Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 181 (2021) 882–889

Procedia Computer Science

www.elsevier.com/locate/procedia

CENTERIS - International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2020

Unsupervised Anomaly Detection of Retail Stores Using Predictive Analysis Library on SAP HANA XS Advanced

João Pedro Oliveira[a], Rui Dinis Sousa[b,]*

[a]*University of Minho, Alameda da Universidade, 4804-533 Guimarães, Portugal*
[b]*University of Minho, ALGORITMI Center, Alameda da Universidade, 4804-533 Guimarães, Portugal*

Abstract

The retail industry is quite exposed to fraudulent situations. Daily, thousands of transactions are processed, which may include some frauds difficult to detect, mainly when the perpetrators are the own employees at the retail stores. Large retailers with several stores across different locations may have considerable difficulty in detecting frauds involving their cashiers since they have to take into account different contexts of operation. To reduce fraud losses, retailers get an overview of the transactions in each store to filter the ones that look suspicious deviating from what would be normal. Data mining algorithms can be useful to detect anomalies, differentiating the normal from the abnormal. This study adopted the k-Means clustering algorithm for anomaly detection on a sample of 90 stores in a large food retail chain, revealing the existence of some outliers in the data. The anomaly detection process was fully implemented in SAP HANA XS Advanced using the Predictive Analysis Library (PAL). In the end, it was possible to identify the stores with abnormal behavior and conclude for the usefulness and ease of use of such a library, despite some lack of documentation to use it.

Keywords: unsupervised anomaly detection; k-Means clustering; retail; PAL; SAP HANA XSA

* Corresponding author.
E-mail address: rds@dsi.uminho.pt

10.1016/j.procs.2021.01.243

1. Introduction

The retail sector is a business area heavily affected by fraud, as thousands of transactions are processed daily and within these transactions, there may be fraud that goes unnoticed. Furthermore, fraud has an extreme impact on the profits of retailers. However, it is not very easy to detect fraud in real time, especially fraud caused, directly or indirectly, by employees, since retail operators have a lot of control over their operations. Considering this, large retailers tend to establish suspicious scenarios, i.e., situations that may contain fraudulent transactions. In these scenarios, transactions can be found regarding discounts, cancellations (voids), returns or refunds of empties.

Large retailers hold multiple stores across several locations, which makes it even more difficult to detect fraud. Therefore, it is necessary to have a broad view of the business, examining the stores where more situations are present that, eventually, can be fraudulent. This is where data mining algorithms can help by trying to detect anomalies in the data of all the stores of a given retailer, to flag the stores that are most prone to fraud. Once this knowledge is obtained, different preventive measures can be applied, depending on the severity of each situation.

The data for this research comes from the points of sale (POS) of a large food retail chain, housed in SAP Customer Activity Repository (CAR), a central repository designed to provide optimum results with SAP HANA [1]. The data could be accessed through the SAP HANA extended application services, advanced model (XSA) [2] (an integral part of the SAP HANA), providing the conditions to use and explore emergent SAP technologies such as the ones offered in the Predictive Analysis Library (PAL). PAL enables, easily and without recourse to data extractions or connections to the database, via external applications, the execution of some data mining algorithms directly over the data. Thus, for this specific problem, it was possible to easily retrieve the data and apply an anomaly detection algorithm, via SQL, identifying, in the end, all the stores with abnormal behaviour and, therefore, considered outliers.

This study follows a methodology that was integrally implemented in SAP HANA XS Advanced, accessed from SAP Web IDE, using PAL to filter out anomalous stores. The algorithm used for anomaly detection in PAL is k-Means which, in a first instance, is used to group the original data into *k* clusters and, subsequently, to identify some points that are far from all centroids as anomalies. In short, the function provided by PAL to detect anomalies is applied to the data and the results are immediately generated in an acceptable format, provided that all the requirements of the algorithm in use are met. Details on important steps to get to the results are presented in the methodology section.

2. Related Concepts

2.1. Unsupervised Anomaly Detection

In data sets, there are often normal data points and data points that stand out as being different from all other data points. Those data points that differ significantly from other data points are called outliers or anomalies [3,4]. Thus, it is extremely important to identify them and understand why these data points may be associated with fraudulent situations. Anomaly detection is the process responsible for identifying unusual data points, discovering previously unknown patterns, and providing important information to the business [5,6].

There are different methods for detecting anomalies: supervised, semi-supervised, and unsupervised methods. This research work focuses on unsupervised methods. When data sets have a label (e.g., normal and outlier) supervised methods are applied. When data sets do not have a label, then, in that case, unsupervised methods must be applied [3]. Moreover, contrary to unsupervised methods, supervised methods have some limitations in terms of efficiency and computational performance [6], being also especially important, since in most cases, a label indicating to which class each data object belongs is unknown. Unsupervised anomaly detection approach establishes the pattern of normal behavior with unlabelled training data composed of normal and abnormal samples, considering two assumptions: the number of normal samples is far greater than the number of abnormal samples, and the number of abnormal samples is qualitatively different from normal samples [7,8].

Unsupervised methods can be divided into three categories: (1) distance-based methods; (2) density-based methods; (3) linear methods [3,4]. Methods based on distance and density calculate the outlier scores, using the distance and density between objects, respectively. On the other hand, linear methods convert high-dimensional data into low-dimensional data and calculate the outlier scores from the residuals of the samples [4].

In the case of this research work, a distance-based method is used (k-Means): clustering can be applied to find abnormal behaviors in the data important to analysts [5]. The classic k-Means clustering method can be easily used by placing similar data in the same cluster and dissimilar data in different clusters and then labeling the data from each cluster as normal or anomalous, according to some measurements [5]. Clustering groups a set of data objects into different subsets (clusters), so that the data objects present in the same cluster have a high similarity to each other, but are quite dissimilar to other data objects belonging to other clusters [9–11]. Particularly, the k-Means algorithm is very useful to analyze data coming from the POS, since its characteristics allow it [12], helping in the decision making process [10]. Furthermore, since k-Means is sensitive to outliers, it is, even more, a reason why it should be used as a good method for detecting anomalies [13].

2.2. Predictive Analysis Library (PAL)

SAP HANA provides a set of techniques capable of moving application logic to the database through the Application Function Library (AFL) where PAL is included. The PAL defines functions that can be called using SQLScript procedures to execute analytical algorithms. SQLScript [14] is an extension of SQL in SAP HANA, which allows developers to define complex application logic through database procedures [15]. PAL includes predictive analysis algorithms in nine data mining categories: clustering, classification, regression, association, time series, pre-processing, statistics, social network analysis, and miscellaneous [15].

Although PAL has several algorithms, this library has a function specifically for anomaly detection, which is already prepared to return outliers in a recognizable and acceptable format. Therefore, this function is the most adequate for this research work, since, directly, using PAL, this will be an effective way of, while solving a problem, explore emergent SAP technologies. Although there are other algorithms, PAL explicitly offers this function based on clustering for anomaly detection, so it was decided to take advantage of it, to check its usefulness and usability.

3. Methodology

As mentioned, the objective of this study is to research on effective ways of detecting fraud making use of emergent SAP technologies, in a large food retail chain. The data is housed in SAP HANA and the entire process was run there, so it was possible to increase performance by computing in the database itself rather than at an application level. The data refers to transactions made for one year. In total, the sample comprises data from 90 stores. In this context, the term "transaction" refers to each receipt and not to each receipt line. The structure of the data used can be seen in the table below.

Table 1. Data structure.

Column Name	Data Type	Description
RETAILSTOREID	String	Unique store identifier.
RETURNS_PROPORTION	Double	Proportion of returns in relation to suspicious transactions (one year).
VOID_PROPORTION	Double	Proportion of voids in relation to suspicious transactions (one year).
DISCOUNTS_PROPOTION	Double	Proportion of discounts in relation to suspicious transactions (one year).
EMPTIES_PROPORTION	Double	Proportion of refund of empties in relation to suspicious transactions (one year).

In this context, a suspicious transaction is a transaction that is a return, cancellation, discount, or empties refund. Each of the above variables, except the identifier for each store, is calculated by dividing the number of transactions relating to that scenario by the total number of suspicious transactions. Taking the variable PROPORTION_RETURNS in a store s as an example, its calculation formula can be described in this way:

$$RETURNS\_PROPORTION_s = \frac{annual\ returns_s}{annual\ suspicious\ transactions_s}, \quad (1)$$

where $annual\ suspicious\ transactions_s = annual\ returns_s + annual\ voids_s + annual\ discounts_s + annual\ empties_s$. Therefore, each value of each of these variables can take a value between 0 and 1. The formulas to calculate the remaining variables are similar, only changing the formula numerator.

As referred, this work was implemented using PAL. So, a set of tasks had to be done to achieve the expected results: firstly, the data had to be retrieved by executing a SQL statement, since the data does not match the model indicated in Table 1 in any table of the original database. After that, there are all the conditions to start the creation of the project on SAP HANA XS Advanced. In the development part of SAP HANA XS Advanced, a Multi-Target Application (MTA) was created in the Cloud Foundry environment. Then, an SAP HANA database module was created, indicating some configurations. Afterward, an SAP HANA Database (HDB) Core Data Services (CDS) artifact was created, to build the entity that will save the SQL statement result. The SAP HANA CDS artifact transforms an SAP HANA CDS resource into a database object defined in the CDS file (e.g., tables, types, and/or views). After the file with the table data model was created, the result of the SQL statement was inserted into that table. After that, another HDB CDS artifact was created to build the types, entities, and views that will send data to the algorithm and save the results received from that algorithm (in this case, the k-Means algorithm, specifically applied to detect anomalies). Then, a .hdbaflprocedure file was added to build the procedure that will invoke the native anomaly detection method of SAP HANA, passing it the information of which tables are input and output. In other words, it turns an AFL language procedure resource into a corresponding database procedure object. After that file was added, a comma-separated value (CSV) file containing the parameters to be sent to the anomaly detection algorithm was also added. Moreover, a .hdbtabledata file was created which maps the parameters inserted into the CSV file to the anomaly detection algorithm. Later, a .hdbprocedure file was added to the project, to transform a procedure resource into a procedure database object. Finally, the built procedure was called to see the results.

The anomaly detection algorithm, like all the others provided by PAL, has a data model for sending and producing data. Input and output tables were used to run the algorithm, which can be seen below, as well as the Parameter table that is probably one of the most important tables since it is the one that sends the parameters to the algorithm.

Table 2. Input and output tables for the anomaly detection algorithm.

Table Name	Purpose	Type
Input Data	Sends the data to the algorithm.	Input
Parameters	Sends the desired parameters to the algorithm.	Input
Outliers	Saves the outliers and its coordinates	Output
Statistics	For each outlier, saves its cluster and the sum of the distance from the point to all centers.	Output
Centers	For each cluster, saves the information about its centroids (average values of each feature).	Output

Table 3. Parameters for the anomaly detection algorithm.

Parameter name	Used value	Meaning
DISTANCE_LEVEL	2	Euclidean distance.
OUTLIER_DEFINE	2	Max sum distance from the point to all centers.
MAX_ITERATION	100	Maximum of 100 iterations.
NORMALIZATION	2	Min-Max Normalization.
THREAD_NUMBER	2	Two threads.
EXIT_THRESHOLD	0.000001	Threshold for exiting the iterations.

4. Results and Discussion

The anomaly detection algorithm generates three tables (Outliers, Statistics, and Centers) with different results, which can be analyzed to draw some interesting conclusions. Moreover, a view called OUTLIERS_CLUSTERED was built to intersect the data in the output Statistics and Outliers tables. Thus, it would be possible to see which cluster each of the outliers belongs to. Instead of looking at separate tables, combined analyses can then be performed, directly in the SAP HANA XS Advanced. After the execution of the algorithm, a value for $k = 6$ was obtained (six clusters were formed). Although this is not the main purpose of the anomaly detection algorithm, it does not provide the number of stores belonging to each of the clusters. The table below summarizes the values of the centroids in each cluster.

Table 4. Centroids values.

CLUSTER	RETURNS_PROPORTION	VOIDS_PROPORTION	DISCOUNTS_PROPORTION	EMPTIES_PROPORTION
0	0.226	0.326	0.800	0.222
1	0.480	0.814	0.127	0.768
2	0.733	0.243	0.619	0.352
3	0.734	1	0.691	0.230
4	0.040	0.044	0.859	0.165
5	0.243	0.456	0.346	0.636

Analyzing the table above, it is possible to detect that, on average, cluster 0 includes stores with a high discount application ratio, compared to other scenarios. Cluster 1 includes stores with a high void and empties refund ratios, having a somewhat low discount application ratio, compared to all other clusters. Cluster 2 includes stores with high values in returns and discounts. Cluster 3 includes stores with high levels of returns, voids, and discounts, having a very interesting value in voids (1), which may suggest that it includes stores with the maximum void ratio. Cluster 4 presents itself as a cluster that includes stores with higher discount levels and very low values when compared to the other variables. Finally, cluster 5 is presented as a cluster, generally homogeneous, with the particularity of presenting one of the highest values for the refund of empties.

In the original data, in general, for all stores, the values registered for discounts and refunds of empties were the highest. Thus, only by analyzing these two variables, directly in SAP Web IDE, it is possible to take the location of the centers from each cluster.

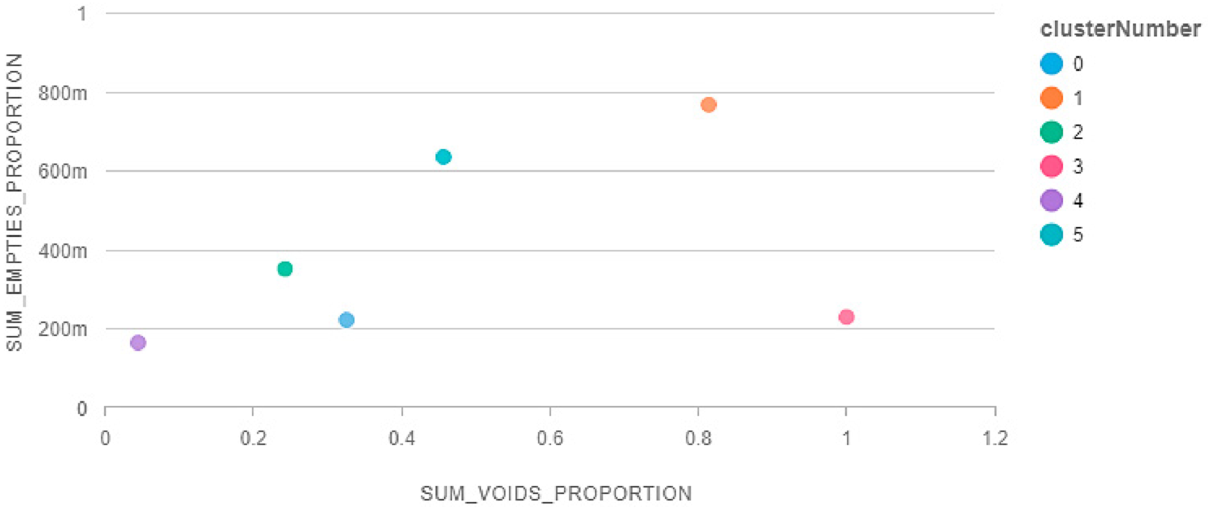

Figure 1. Centroids location

By analyzing the scatter plot in Figure 1, it can be concluded that the data in Table 4 conform to this plot. Each distinct color represents the centroids of each of the five existing clusters. This graph is a more appealing and simple way to visualize the values taken by each centroid. Similarly, compared to Table 4, cluster 1 (represented in orange) is the one with the highest values, while cluster 4 (represented in purple) is the one with the lowest values of the variables under analysis. With this more illustrative view, it is also possible to notice that cluster 2 (represented in green) and 0 (represented in light blue) are very close in space. This may suggest that some data objects present in those two clusters may be quite close to each other, being difficult to make a clear distinction to know to which cluster they belong, without a colorization in the plot. Moreover, no unit of measurement is shown in the data, either in Table 4 or in Figure 1, since they are proportions (nonetheless, when multiplied by 100, they are converted into a percentage).

In total, eight outliers were detected (about 10% of the original data). Only some clusters include outliers, while others can be considered as normal clusters. Only clusters 1, 2, 4, and 5 contain outliers, with cluster 1 having the highest number of outliers. The number of outliers per cluster can be seen in the table below.

Table 5. Number of outliers by cluster

Cluster	Number of outliers
1	4
2	1
4	2
5	1

After the algorithm is executed, it is not possible to view the normalized data, and only the original data is available for viewing. Although the data is not available after normalization, it is still possible to check the disposition of each outlier store in a scatter plot, analyzing the same two variables.

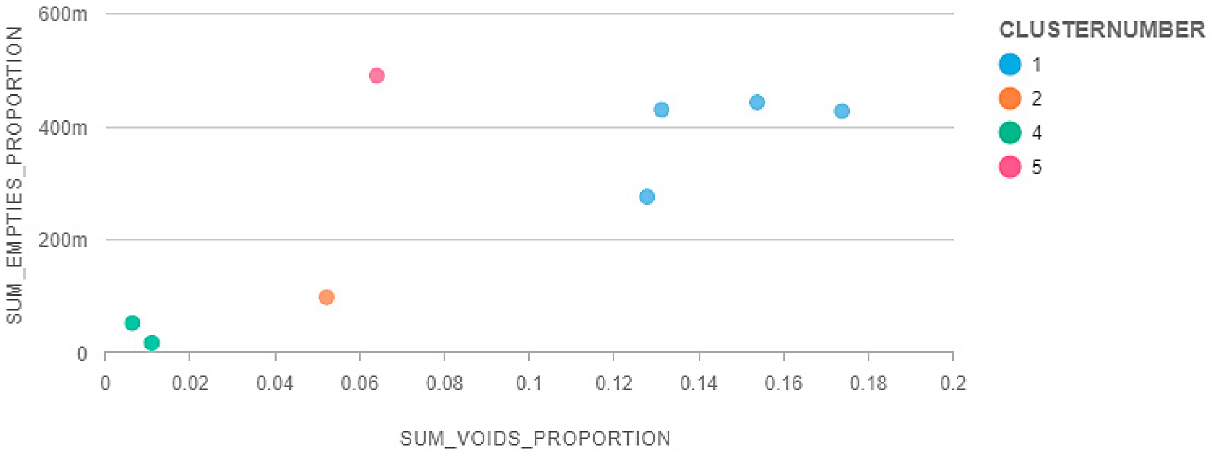

Figure 2. Outliers disposition

Comparing the last two figures, it is possible to see that some stores are more distant from their centroids than others. Nevertheless, in general, all identified outliers are very distant from their centroids. Specifically, the store belonging to cluster 2 appears to be closer to the centroid of cluster 0 than to its centroid. Clusters 0 and 3, not being represented in the scatter plot in Figure 2, are considered clusters that only contain normal data objects. Thus, from the business point of view, these clusters can be discarded for these variables under analysis. The remaining clusters, although with some differences between them and containing outliers, should receive attention from the business. Logically, cluster 1 being the one that contains the most outliers, it is the one that should be the main target of attention by the business. Similarly, it is the one that presents the stores with the highest values, in general, concerning the variables under analysis. Although cluster 5 contains only one anomalous store, it also shows quite high values, at least comparing to the variable of the ordinate axis. On the other hand, although the stores in clusters 2 and 4 have low values for the two variables, in the long term it may become harmful, however, it should not be the primary focus. The ideal strategy is to identify the stores marked and understand why there are high deviations within each cluster.

5. Conclusions

From the results of this study, it is possible to identify which stores have an abnormal behavior standing out from the others. In general, all stores have higher discount and empties refund ratios, since these two scenarios are part of retail daily life. Every day, a discount is applied to any product, or a customer wants to be refunded for the empties. However, the combination of return, cancellation, discount, and empties refunds, the four variables under analysis in this study, provides a broader picture, allowing for the signalization of fraudulent situations in stores with a negative impact on the business.

All work was particularly focused on using the anomaly detection algorithm provided by SAP to detect stores with abnormal behavior concerning suspicious transactions in the retail sector. Additionally, an analysis of the results in SAP HANA XS Advanced, returned by the algorithm, was also conducted.

The anomaly detection algorithm provided by PAL uses the k-Means method to detect outliers in the data, just an example of what PAL offers to the developer community. PAL from SAP HANA is equipped with several algorithms that allow the execution of application logic in the database, considerably increasing computational efficiency. PAL is a resourceful library quite effective for developers that can take advantage of the HANA environment without having to use third-party applications. However, the scarcity of documentation on PAL may be still an obstacle to its widespread use.

The anomaly detection algorithm has proven to be very efficient in terms of processing. The anomaly detection algorithm is also able to automatically calculate the best value for *k* without requiring the user to provide it. The possibility of analyzing the results, directly in the SAP Web IDE, using graphs, can be seen as a great advantage of this platform, making the understanding of the data easier without the need to perform any kind of extraction to another tool.

Acknowledgments

This work has been supported by *FCT – Fundação para a Ciência e Tecnologia* within the R&D Units Project Scope: UIDB/00319/2020.

References

[1] Kempen D van. SAP HANA 2.0: An Introduction. Rheinwerk Verlag GmbH; 2019.

[2] Alborghetti F, Kohlbrenner J, Pattanayak A, Schrank D, Sboarina P. SAP HANA XSA: Native Development for SAP HANA. Rheinwerk Verlag GmbH; 2018.

[3] Chen J, Sathe S, Aggarwal C, Turaga D. Outlier Detection with Autoencoder Ensembles. 17th SIAM Int. Conf. Data Mining, SDM 2017, 2017, p. 90–8. https://doi.org/10.1137/1.9781611974973.11.

[4] Ishii Y, Takanashi M. Low-cost Unsupervised Outlier Detection by Autoencoders with Robust Estimation Yoshinao. J Inf Process 2019;27:335–9. https://doi.org/10.2197/ipsjjip.27.335.

[5] Agrawal S, Agrawal J. Survey on Anomaly Detection using Data Mining Techniques. 19th Int. Conf. Knowl. Based Intell. Inf. Eng. Syst., vol. 60, Elsevier Masson SAS; 2015, p. 708–13. https://doi.org/10.1016/j.procs.2015.08.220.

[6] Fan C, Xiao F, Zhao Y, Wang J. Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. Appl Energy 2018;211:1123–35. https://doi.org/10.1016/j.apenergy.2017.12.005.

[7] Han L. Using A Dynamic K-means Algorithm to Detect Anomaly Activities. 2011 7th Int. Conf. Comput. Intell. Secur., 2011, p. 1049–52. https://doi.org/10.1109/CIS.2011.233.

[8] Han L. Research of K-MEANS Algorithm based on Information Entropy in Anomaly Detection. 2012 4th Int. Conf. Multimed. Secur. MINES 2012, 2012, p. 71–4. https://doi.org/10.1109/MINES.2012.169.

[9] Hossain ASMS. Customer Segmentation using Centroid Based and Density Based Clustering Algorithms. 3rd Int. Conf. Electr. Inf. Commun. Technol. EICT 2017, 2017, p. 1–6. https://doi.org/10.1109/EICT.2017.8275249.

[10] Kansal T, Bahuguna S, Singh V, Choudhury T. Customer Segmentation using K-means Clustering. Proc. Int. Conf. Comput. Tech. Electron. Mech. Syst. CTEMS 2018, IEEE; 2018, p. 135–9. https://doi.org/10.1109/CTEMS.2018.8769171.

[11] Syakur MA, Khotimah BK, Rochman EMS, Satoto BD. Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster. IOP Conf. Ser. Mater. Sci. Eng., vol. 336, 2018, p. 1–6. https://doi.org/10.1088/1757-899X/336/1/012017.

[12] Yoseph F, Heikkila M. Segmenting retail customers with an enhanced RFM and a hybrid regression/clustering method. Proc. - Int. Conf. Mach. Learn. Data Eng. iCMLDE 2018, 2019, p. 108–16. https://doi.org/10.1109/iCMLDE.2018.00029.

[13] Oliveira JP, Sousa RD. K-Means Clustering of Stores in Retail Industry Using Predictive Analysis Library on SAP HANA XS Advanced. Proc. 35th Int. Bus. Inf. Manag. Assoc. Conf., Seville: 2020, p. 15806–19.

[14] Brandeis J. SQLScript for SAP HANA. Rheinwerk Verlag GmbH; 2019.

[15] SAP. SAP HANA Predictive Analysis Library (PAL). 2015.