Conference on Advanced Cloud and Big Data (CBD)*. IEEE. 2023, pp. 122–127. https://doi.org/10.1109/CBD63341.2023.00030.

[32] Zilong He et al. “A spatiotemporal deep learning approach for unsupervised anomaly detection in cloud systems”. In: *IEEE Transactions on Neural Networks and Learning Systems* (2020). https://doi.org/10.1109/TNNLS.2020.3027736.

[33] Yaya Heryadi and Dandalina. “The effect of several kernel functions to financial transaction anomaly detection performance using one-class SVM”. In: *2019 International Congress on Applied Information Technology (AIT)*. IEEE. 2019, pp. 1–7. https://doi.org/10.1109/AIT49014.2019.9144956.

[34] Mohammad Kazim Hooshmand and Doreswamy Hosahalli. “Network anomaly detection using deep learning techniques”. In: *CAAI Transactions on Intelligence Technology* 7.2 (2022), pp. 228–243. https://doi.org/10.1049/cit2.12078.

[35] Alaiñe Iturria et al. “otsad: A package for online time-series anomaly detectors”. In: *Neurocomputing* 374 (2020), pp. 49–53. https://doi.org/https://doi.org/10.1016/j.neucom.2019.09.032.

[36] Vincent Jacob et al. “Exathlon: a benchmark for explainable anomaly detection over time series”. In: 14.11 (July 2021), pp. 2613–2626. ISSN: 2150-8097. https://doi.org/10.14778/3476249.3476307.

[37] Tammy Jiang, Jaimie L Gradus, and Anthony J Rosellini. “Supervised machine learning: a brief primer”. In: *Behavior Therapy* 51.5 (2020), pp. 675–687. https://doi.org/10.1016/j.beth.2020.05.002.

[38] Leonid Kalinichenko, Ivan Shanin, and Ilia Taraban. “Methods for anomaly detection: A survey”. In: *16th Russian Conference on Digital Libraries RCDL 2014 Proceedings. CEUR Workshop Proceedings 1297:20-25*. Vol. 1297. 2014, p. 2025.

[39] Jian-Bin Kao and Jehn-Ruey Jiang. “Anomaly detection for univariate time series with statistics and deep learning”. In: *2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*. IEEE. 2019, pp. 404–407. https://doi.org/10.1109/ECICE47484.2019.8942727.

[40] Daniel A Keim et al. “A spectral visualization system for analyzing financial time series data”. In: *Eurographics/IEEE TCVG Symposium on Visualization*. 2006, pp. 195–202. https://doi.org/10.2312/VisSym/EuroVis06/195-202.

[41] Yufeng Kou et al. “Survey of fraud detection techniques”. In: *IEEE International Conference on Networking, Sensing and Control, 2004*. Vol. 2. IEEE. 2004, pp. 749–754. https://doi.org/10.1109/ICNSC.2004.1297040.

[42] Raksha Kumaraswamy et al. “Anomaly detection in text: The value of domain knowledge”. In: *The Twenty-Eighth International Flairs Conference*. 2015.

[43] Nikolay Laptev, Saeed Amizadeh, and Ian Flint. “Generic and scalable framework for automated time-series anomaly detection”. In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015, pp. 1939–1947. https://doi.org/10.1145/2783258.2788611.

[44] Alexander Lavin and Subutai Ahmad. “Evaluating real-time anomaly detection algorithms–the Numenta anomaly benchmark”. In: *2015 IEEE 14th international conference on machine learning and applications (ICMLA)*. IEEE. 2015, pp. 38–44. https://doi.org/10.1109/ICMLA.2015.141.

[45] Błażej Leporowski, Casper Hansen, and Alexandros Iosifidis. “DeepTimeAnomalyViz: A Tool for Visualizing and Post-processing Deep Learning Anomaly Detection Results for Industrial Time-Series”. In: *arXiv preprint arXiv:2109.10082* (2021). https://doi.org/10.48550/arXiv.2109.10082.

[46] Dan Li et al. “MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks”. In: *International conference on artificial neural networks*. Springer. 2019, pp. 703–716. https://doi.org/10.1007/978-3-030-30490-4_56.

[47] Jia Li et al. “FluxEV: a fast and effective unsupervised framework for time-series anomaly detection”. In: *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 2021, pp. 824–832. https://doi.org/10.1145/3437963.3441823.

[48] Jinbo Li, Witold Pedrycz, and Iqbal Jamal. “Multivariate time series anomaly detection: A framework of Hidden Markov Models”. In: *Applied Soft Computing* 60 (2017), pp. 229–240. https://doi.org/10.1016/j.asoc.2017.06.035.

[49] Jinbo Li et al. “Clustering-based anomaly detection in multivariate time series data”. In: *Applied Soft Computing* 100 (2021), p. 106919. https://doi.org/10.1016/j.asoc.2020.106919.

[50] Kun-Lun Li et al. “Improving one-class SVM for anomaly detection”. In: *Proceedings of the 2003 international conference on machine learning and cybernetics (IEEE Cat. No. 03EX693)*. Vol. 5. IEEE. 2003, pp. 3077–3081. https://doi.org/10.1109/ICMLC.2003.1260106.

[51] Tie Li et al. “An integrated cluster detection, optimization, and interpretation approach for financial data”. In: *IEEE Transactions on Cybernetics* (2021). https://doi.org/10.1109/TCYB.2021.3109066.

[52] Zhi Li, Hong Ma, and Yongbing Mei. “A unifying method for outlier and change detection from data streams based on local polynomial fitting”. In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2007, pp. 150–161. https://doi.org/10.1007/978-3-540-71701-0_17.

[53] Bryan Lim et al. “Temporal fusion transformers for interpretable multi-horizon time series forecasting”. In: *International Journal of Forecasting* 37.4 (2021), pp. 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012.

[54] Jessica Lin et al. “Visually mining and monitoring massive time series”. In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2004, pp. 460–469. https://doi.org/10.1145/1014052.1014104.

[55] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation forest”. In: *2008 eighth ieee international conference on data mining*. IEEE. 2008, pp. 413–422. https://doi.org/10.1109/ICDM.2008.17.

[56] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation-based anomaly detection”. In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 6.1 (2012), pp. 1–39. https://doi.org/10.1145/2133360.2133363.

[57] Hongyu Liu and Bo Lang. “Machine learning and deep learning methods for intrusion detection systems: A survey”. In: *applied sciences* 9.20 (2019), p. 4396. https://doi.org/10.3390/app9204396.

[58] Ruikang Liu et al. “Anomaly-GAN: A data augmentation method for train surface anomaly detection”. In: *Expert Systems with Applications* 228 (2023), p. 120284. https://doi.org/10.1016/j.eswa.2023.120284.

[59] Mark Lokanan, Vincent Tran, and Nam Hoai Vuong. “Detecting anomalies in financial statements using machine learning algorithm: The case of Vietnamese listed firms”. In: *Asian Journal of Accounting Research* 4.2 (2019), pp. 181–201. https://doi.org/10.1108/AJAR-09-2018-0032.

[60] André Lourenço et al. “Outlier detection in non-intrusive ECG biometric system”. In: *International Conference Image Analysis and Recognition*. Springer. 2013, pp. 43–52. https://doi.org/10.1007/978-3-642-39094-4_6.

[61] Pankaj Malhotra et al. “Long short term memory networks for anomaly detection in time series”. In: *The European Symposium on Artificial Neural Networks*. Vol. 89. 2015, pp. 89–94.

[62] Kishan G Mehrotra, Chilukuri K Mohan, and HuaMing Huang. *Anomaly detection principles and algorithms*. Vol. 1. Springer, 2017. https://doi.org/10.1007/978-3-319-67526-8.

[63] H Zare Moayedi and MA Masnadi-Shirazi. "Arima model for network traffic prediction and anomaly detection". In: *2008 international symposium on information technology*. Vol. 4. IEEE. 2008, pp. 1–6. https://doi.org/10.1109/ITSIM.2008.4631947.

[64] Aji Mubalaike Mubarek and Eşref Adalı. "Multilayer perceptron neural network technique for fraud detection". In: *2017 International Conference on Computer Science and Engineering (UBMK)*. IEEE. 2017, pp. 383–387. https://doi.org/10.1109/UBMK.2017.8093417.

[65] Mohsin Munir et al. "DeepAnT: A deep learning approach for unsupervised anomaly detection in time series". In: *IEEE Access* 7 (2018), pp. 1991–2005. https://doi.org/10.1109/ACCESS.2018.2886457.

[66] Sheraz Naseer et al. "Enhanced network anomaly detection based on deep neural networks". In: *IEEE access* 6 (2018), pp. 48231–48246. https://doi.org/10.1109/ACCESS.2018.2863036.

[67] Ali Bou Nassif et al. "Machine learning for anomaly detection: A systematic review". In: *IEEE Access* 9 (2021), pp. 78658–78700. https://doi.org/10.1109/ACCESS.2021.3083060.

[68] Eric WT Ngai et al. "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature". In: *Decision support systems* 50.3 (2011), pp. 559–569. https://doi.org/10.1016/j.dss.2010.08.006.

[69] Jesse Nyyssölä and Mika Mäntylä. "Efficiency of Unsupervised Anomaly Detection Methods on Software Logs". In: *arXiv preprint arXiv:2312.01934* (2023). https://doi.org/10.48550/arXiv.2312.01934.

[70] Ying Pan and Xuhua Ding. "Anomaly based web phishing page detection". In: *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*. IEEE. 2006, pp. 381–392. https://doi.org/10.1109/ACSAC.2006.13.

[71] Guansong Pang et al. "Deep learning for anomaly detection: A review". In: *ACM Computing Surveys (CSUR)* 54.2 (2021), pp. 1–38. https://doi.org/10.1145/3439950.

[72] Jingyue Pang et al. "Anomaly detection based on uncertainty fusion for univariate monitoring series". In: *Measurement* 95 (2017), pp. 280–292. https://doi.org/10.1016/j.measurement.2016.10.031.

[73] John Paparrizos et al. "TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection". In: *Proceedings of the VLDB Endowment* 15.8 (2022), pp. 1697–1711. https://doi.org/10.14778/3529337.3529354.

[74] Ebberth L Paula et al. "Deep learning anomaly detection as support fraud investigation in brazilian exports and anti-money laundering". In: *2016 15th ieee international conference on machine learning and applications (icmla)*. IEEE. 2016, pp. 954–960. https://doi.org/10.1109/ICMLA.2016.0172.

[75] Russel Pears, Jacqui Finlay, and Andy M Connor. "Synthetic Minority over-sampling technique (SMOTE) for predicting software build outcomes". In: *arXiv preprint arXiv:1407.2330* (2014). https://doi.org/10.48550/arXiv.1407.2330.

[76] Xinggan Peng et al. "TCF-Trans: Temporal Context Fusion Transformer for Anomaly Detection in Time Series". In: *Sensors* 23.20 (2023), p. 8508. https://doi.org/10.3390/s23208508.

[77] Tahereh Pourhabibi et al. "Fraud detection: A systematic literature review of graph-based anomaly detection approaches". In: *Decision Support Systems* 133 (2020), p. 113303. https://doi.org/10.1016/j.dss.2020.113303.

[78] Chen Qiu et al. "Neural transformation learning for deep anomaly detection beyond images". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8703–8714.

[79] Smith Quintin-John and Raul Valverde. "A perceptron based neural network data analytics architecture for the detection of fraud in credit card transactions in financial legacy systems". In: *WSEAS Transactions on Systems and Control* 16 (2021). https://doi.org/10.37394/23203.2021.16.31.

[80] Gopinath Rebala, Ajay Ravi, and Sanjay Churiwala. *An introduction to machine learning*. Springer, 2019. https://doi.org/10.1007/978-3-030-15729-6.

[81] Mayu Sakurada and Takehisa Yairi. "Anomaly detection using autoencoders with nonlinear dimensionality reduction". In: *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*. 2014, pp. 4–11. https://doi.org/10.1145/2689746.2689747.

[82] Bernhard Schölkopf et al. "Support vector method for novelty detection". In: *Advances in neural information processing systems* 12 (1999).

[83] Philip Olaseni Shoetan et al. "Reviewing the role of big data analytics in financial fraud detection". In: *Finance & Accounting Research Journal* 6.3 (2024), pp. 384–394. https://doi.org/10.51594/farj.v6i3.899.

[84] Meir Toledano et al. "Real-time anomaly detection system for time series at scale". In: *KDD 2017 Workshop on Anomaly Detection in Finance*. PMLR. 2018, pp. 56–65.

[85] John W Tukey et al. *Exploratory data analysis*. Vol. 2. Reading, MA, 1977.

[86] Jung-Ying Tzeng et al. "Outlier detection and false discovery rates for whole-genome DNA matching". In: *Journal of the American Statistical Association* 98.461 (2003), pp. 236–246. https://doi.org/10.1198/016214503388619256.

[87] Sophia Beckett Velez. "Bank Secrecy Act Anti-Money Laundering Compliance Practices–Effective Practices". In: *Compliance and Financial Crime Risk in Banks*. Emerald Publishing Limited, 2024, pp. 125–142. https://doi.org/10.1108/978-1-83549-041-920241012.

[88] Yuan Wang, Liming Wang, and Jing Yang. "Egonet based anomaly detection in E-bank transaction networks". In: *IOP Conference Series: Materials Science and Engineering*. Vol. 715. 1. IOP Publishing. 2020, p. 012038. https://doi.org/10.1088/1757-899X/715/1/012038.

[89] Zhenghong Wang et al. "Spatiotemporal Fusion Transformer for large-scale traffic forecasting". In: *Information Fusion* 107 (2024), p. 102293. https://doi.org/10.1016/j.inffus.2024.102293.

[90] Baolin Wu. "Cancer outlier differential gene expression detection". In: *Biostatistics* 8.3 (2007), pp. 566–575. https://doi.org/10.1093/biostatistics/kxl029.

[91] Renjie Wu and Eamonn Keogh. "Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress". In: *IEEE Transactions on Knowledge and Data Engineering* (2021). https://doi.org/10.1109/ICDE53745.2022.00116.

[92] Hu-Sheng Wu. "A survey of research on anomaly detection for time series". In: *2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*. IEEE. 2016, pp. 426–431. https://doi.org/10.1109/ICCWAMTIP.2016.8079887.

[93] Yulei Wu, Hong-Ning Dai, and Haina Tang. "Graph neural networks for anomaly detection in industrial Internet of Things". In: *IEEE Internet of Things Journal* 9.12 (2021), pp. 9214–9231. https://doi.org/10.1109/JIOT.2021.3094295.

[94] Qingsai Xiao et al. "Towards network anomaly detection using graph embedding". In: *International Conference on Computational Science*. Springer. 2020, pp. 156–169. https://doi.org/10.1007/978-3-030-50423-6_12.

[95] Tianming Xie, Qifa Xu, and Cuixia Jiang. "Anomaly detection for multivariate times series through the multi-scale convolutional recurrent variational autoencoder". In: *Expert Systems with Applications* 231 (2023), p. 120725. https://doi.org/10.1016/j.eswa.2023.120725.

[96] Yanan Xin. "Anomaly detection for volunteered geographic information: A case study of Safecast data". In: *International Journal of Geographical Information Science* 36.7 (2022), pp. 1423–1442. https://doi.org/10.1080/13658816.2021.1981333.

[97] Feiyu Xu et al. "A Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping". In: *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*. Ed. by Manuel González Rodríguez and Carmen Paz Suarez Araujo. Las Palmas, Canary Islands - Spain: European Language Resources Association (ELRA), May 2002. URL: https://aclanthology.org/L02-1351/.

[98] Asrul H Yaacob et al. "Arima based network anomaly detection". In: *2010 Second International Conference on Communication Software and Networks*. IEEE. 2010, pp. 205–209. https://doi.org/10.1109/ICCSN.2010.55.

[99] Chin-Chia Michael Yeh et al. "Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile". In: *Data Mining and Knowledge Discovery* 32.1 (2018), pp. 83–123. https://doi.org/10.1007/s10618-017-0519-9.

[100] Umaporn Yokkampon et al. "Robust unsupervised anomaly detection with variational autoencoder in multivariate time series data". In: *IEEE Access* 10 (2022), pp. 57835–57849. https://doi.org/10.1109/ACCESS.2022.3178592.

[101] Chuxu Zhang et al. "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 1409–1416. https://doi.org/10.1609/aaai.v33i01.33011409.

[102] Wen Zhang, Taketoshi Yoshida, and Xijin Tang. "A comparative study of TF* IDF, LSI and multi-words for text classification". In: *Expert systems with applications* 38.3 (2011), pp. 2758–2765. https://doi.org/10.1016/j.eswa.2010.08.066.

[103] Wen Zhang, Taketoshi Yoshida, and Xijin Tang. "TFIDF, LSI and multi-word in information retrieval and text categorization". In: *2008 IEEE International Conference on Systems, Man and Cybernetics*. IEEE. 2008, pp. 108–113. https://doi.org/10.1109/ICSMC.2008.4811259.

[104] Hang Zhao et al. "Multivariate time-series anomaly detection via graph attention network". In: *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2020, pp. 841–850. https://doi.org/10.1109/ICDM50108.2020.00093.

[105] Joey Tianyi Zhou et al. “Anomalynet: An anomaly detection network for video surveillance”. In: *IEEE Transactions on Information Forensics and Security* 14.10 (2019), pp. 2537–2550. https://doi.org/10.1109/TIFS.2019.2900907.

[106] Liwen Zhou, Qingkui Zeng, and Bo Li. “Hybrid anomaly detection via multihead dynamic graph attention networks for multivariate time series”. In: *IEEE Access* 10 (2022), pp. 40967–40978. https://doi.org/10.1109/ACCESS.2022.3167640.