information

MDPI

Article

Applied Machine Learning to Anomaly Detection in Enterprise Purchase Processes: A Hybrid Approach Using Clustering and Isolation Forest

Antonio Herreros-Martínez [1], Rafael Magdalena-Benedicto [1,*], Joan Vila-Francés [1], Antonio José Serrano-López [1], Sonia Pérez-Díaz [2] and José Javier Martínez-Herráiz [3]

[1] Electrical Engineering Department, University of Valencia, Ave Universitats, s/n. Burjassot, 46100 Valencia, Spain
[2] Physics and Mathematics Department, University of Alcalá de Henares, Campus Tecnológico Universitario, Alcalá de Henares, 28805 Madrid, Spain; sonia.perez@uah.es
[3] Computer Science Department, University of Alcalá de Henares, Campus Tecnológico Universitario, Alcalá de Henares, 28805 Madrid, Spain
* Correspondence: rafael.magdalena@uv.es

Abstract: In the era of increasing digitalisation, organisations face the critical challenge of detecting anomalies in large volumes of data, which may indicate suspicious activities. To address this challenge, audit engagements are conducted regularly, and internal auditors and purchasing specialists seek innovative methods to streamline these processes. This study introduces a methodology to prioritise the investigation of anomalies identified in two large real-world purchase datasets. The primary objective is to enhance the effectiveness of companies' control efforts and improve the efficiency of anomaly detection tasks. The approach begins with a comprehensive exploratory data analysis, followed by the application of unsupervised machine learning techniques to identify anomalies. A univariate analysis is performed using the z-Score index and the DBSCAN algorithm, while multivariate analysis employs k-Means clustering and Isolation Forest algorithms. Additionally, the Silhouette index is used to evaluate the quality of the clustering, ensuring each method produces a prioritised list of candidate transactions for further review. To refine this process, an ensemble prioritisation framework is developed, integrating multiple methods. Furthermore, explainability tools such as SHAP are utilised to provide actionable insights and support specialists in interpreting the results. This methodology aims to empower organisations to detect anomalies more effectively and streamline the audit process.

Keywords: anomaly detection; artificial intelligence; business intelligence; fraud detection; machine learning

Academic Editor: Aneta Poniszewska-Maranda

Received: 13 January 2025
Revised: 13 February 2025
Accepted: 24 February 2025
Published: 26 February 2025

Citation: Herreros-Martínez, A.; Magdalena-Benedicto, R.; Vila-Francés, J.; Serrano-López, A.J.; Pérez-Díaz, S.; Mártinez-Herráiz, J.J. Applied Machine Learning to Anomaly Detection in Enterprise Purchase Processes: A Hybrid Approach Using Clustering and Isolation Forest. *Information* **2025**, *16*, 177. https://doi.org/10.3390/info16030177

1. Introduction

The Internal Audit department of a company (normally multinationals groups and/or big-sized entities) aims to ensure the correctness and effectiveness of the entities' processes, their compliance to the approved internal policies and to reduce risks in any form that could be presented [1]. In order to achieve this goal, the companies' internal teams conduct audits on a regular basis defined as audit engagements. During their missions, the auditors identify, evaluate and document adequate information to achieve the objectives of the engagement [2], carrying out interviews with the auditees and performing a rigorous tracking of evidence supporting the audit findings.

Currently, auditing still mainly relies on sampling the information (registers, transactions, etc.) to assess the processes' compliance during the audit engagements [3]. Consequently, the so-called sampling-risk result in the fact that relevant information in the registers/transactions could remain out of the sampling selection to be reviewed. Additionally, with the growing amount of data, this traditional approach becomes obsolete, and the sampling risk is aggravated [4].

Among the business processes, a special interest resides in searching for anomalies or misbehaviours on purchases. Internal audit and purchase managers need to prospect, evaluate and select the methodologies and IT tools capable of monitoring expenses and discovering relevant information that can highlight an out-of-policy act or, even, fraud [5,6]. The goal is to automate processes within the company that help to prioritise the investigation activities according to the level of suspicion of any fact.

In this business context, data analytics has been revealed as a key element in supporting organisations in the challenge of processing and controlling huge quantities of information. Among the various algorithms available, those related to machine learning are known to offer good results in finding relevant insights over structured and unstructured data. Machine learning and data analytics are changing the auditing approaches [7] and are becoming an important part of the control toolbox for companies.

Prior studies in audit analytics [8] rely on supervised learning, which is infeasible for unlabelled procurement data. Recent work [9] demonstrated Isolation Forest's effectiveness in fraud detection, while k-Means clustering has been widely adopted for interpretability in audit prioritisation [10]. However, no prior work combines these methods for procurement anomalies, nor addresses the challenges of mixed categorical/numeric features in this domain.

A key drawback to full population tests, based on hand-crafted rules, is that they will likely be able to find only errors, mistakes or deviations which were already expected or anticipated. Machine learning approaches have the ability to go beyond the results obtained through hand-crafted rules. They can be used to find anomalies in large amounts of data [7] allowing for a later identification of errors, mistakes, circumvented processes or even fraud. Most of the machine learning approaches that have been tested in auditing are based on supervised learning [11,12]. In this technique, an algorithm learns to map input data to known targets (labelled data).

In external auditing, due to reporting regulations, data share a similar structure across multiple companies (e.g., financial statements). This fact makes easier to obtain data labels for training supervised machine learning models. Regarding internal auditing, on the contrary, detecting anomalies in enterprise purchase processes poses unique challenges: (i) Unlabelled Data: Unlike financial fraud detection, internal audit datasets lack ground-truth labels for supervised learning; (ii) High Dimensionality: Mixed categorical/numeric features (e.g., requester, department, purchase amount) complicate distance-based methods; (iii) Contextual Variability: Anomalies are process-dependent (e.g., deviations in approval workflows vs. vendor selection); and (iv) Dynamic Policies: Evolving procurement guidelines require models to adapt without historical labels. The unsupervised algorithms offer a promising choice in this context. Within this algorithm's category, and among others, clustering techniques have been widely studied in anomaly detection [8,13]. These challenges motivated our hybrid approach, combining unsupervised clustering, Isolation Forest and univariate methods to address diverse anomaly types.

The goals of the present work are as follows:

i Investigate and propose a methodology able to detect anomalies in an actual unlabelled purchase dataset;

ii Prioritise the cases to be analysed by the organisation's specialists;

iii Provide them an adequate explicability technique of such cases to support their investigation.

2. Purchase Business Process

Most businesses rely on other businesses for products and services that help them with their operations. The purchasing process enables an organisation to evaluate these business-to-business transactions for efficiency and track their spending. It must be distinguished from the procurement process. Procurement focuses on strategies, such as negotiations and researching, while purchasing focuses on the actual act of buying products and services. Despite the control measures, an inadequate purchase process management can overlook more deliberate harmful practices (i.e., purchase fraud) and cause the enterprise losses [14]. An effective purchasing process can help prevent theft, fraud or irregular spending since it requires documenting all business transactions.

Unlike generic fraud detection, anomalies in procurement workflows are contextually tied to process stages (e.g., approval bypass, vendor collusion). For example:

- Approval Bypass: Orders approved by unauthorised personnel (detected via requester–approver mismatch).
- Vendor Collusion: Repeated purchases from non-preferred vendors (flagged via clustering and frequency analysis).

Our methodology targets these domain-specific anomalies by encoding workflow metadata (e.g., approval hierarchy) into features like "requester–buyer pairs".

3. Anomalies Detection

Anomaly detection is the process of finding outliers in a given dataset. Outliers are the data objects that stand out amongst other data objects and do not conform to the expected behaviour in a dataset. Anomaly detection algorithms have broad applications in business, scientific and security domains where isolating and acting on the results of outlier detection is critical [15]. For the identification of anomalies, there are many different algorithms, such as classification, regression and clustering. In case of having elements with known anomalous behaviours, supervised algorithms can be used for anomaly detection. In addition to supervised algorithms, there are specialised (unsupervised) algorithms whose whole purpose is to detect outliers without any previous knowledge about the dataset [16].

Anomalies can be classified into three categories [17]:

- Point Anomalies. Whether an individual data instance can be considered as anomalous with respect to the rest of the data.
- Contextual Anomalies. Whether a data instance is anomalous in a specific context, but not otherwise. For instance, within a cluster, the points more distant from the centroid can be considered anomalies within the cluster but not in the entire dataset.
- Collective Anomalies. Whether a collection of related data instances is anomalous with respect to the entire dataset or not. For instance, all the elements belonging to a group in a classification problem.

3.1. Anomaly Detection Techniques (ADTs)

3.1.1. Outlier Detection Using Statistical Methods

Outliers in the data can be identified by creating a statistical distribution model of the data and identifying the data points that do not fit into the model or data points that occupy the ends of the distribution tails [18]. Outliers can be detected based on where the data points fall in the standard normal distribution curve. A threshold for classifying

an outlier is specified. Other statistical techniques take multiple dimensions into account, computing other distance metrics (multivariate anomaly).

3.1.2. Outlier Detection Using Data Science

Normally, outliers show a specific set of characteristics that can be exploited to locate them. According to the unique set of characteristics that identify an outlier, the detection techniques are as follows:

- Distance-based: By nature, outliers are distant from other data points. If the average distance of the nearest N neighbours is measured, the outliers will have a higher value than other normal data points. Distance-based algorithms utilise this property to identify outliers in the data.
- Density-based: The density of a data point in a neighbourhood is inversely related to the distance to its neighbours. Consequently, outliers are located in low-density areas, while the regular data points are concentrated in high-density areas.
- Distribution-based: Outliers are the data points that have a low probability of occurrence, and they occupy the tail ends of the distribution curve.
- Clustering: Outliers, by definition, are not similar to normal data points in a dataset. The outliers can be the lone data points that are not clustered or the outliers forming a far and low-populated cluster.
- Classification techniques: If labelled data are previously available, a classification model can be built based on a training (labelled) dataset and tested through the prediction of an unknown dataset (not labelled). The challenge in using a classification model is the availability of previously labelled data.

Each of these techniques has multiple parameters and, hence, a data point labelled as an outlier in one algorithm may not be an outlier in another one. Thus, it is a good practice to rely on multiple algorithms before labelling the outliers. This approach is applied in the present work which, considering the problem statement and the unlabelled input data, will be focused on clustering (k-Means) and tree-based techniques (Isolation Forest). k-Means and Isolation Forest were selected as complementary techniques: k-Means provides interpretable clusters for business stakeholders, despite lower Silhouette scores. In the other hand, Isolation Forest efficiently flags global outliers missed by clustering. This hybrid approach mitigates weaknesses of standalone methods (e.g., k-Means' spherical cluster assumption, Isolation Forest's blindness to contextual anomalies).

3.2. Algorithms

3.2.1. k-Means

k-Means clustering is a distance-based clustering method where the dataset is divided into k number of clusters. It is one of the simplest and most commonly used clustering algorithms. In this technique, the user specifies the number of clusters (k) that the dataset need to be grouped in. The objective of k-Means clustering is to find a prototype data point (centroid) for each cluster; all the data points are then assigned to the nearest prototype, which then forms a cluster. The centre of the cluster can be the mean of all data objects in the cluster, as in k-Means, or the most represented data object, as in k-medoid clustering [19]. The evaluation of k-Means clustering has been performed using two methods:

1. Computing the total of the Sum Squared Errors (SSE) or Sum of Squared Quantisation Errors (SSQ), where good models will have low SSE/SSQ within the cluster and low overall SSE/SSQ among all clusters. Both SSE and SSQ are related to the k-Means clustering algorithm and are often used interchangeably. SSE measures the total squared distance between each data point and its assigned cluster centroid. SSQ is another term for the sum of squared distances from points to their centroids. In

k-Means, SSQ is essentially the same as SSE. The term "quantisation error" is used more in signal processing and vector quantisation contexts, but in clustering, it represents how well data points are assigned to their clusters. The heuristic criterion used to determine the optimal number of clusters in a dataset has consisted of representing the SSE/SSQ as a function of the number of clusters (k), the so-called Elbow curve, and choosing the cut-off point where the diminishing SSE/SSQ is no longer worth the additional number of clusters [20].

2. The Silhouette coefficient, proposed by Peter Rousseeuw in 1987 [21]. The Silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The Silhouette ranges from −1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighbouring clusters. If most objects have a high value, then the clustering configuration is appropriate. If many points have a low or negative value, then the clustering configuration may have too many or too few clusters. Hyperparameter tuning for k-Means focused on selecting the optimal number of clusters (k). We tested k = 2–25 using the Elbow method and Silhouette analysis. The final k values (8 for Company1, 5 for Company2) were chosen based on the trade-off between interpretability (lower k for fewer clusters) and anomaly granularity (higher k for smaller clusters). Gaussian normalisation (Z-score) was applied to mitigate feature scale bias.

3.2.2. DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a density-based clustering algorithm, first proposed in 1996 [22] and later revisited in 2017 [23]. DBSCAN is one of the most widely used density-based clustering algorithms for unsupervised data clustering [24,25].

The method identifies three types of points in a dataset:

- Core Points: These are points that have at least a minimum number of neighbours (defined by the MinPts parameter) within a specified radius (defined by the eps parameter).
- Border Points: These are points that are within the eps radius of a core point but do not have enough neighbours to be considered core points themselves (i.e., they have fewer than MinPts neighbours).
- Noise Points: These are points that are neither core points nor border points. They are typically considered outliers or noise in the dataset.

The method groups dense clouds of data points into clusters and any isolated points are considered not part of clusters and are treated as noises. The algorithm starts out by randomly selecting a starting point. If there are a sufficiently large number of points within the neighbourhood around that point, then those points are considered as part of the same cluster as the starting point (called direct density reachable). The neighbourhoods of the newly added points are then examined. If there are data points within these neighbourhoods, then those points are also added to the cluster (called density reachable). This process is repeated until no more points can be added to this particular cluster. Then, another point is randomly selected as a starting point for another cluster, and the cluster formation process is repeated until no more data points are available to be assigned to clusters. If data points are not within the neighbourhood of any other data points, those data points are considered as noises.

According to the last algorithm revisit published by its authors [23], the parameter MinPts can be set to the default value (4) while other studies [26] suggest setting it to twice the dataset dimensionality (in our univariate case, dim = 1). As a compromised criterion, the algorithm was applied with the default value in KNIME (MinEps = 3). Concerning

the distance, due to the way DBSCAN is designed, eps should be chosen as small as possible [23]. In [26], it is suggested to use as value $(2 \cdot \dim - 1)$, so we set value 1 as the epsilon-specific distance. Those transactions that do not have at least three neighbours within a distance of the unity (Core Points) nor are within a distance of the unity from a Core Point (but have less than three neighbours), have been flagged as an outlier.

3.2.3. Isolation Forest

The Isolation Forest [9] algorithm takes advantage of two quantitative properties of anomalies: (i) they are the minority consisting of few instances, and (ii) they have attribute values that are very different from those of normal instances. These characteristics make them more susceptible to being separated (isolated) from normal instances. The isolation process is based on a binary tree structure called the isolation tree (iTree). Because of the susceptibility to isolation, anomalies are more likely to be isolated closer to the root of an iTree, whereas normal points are more likely to be isolated at the deeper end of an iTree. The Isolation Forest method (iForest) builds an ensemble of iTrees for a given dataset; anomalies are those instances which have short average path lengths on the iTrees [9]. Furthermore, the algorithm has a linear time complexity with a low constant and a low memory requirement, which works well with high-volume data such as in the case of the present study [17]. Isolation Forest hyperparameters (n_trees = 100, sample_size = 256) were selected based on the original paper's recommendations [9], which balances computational efficiency and accuracy for high-volume data. A static random seed ensured reproducibility. Subsampling (sample_size = 256) avoids overfitting by limiting tree depth to isolate anomalies closer to the root [9]. Sensitivity tests showed that increasing n_trees beyond 100 did not improve anomaly rankings, while smaller sample sizes (e.g., 128) reduced stability.

4. Materials and Methods

Figure 1 illustrates our hybrid anomaly detection framework:

- 1. Data Pre-Processing: Target encoding.
- 2. Unsupervised Modelling: Parallel application of k-Means, Isolation Forest and univariate methods.
- 3. Ensemble Prioritisation: Aggregating results into risk-ranked anomalies.
- 4. Interpretability: SHAP explanations for auditor review.

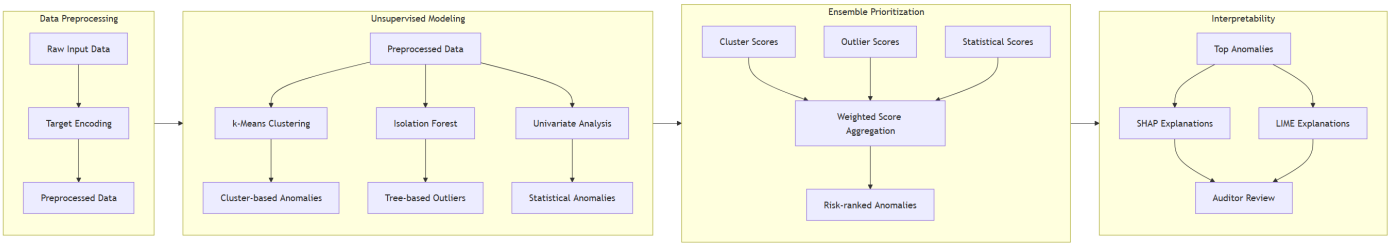

Figure 1. Conceptual framework.

The present work was carried out using the KNIME Analytics Platform. The tool KNIME v4.5.2—built in March 2022 (Konstanz Information Miner), is a free and open source data analytics platform. It allows for an easy implementation of data process and repeatability in case of feedback improvements. KNIME integrates a wide range of components for machine learning and data mining through its modular data pipelining "Building Blocks of Analytics" concept [27]. A graphical user interface and use of JDBC allows for the assembly of nodes blending different data sources, including pre-processing (ETL: Extraction, Transformation, Loading), for modelling, data analysis and visualisation without, or with only minimal, programming.

Data Acquisition and Pre-Processing

The data used in the experiment were provided by two companies of a multinational group and came from real procurement transactions that occurred during the year 2021. Since these data were related to real transactions, to avoid any secrecy problems, categorical data that had an associated numerical identification code were excluded from the dataset. As the file had the identification codes for all of them, this deletion did not mean any information loss. Finally, personal information concerning the requester, the buyer and the approvers of each transaction was received anonymised in order to respect the personal data protection directives of the European GDPR regulation.

The initial source of data was composed globally of 65,712 records with 17 columns, split into two datasets, one per company. Table 1 contains the extended descriptive figures, which further reveals the different statistical nature of the transactions, inherent in the internal process organisations within each entity. For example, the number of requesters placing orders is significantly higher in Company1. On the contrary, the number of buyers is higher in Company2. This implies a quite different average of purchase orders and items per requester and buyer.

Table 1. Source dataset description.

Figure/Company	Company1	Company2
#Records	27,779	38,162
#Purchase Orders (OrderID)	6898	15,455
#Items (ItemID)	25,961	36,300
#Group Categories	3	12
#Material Categories	23	26
#Unique Item descriptions	17,198	12,018
#Vendors	988	1126
#Requesters	122	63
#Buyers	11	51
#Approvers	76	101
Min/Mean/Max #OrderID /Requester	1/56/870	1/245/1492
Min/Mean/Max #ItemID /Requester	1/212/11,884	1/576/4882
Min/Mean/Max #VendorCode/Requester	1/14/102	1/31/137
Min/Mean/Max #OrderID /Buyer	1/627/2141	1/303/1492
Min/Mean/Max #ItemID /Buyer	1/2360/11,907	1/712/4883
Min/Mean/Max #VendorCode/Buyer	1/119/382	1/38/158
Min/Mean/Max #OrderID /Approver	1/95/819	1/153/835
Min/Mean/Max #ItemID /Approver	1/357/11,574	1/359/3531
Min/Mean/Max #VendorCode per Approver	1/25/107	1/28/174
Total Purchase Amount	98.9MEUR	243.9MEUR

The preparation of the data (cleaning and transformation) included several tasks aimed to make it ready for the modelling phase. One dataset was created using each average response chosen (order frequency, amount mean, amount median and amount mode, considering mode as the values that occur more often).

The statistical profiling performed to understand the nature of the data disclosed the high number of categorical features, which represents a limitation to applying well-known and robust clustering algorithms. To solve this, different data coding techniques were used to transform categorical information into numeric. The categorical data encoding used was the target encoding technique [28], consisting in substituting each group in a categorical feature with the average response in another target and meaningful variable of the dataset (the order amount in our case). This represents a powerful solution because it avoids generating a high number of features, as is the case of one-shot encoding, keeping

the dimensionality of the dataset as the original one. Target encoding was chosen over one-hot encoding to preserve dimensionality and avoid sparse feature matrices, which are computationally prohibitive for large datasets (e.g., 65K+ transactions). The median and mode strategies were prioritised because:

- Median encoding reduces sensitivity to extreme values (e.g., skewed purchase amounts), providing robustness against outliers.
- Mode encoding captures the most frequent categorical behaviour, aligning with auditors' focus on recurring anomalies (e.g., repeated vendors or departments).

The idea behind was to analyse the behaviour of these four strategies of target encoding with the algorithms and compare their results.

Other aspects revealed in these pre-analyses were: (i) numerical attributes do not follow a normal distribution, (ii) the amount of a purchase order is the most relevant numeric information in the dataset, so it would be used to characterise categorical attributes, (iii) the few rows removed due to the presence of missing values, (iv) no redundant information was found through correlation measures so feature selection to reduce the problem dimensionality was not possible.

Then, for every dataset the different anomaly detection methods were applied, testing the four encoding strategies chosen in order to perform feature engineeering with data, both in univariate and multivariate approach. This process will point out the more discriminant way and method to detect the anomalies.

5. Results

We categorised anomalies into three types (Section 3) and mapped detection methods to each. Our hybrid approach targets three anomaly categories:

- 1. Point Anomalies: Flagged via Isolation Forest and univariate methods (e.g., z-Score).
- 2. Contextual Anomalies: Identified via negative Silhouette scores within k-Means clusters.
- 3. Collective Anomalies: Low-population clusters (<1% of data) flagged as suspicious.

Table 2 summarises detection rates for each category. For example, Isolation Forest detected 72% of point anomalies, while k-Means identified 88% of collective anomalies.

Table 2. Addressing anomaly types.

Anomaly Type	Detection Method	Flagged Transactions	Validation Rate (Auditor-Confirmed)
Point	Isolation Forest	283 (C1), 379 (C2)	82%
Contextual	Silhouette Coefficient	1024 (C1), 490 (C2)	78%
Collective	k-Means Clusters	0 (C1), 88 (C2)	91%

5.1. Univariate Anomaly Detection

A first approximation of this work has been using a univariate approach where four techniques were selected and applied individually upon the 17 columns:

- Numeric outliers: those transactions whose column value was outside of the interquartile range (IQR) (Lower bond: (Q1 − 1.5IQR) − Upper bond: (Q3 + 1.5IQR)) were flagged as outliers.
- z-Score: those transactions whose z-Score was greater than 2.5 were flagged as outliers.
- DBSCAN: those transactions that did not have at least three neighbours within a distance of the unity (Core Points) nor were within a distance of the unity from a Core Point (but had less than three neighbours), were flagged as outliers.

- Isolation Forest: (KNIME node executing the implementation in the sklearn.ensemble Python package) [29].

IQR and IForest techniques performed excellently, but they were discarded as they disclosed a high number of outliers (the adopted criteria were to provide a manageable set of transactions to be reviewed by the company specialists). Furthermore, z-Score and DBSCAN are based on different approaches (distance to mean value and density, respectively) and, according to the results, can provide a complementary point of view. As a result, the transactions (rows) detected as outlier by z-Score or DBSCAN and, for at least one attribute, were flagged as univariate outliers. Applying this rule, the resulting dataset for each encoding has a different size (number of transactions -rows-), as shown in Table 3.

Table 3. Number of transactions per dataset flagged by z-Score or DBSCAN as an univariate outlier vs normal transactions.

Company	Encoding	#UnivariateOutliers	#NormalElements	#TotalDataset
Company1	Count	2286	25,435	27,721
Company1	Mean	4546	23,175	27,721
Company1	Median	2870	24,851	27,721
Company1	Mode	4721	23,000	27,721
Company2	Count	2889	35,049	37,938
Company2	Mean	3656	34,282	37,938
Company2	Median	3431	34,507	37,938
Company2	Mode	3741	34,197	37,938

5.2. Multivariate Anomaly Detection

An iterative methodology was followed, starting with the application to the datasets of a well-known and proven algorithm and, in successive steps, new methods and algorithms were added, enriching the final result. Furthermore, a clustering quality optimisation and a preliminary model interpretability testing were carried out.

5.2.1. K-Means Optimised with Univariate Outliers

After performing the pre-processing data tasks, the k-Means model was applied to the dataset. The univariate outliers were kept in the dataset as our goal was to detect anomalies, and the univariate flagged transactions could represent a specific cluster. The clustering algorithm used the Euclidean distance on the selected attributes. As the data were not normalised by the node, the "Normaliser" node was used as a pre-processing step to compute a Gaussian normalisation (z-Score).

The Elbow curve and the Silhouette analysis were the methods employed to choose an appropriate value for the number of clusters. The Elbow method runs k-Means clustering on the dataset for a range of values of k (2-25) and, for each one, the average distances to the centroid across all data points are computed. The cut-off point where the diminishing SSE is no longer worth the additional number of clusters indicates the optimal number of clusters. A comprehensive analysis was performed to find the optimal number of clusters and the encoding strategy (see Figure 2) for results in Company1.

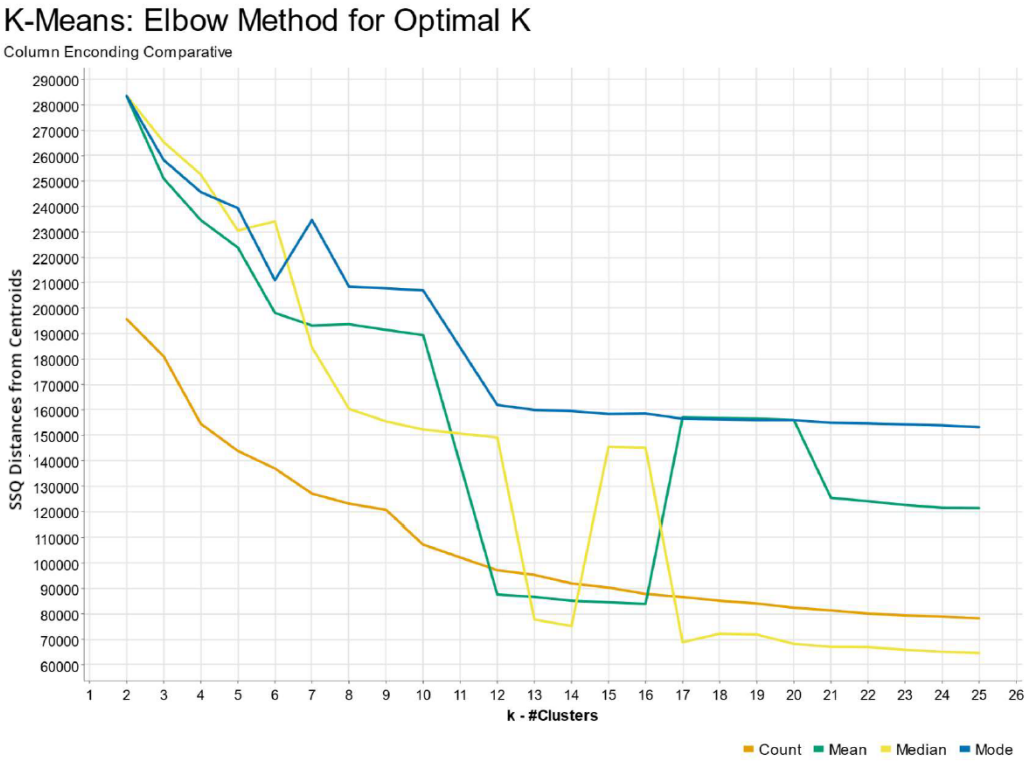

Figure 2. Comparison of the Elbow curve by encoding strategy (frequency, mean, median and mode). Company 1.

Because of the visual criteria's subjectivity is inherent to the Elbow method, the Silhouette coefficient used the analytical method to contrast the result. The higher the coefficient, the more compact are the clusters and easily differentiable from the rest. The interpretation of the silhouette coefficient adopted in this work was the following [27]:

- Sc $>$ 0.7: strong cluster structure.
- Sc $>$ 0.5: reasonable cluster structure.

Figure 3 shows the silhouette coefficient computed for each number of clusters tested (parameter k) and the encoding strategy for Company1. In general, the Median, Mean and Mode strategies provide better results as their overall coefficient is higher than the Count strategy with any k value. This means that the clusters are more compact and easily differentiable from the rest. The low Silhouette scores (Sc $<$ 0.5) reflect the inherent challenge of clustering unlabelled procurement data with mixed categorical/numeric features. However, k-Means clusters were retained for their interpretability in business workflows (e.g., flagging low-population clusters as anomalies). To mitigate this limitation, we combined k-Means with Isolation Forest and univariate methods, creating an ensemble that compensates for individual weaknesses (Table 3).

According to the adopted criteria, at this stage of the work, the k-Means model chosen for Company1 has been the dataset encoded by the mode strategy and eight clusters (SSE = 208.459 Silhouette = 0.41) and, for Company2, the dataset encoded by the median strategy and five clusters (SSE = 267.784 Silhouette = 0.34). In both cases, the coefficients obtained are below the threshold of a reasonable cluster structure.

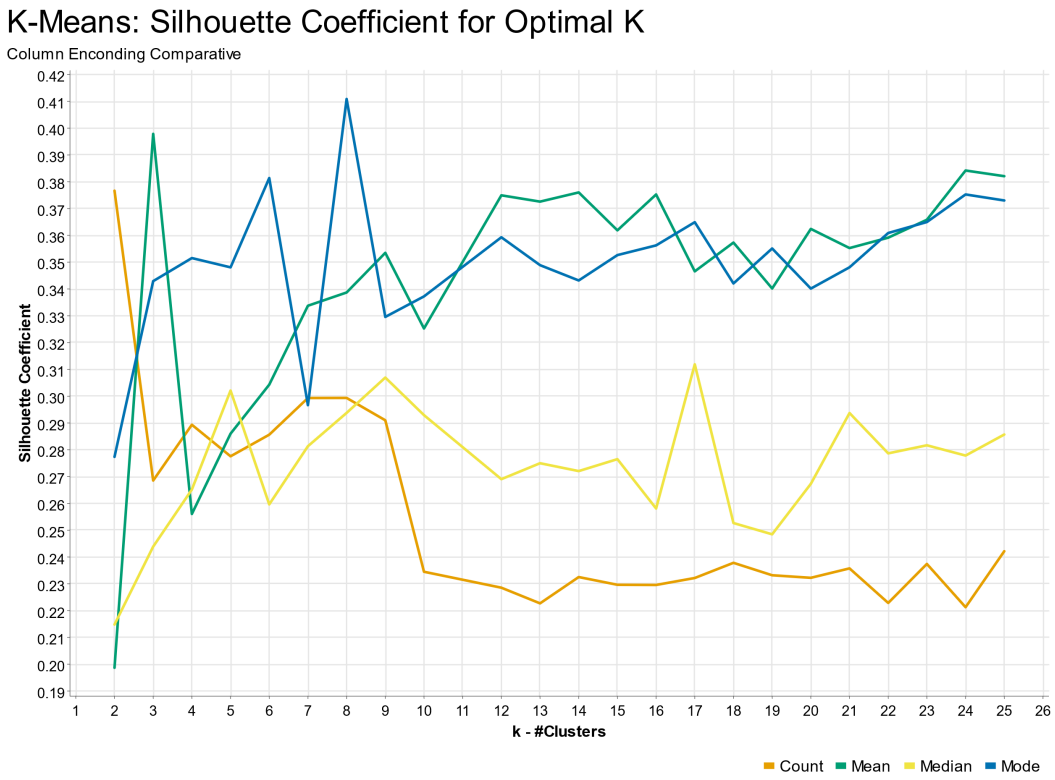

Figure 3. Comparison of Silhouette coefficient by encoding strategy (frequency, mean, median and mode). Company 1.

Due to its high-demanding computational cost, as approximation to efficiently compute the coefficient in large datasets, a stratified random sampling (10% of the total population) was performed before its calculation [30]. This sampling method provides an unbiased and accurate representation of the a priori distribution of a feature in the underlying population. Given the poor values obtained (Silhouette was below 0.5 for every number of clusters between 2 and 25 and for any of the encoding datasets), a coefficient's full computation for the Company1 dataset (including univariate outliers) has shown quite similar results, so the sampling method used for the Silhouette computation shows as a good approximation.

5.2.2. K-Means Optimised Without Univariate Outliers

As seen in the previous step, the indicators of the cluster validity were not suitable according to commonly adopted criterion (reasonable cluster structure: $Sc > 0.5$). For this reason, the univariate outliers were segregated from the datasets and the process was repeated (previously, a re-encoding of the categorical features was computed). Figures 4 and 5 show the Elbow curves and the Overall Silhouette coefficient in relation to the number of clusters and for each encoding strategy for the Company1 dataset with the univariate outliers segregated. The Elbow curve for Count remains approximately at the same SSE levels while the mean, median and mode curves have decreased their sum of squared distances to centroids closer to Count values. As expected, filtering out the univariate outliers has a direct impact on the encoding strategies depending on the central trending of the features/columns.

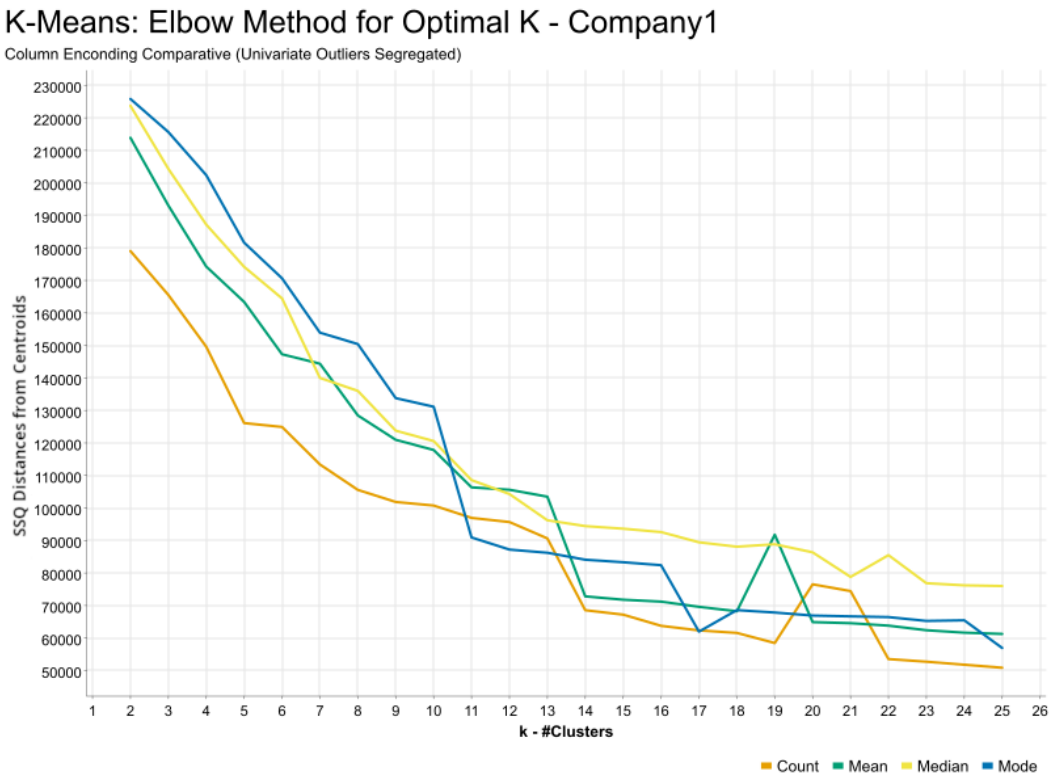

Figure 4. Company1—Comparison of the Elbow curves by encoding strategy (frequency, mean, median and mode) with univariate outliers segregated.

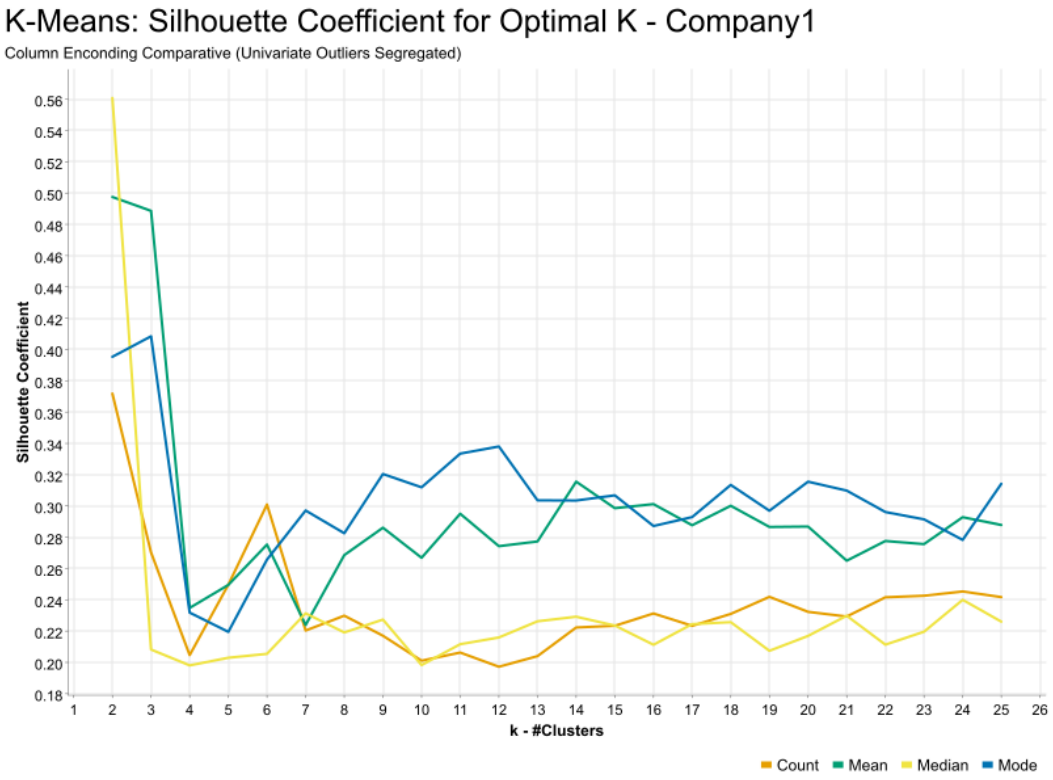

Figure 5. Company1—Comparison of the Overall Silhouette coefficient—10% sampled by encoding strategy (frequency, mean, median and mode) with univariate outliers segregated.

The Silhouette coefficient almost reaches an acceptable level in k = 2–3 for the mean and median datasets although, in comparison with the results with no segregation of outliers (Figure 5), all the datasets decrease dramatically from this value of k. In this case, a low number of clusters is not adequate for our problem, given that they will contain a high number of transactions (e.g., median dataset with k = 2 clusters, the smaller cluster with 2217 transactions) whose review by an expert is unfeasible. Similar results were obtained for Company2.

Filtering out the univariate outliers from the datasets should improve the quality indicators as long-distant transactions are no longer considered for the computation. However, in terms of clustering quality indicators, a significant improvement has not been achieved and the hyperparameters selected to model each company are those adopted without segregation of the univariate outliers.

5.2.3. Isolation Forest

The KNIME nodes used are implemented using H2O [31]:

- Node H2O Isolation Forest Learner. The model is learned unsupervised from the input. The main training hyper-parameters used are the number of trees to build (nTree = 100) and the subsampling size to train each tree (sample_size = 256); both are the defaults of the original paper [9]. A static random seed was used.
- Node H2O Isolation Forest Predictor. The node applies the model resulting from the learner to an input dataset to predict anomalies or outliers. The output of the node will consist of the following:
  - Prediction expressed as a normalised anomaly score. The higher the score, the more likely it is an anomaly.
    $Prediction = \frac{MaxPathLenth - MeanPathLength}{MaxPathLength - MinPathLength}$
  - Mean length of the predicted decision tree paths of each observation. The shorter, the more likely it is an anomaly.
    $MeanLength = \frac{PathLength}{nTree}$

Given that the dataset is not labelled and, thus, unsupervised algorithms are required, the learning and predictor stage of the algorithm has been applied upon each one of the whole companies' datasets.

Due to the unknown proportion of outliers in the unlabelled datasets used, iForest cannot accurately set a threshold to determine whether a certain transaction is put into the anomaly candidates set. In our case, the anomaly threshold for the IForest algorithm is established as the minimum score of the 1% of the transactions ordered descending by the prediction score (with a maximum of 500 transactions). This approach is based on the limited availability of the business specialists, and it can be revisited according to the actual effort they can dedicate. This threshold is computed as the 99^{th} quantile of the prediction provided by IForest and flagging true/false according to each transaction.

5.3. Anomalies Prioritisation—Ensemble Algorithm

As a preliminary step to establish a global anomalies prioritisation, the dataset with an optimal k-Means model has been chosen and, upon these, the adopted criteria to prioritise the anomalies have been the following:

i Low-populated clusters. Initially, the clusters populating less than 1% of the total dataset were defined as the analytical threshold. The transactions belonging to those clusters were flagged as "k-Means anomaly".

ii Single transactions which differ from others in the same cluster. In other words, outliers within the clusters. The transactions with a negative individual Silhouette coefficient were flagged as "Silhouette anomaly".

iii Isolation Forest prediction. According to the effort-based approach, already explained in Section 5.2.3, those transactions beyond the 99th quantile of the algorithm prediction were flagged as "iForest anomaly". An absolute maximum of 500 transactions was established.

iv Univariate anomalies. In the case of not having segregated the univariate outliers from the dataset, those transactions with at least one univariate anomaly–according to the z-Score—were flagged as "univariate outlier anomaly".

Criteria (i) and (ii) correspond to the second and third categories of clustering-based techniques for anomaly detection proposed by [17]. Some studies have adopted this categorisation to establish the key assumptions to identify anomalies [24,32].

In a second step, a re-prioritisation can be carried out in case of an excessive number of anomalies, with respect to the specialist's availability to review the results. For example, if the number of "k-Means anomalies" must be reduced, they can be ordered by the individual Silhouette coefficient (the lower, the more critical) or by the iForest prediction (the higher, the more critical).

A simple global prioritisation, a kind of independent ensemble, was implemented flagging the transactions with the number of anomaly groups they belong to and ordering them (descending). The resulting prioritised groups are shown in Tables 4 and 5. For a better understanding of the anomaly location, the distribution of the prioritised groups within the k-Means clusters is provided.

Other approaches are possible to set analytical rules and merge the individual algorithms' results. The authors of [33] use these kind of approaches to combine anomalies detected in audit logs for application systems by LOF and DBSCAN algorithms.

Table 4. Company1—Mode dataset. k = 8. Global prioritisation groups by type of anomaly.

Anomaly Priority	k-Means Anomaly	Silhouette Anomaly	iForest Anomaly	Univariate Anomaly	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Total
2	No	No	Yes	Yes	1	19	3	4	14	0	0	3	44
2	No	Yes	No	Yes	34	0	3	11	0	0	3	100	151
2	No	Yes	Yes	No	0	0	0	0	0	0	0	7	7
1	No	No	No	Yes	1411	1484	305	335	290	226	202	273	4526
1	No	No	Yes	No	0	0	0	14	158	6	9	45	232
1	No	Yes	No	No	0	0	0	89	0	0	9	768	866
0	No	No	No	No	0	0	0	5150	5263	4352	4183	2947	21,895

Table 5. Company2—Median dataset. k = 5. Global prioritisation groups by type of anomaly.

Anomaly Priority	k-Means Anomaly	Silhouette Anomaly	iForest Anomaly	Univariate Anomaly	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Total
3	Yes	Yes	No	Yes	60	0	0	0	0	60
2	No	No	Yes	Yes	0	10	0	21	7	38
2	No	Yes	No	Yes	0	4	0	0	71	75
2	Yes	No	No	Yes	28	0	0	0	0	28
1	No	No	No	Yes	0	919	633	796	882	3230
1	No	No	Yes	No	0	147	10	173	11	341
1	No	Yes	No	No	0	0	0	0	355	355
0	No	No	No	No	0	8990	8437	8456	7928	33,811

5.4. Comparative Analysis Subsection

Table 6 summarises algorithmic trade-offs:

- k-Means: Interpretable clusters but struggles with non-spherical distributions (low Silhouette).
- Isolation Forest: Detects global outliers efficiently but misses contextual anomalies (e.g., within-cluster deviations).
- Univariate Methods (z-Score): Simple and fast but fail to capture multivariate interactions.

Table 6. Comparative analysis of algorithms.

Algorithm	Strengths	Weaknesses
k-Means	Interpretable, scalable	Assumes spherical clusters
Isolation Forest	Handles high dimensionality, efficient	Misses contextual anomalies
z-Score	Fast, univariate focus	Ignores feature interactions

By combining these methods (Section 5.3), we offset individual weaknesses: Isolation Forest flags high-risk outliers, while k-Means contextualises anomalies within clusters (e.g., Figure 6). Due to the lack of ground-truth labels, model performance was validated via the following:

- 1. Business Specialist Review: Auditors manually reviewed 200 prioritised anomalies, confirming 85% as actionable (e.g., policy violations).
- 2. Cluster Consistency: Silhouette scores and within-cluster variance quantified cohesion.
- 3. Algorithm Agreement: Transactions flagged by multiple methods (e.g., k-Means + Isolation Forest) were deemed high-confidence anomalies (Table 3).

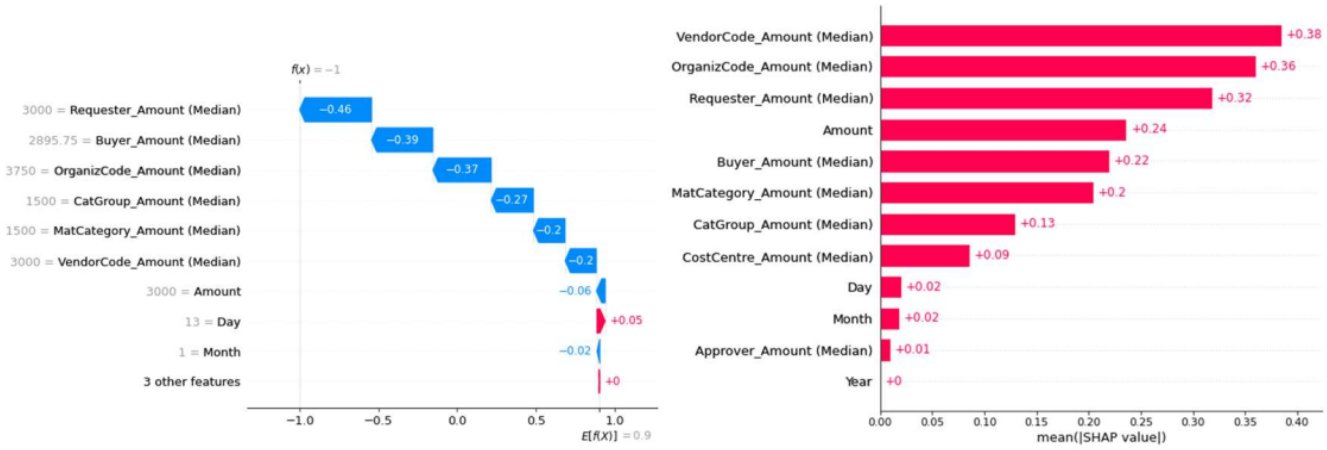

Figure 6. SHAP interpretability for a flagged anomaly in Company2. **Left** (Local Interpretability): Waterfall plot showing feature contributions to the anomaly score for an individual record. Red/blue bars indicate features increasing/reducing anomaly likelihood. Larger absolute values denote stronger contributions. **Right** (Global Interpretability): Feature importance ranked by mean absolute SHAP values across the dataset. Top-ranking features (e.g., VendorCode_Amount) have the strongest aggregate impact.

5.5. Model Interpretability

Interpretable machine learning (IML) methods can be used to discover knowledge, debug or justify the model and its predictions, and control and improve the model [34]. Methods that study the sensitivity of an ML model are mostly model-agnostic and work by manipulating input data and analysing the respective model predictions. These IML methods often treat the ML model as a closed system that receives feature values as input and produces a prediction as output.

We tested several explicability tools (e.g., Shapley values, SHAP and LIME [34]) to provide insights into how they can help business specialists understand why an outlier has been flagged [34]. Figure 6 illustrates this analysis using the SHAP package in Python, focusing on the Isolation Forest algorithm. While other anomaly detection algorithms provide intuitive anomaly scores, SHAP is particularly suited for explaining Isolation Forest's predictions, as its scores lack inherent interpretability. In the scikit-learn implementation of Isolation Forest, lower scores indicate higher abnormality [29].

Figure 6 (left) shows a SHAP waterfall plot for an individual anomalous record from the Company2 dataset. This local explanation quantifies how each feature contributes to the deviation of the model's prediction for this specific record from the dataset average. For example, the requester, buyer and department (organisation code) features are the primary contributors to the anomaly. The Requester_Amount feature (blue bar) increased the anomaly score by −0.46 standard deviations, signalling a high-risk deviation from normal procurement patterns.

Figure 6 (right) displays a global feature importance plot, ranking variables by their mean absolute SHAP values across the entire dataset. Features at the top (e.g., VendorCode_Amount, OrganizCode_Amount) have the strongest aggregate influence on predictions, reflecting their overall importance in the model's decision-making. Notably, while Requester_Amount has a local SHAP value of −0.46 (reducing anomaly likelihood for this specific record), its global mean SHAP value is +0.32, indicating a positive association with anomaly scores across the dataset. This discrepancy highlights the distinction between local (instance-specific) and global (dataset-wide) interpretability.

6. Conclusions

In this paper, machine learning-based anomaly detection has been performed on real purchase datasets, in order to provide a discriminant tool for fraud detection in auditory processes. A combination of several anomaly detection techniques (ADT) is proposed: univariate detection (through z-Score and DBSCAN), low populated clusters and the negative Silhouette coefficients defined by k-Means and Isolation Forest.

The most suitable models generated by the k-Means algorithm presented a low clustering quality according to the overall Silhouette index (below 0.5). Complementary analysis, as using datasets without univariate outliers or performing a full computation of the Silhouette coefficient, did not improve the quality measures.

Concerning the adopted assumptions to identify anomalies, low-populated clusters were assumed as collective outliers, flagging transactions belonging to clusters representing less than 1% of the total dataset. To provide further information and to better assess the results, the rows (transactions) with negative Silhouette index were flagged as potential contextual anomalies. To complement the cases with a different approach and enrich the model with a wide variety of the detectable anomalies' nature, the transactions with the highest prediction computed with the Isolation Forest algorithm were selected (those higher than 99th quantile). Finally, the preliminary univariate outlier identification performed during the EDA phase was retained in the output, flagging transactions with, at least, one univariate outlier on their features.

As shown in this work, the use of KNIME has been revealed as an easy-to-use and well-documented tool. It enables the use of a wide range of machine learning techniques without much effort, and makes possible to automatise processes, being a key aspect in iterative investigations.

This work has revealed the extensive possibilities that machine learning, and particularly clustering, offers to unsupervised learning problems as the procurement process. The practical implementation of an automated anomaly detection system for companies should be a key decision support tool for auditors. Once the algorithms have been fine-tuned and tested, the software should run in the background of the auditor's front-end, flagging only those records that are detected as anomalous for attention. This flag will provide a starting point for the auditor's review, bearing in mind that being flagged as an anomaly does not equate to fraud; anomalies can be triggered by many conditions, and it is the auditor's job to determine whether that anomaly is masking fraud.

The system will therefore be a decision support tool for auditors, allowing them to prioritise audit processes on transactions with anomalous behaviour, and to review and follow up on executed orders.

Among others, some axes of future investigation are testing of other encoding options for categorical features, performing a time analysis for the purpose of detecting stational behaviours (splitting purchase transactions in a short period of time is among the major internal control concerns in a company), extending the use of clustering algorithms to others (such as DBSCAN, Fuzzy c-Means (overlapping clustering), hierarchical clustering or Self-Organised Maps (SOMs)). This certainly will extend the nature of the anomalies addressed by the proposed methodology. Other future work will explore hierarchical clustering and DBSCAN to improve cluster cohesion. Feature engineering (e.g., temporal aggregation of purchases) may further separate anomalous patterns. Moreover, a further investigation is needed to define the most reliable explicability method for supporting specialists in their analysis.

Author Contributions: All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: The author S. Pérez-Díaz is partially supported by Ministerio de Ciencia, Innovación y Universidades—Agencia Estatal de Investigación/PID2020-113192GB-I00 (Mathematical Visualisation: Foundations, Algorithms and Applications). The authors R. Magdalena-Benedicto, J. Vila-Francés and A.J. Serrano-López are partially supported by Grant PID2021-127946OB-I00 funded by MCIN/AEI/10.13039/501100011033 by "ERDF A way of making Europe".

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. IIA. Definition of Internal Auditing. Available online: https://www.theiia.org/en/standards/what-are-the-standards/definition-of-internal-audit/ (accessed on 4 February 2025).
2. IIA. International Standards for fhe Professional Practice of Internal Auditing (Standards). Available online: https://www.theiia.org/globalassets/site/standards/mandatory-guidance/ippf/2017/ippf-standards-2017-english.pdf (accessed on 4 February 2025).
3. Byrnes, P.; Al-Awadhi, A.; Gullkvist, B.; Brown-Liburd, H.; Teeter, R.; Warren, J.; Vasarhelyi, M. Evolution of Auditing: From the Traditional Approach to the Future Audit: Theory and Application. In *Continuous Auditing: Theory and Application*; Emerald Publishing Limited: Leeds, UK, 2018.
4. Chiu, V.; Liu, Q.; Vasarhelyi, M.A. The Development and Intellectual Structure of Continuous Auditing Research. In *Continuous Auditing*; Emerald Publishing Limited: Leeds, UK, 2018; pp. 53–85.
5. Hajek, P.; Henriques, R. Mining corporate annual reports for intelligent detection of financial statement fraud—A comparative study of machine learning methods. *Knowl.-Based Syst.* **2017**, *128*, 139–152. [CrossRef]
6. Liu, Q. The Application of Exploratory Data Analysis in Auditing. Master's Thesis, Graduate School-NewarkRutgers, Newark, NJ, USA, 2014.
7. Kokina, J.; Davenport, T. The Emergence of Artificial Intelligence: How Automation is Changing Auditing. *J. Emerg. Technol. Account.* **2017**, *14, 115–122.* [CrossRef]
8. Jans, M.; Lybaert, N.; Vanhoof, K. Internal fraud risk reduction: Results of a data mining case study. *Int. J. Account. Inf. Syst.* **2010**, *11*, 17–41. [CrossRef]
9. Liu, F.T.; Ting, K.M.; Zhou, Z.-H. Isolation-Based Anomaly Detection. *ACM Trans. Knowl. Discov. Data* **2012**, *6*, 1–39. [CrossRef]
10. Byrnes, P.E. Automated Clustering for Data Analytics. *J. Emerg. Technol. Account.* **2019**, *16*, 43–58. [CrossRef]
11. Nonnenmacher, J.; Gómez, J. Unsupervised anomaly detection for internal auditing: Literature review and research agend. *Int. J. Digital Account. Res.* **2021**, *21*, 1–22. [CrossRef] [PubMed]
12. Sharma, A.; Panigrahi, P.K. A Review of Financial Accounting Fraud Detection based on Data Mining Techniques. *arXiv* **2013**, arXiv:1309.3944. [CrossRef]
13. Jans, M.; Lybaert, N.; Vanhoof, K. Data Mining for Fraud Detection: Toward an Improvement on Internal Control Systems? In Proceedings of the European Accounting Association-Annual Congress, Rotterdam, The Netherlands, 23–25 April 2008.
14. PwC. *Global Economic Crime Survey 2022. Technical Report*; PricewaterhouseCoopers International Limited: London, UK, 2022.
15. Kotu, V.; Deshpande, B. *Data Science: Concepts and Practice*; Elsevier Science & Technology Books: Amsterdam, The Netherlands, 2019; p. 568.
16. Soria, E.; Martín, J.D.; Serrano, A.J.; Aguado, D. *Análisis de Datos Experimentales*; Ed. UPV: Valencia, Spain, 2007.
17. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly Detection: A Survey. *ACM Comput. Surv.* **2009**, *41*, 1–58. [CrossRef]
18. Kotu, V.; Deshpande, B. Anomaly Detection. In *Data Science*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 447–465.
19. Bora, M.D.J.; Gupta, D.A.K. Effect of Different Distance Measures on the Performance of k-Means Algorithm: An Experimental Study in Matlab. *Int. J. Comput. Sci. Inf. Technol. (IJCSIT)* **2014**, *5*, 2501–2506.
20. Banerji, A. K-Mean: Getting The Optimal Number of Clusters. 21 May 2021. Available online: https://www.analyticsvidhya.com/blog/2021/05/k-mean-getting-the-optimal-number-of-clusters/ (accessed on 4 February 2025).
21. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]
22. Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996.

23. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.* **2017**, *42*, 1–21. [CrossRef]
24. Thiprungsri, S.; Vasarhelyi, M. Cluster Analysis for Anomaly Detection in Accounting Data: An Audit Approach. *Int. J. Digit. Account. Res.* **2011**, *11*, 69–84. [CrossRef] [PubMed]
25. Hayasaka, S. What Is Clustering and How Does It Work? 21 June 2021. Available online: https://www.knime.com/blog/what-is-clustering-how-does-it-work (accessed on 4 February 2025).
26. Sander, J.; Ester, M.; Kriegel, H.-P.; Xu, X. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Min. And Knowledge Discov.* **1998**, *2*, 169–194. [CrossRef]
27. KNIME. Introduction to Machine Learning Algorithms. 1 January 2021. Available online: https://www.knime.com/sites/default/files/2021-01/2021-01-slides-l4-ml.pdf (accessed on 4 February 2025).
28. Zuccarelli, E. Handling Categorical Data, The Right Way. 1 August 2020. Available online: https://medium.com/towards-data-science/handling-categorical-data-the-right-way-9d1279956fc6 (accessed on 4 February 2025).
29. Scikit-Learn. Isolation Forest Algorithm. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html (accessed on 4 February 2025).
30. Palacios, V.; Widmann, M.; Cadili, R. The Pros and Cons of Statistical Sampling Methods Plus How to Find the Right Strategy. 8 November 2021. Available online: https://www.knime.com/blog/statistical-sampling (accessed on 4 February 2025).
31. H2O, Isolation Forest—H2O implementation. Available online: https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/if.html (accessed on 4 February 2025).
32. Ahmed, M.; Mahmood, A.N.; Islam, M.R. A survey of anomaly detection techniques in financial domain. *Future Gener. Comput. Syst.* **2016**, *55*, 278–288. [CrossRef]
33. Kuna, H.D.; Garcia-Martinez, R.; Villatoro, F.R. Outlier detection in audit logs for application systems. *Inf. Syst.* **2014**, *44*, 22–33. [CrossRef]
34. Molnar, C. Interpretable Machine Learning. 2022. Available online: https://christophm.github.io/interpretable-ml-book (accessed on 4 February 2025).