Abstract

Finding irregularities or detecting “not-normal” instances in a small amount of time is the main objective of an audit. This can be cumbersome if it involves a voluminous amount of data. It takes three (3) days on the average for auditors to manually produce an audit report. thus, anomalous cases take time to determine and full investigation of these cases were delayed. Also, auditors are having difficulty in prioritizing which item should come first thus it is important to have a formal framework that auditors can use to conduct the audit more efficiently. This capstone project proposed an alternative framework for intelligent prioritization of account. Statistical and machine learning techniques were used in identifying the priority level of audit of foreign exchange records. These techniques involve data decomposition using Seasonal and Trend decomposition using Loess (STL), Cubic Spline Smoothing, Automatic autoregressive integrated moving average (ARIMA), Generalized Extreme Studentized Deviate (GESD) test, unsupervised outlier detection model using Isolation Forest and Density-based spatial clustering of applications with noise (DBSCAN) and Clustering Large Applications based on RANdomized Search (CLARANS) with Recency, Frequency and Monetary (RFM) Analysis for its customer segmentation. The proposed methodology seeks to augment the existing audit process and reduce processing time in auditing monthly foreign exchange records and not necessarily replace the current audit process. Since each component investigated different aspect that influences a record, scoring of a record was done equally. While results of each component was produced independently, results were designed to be read as one output, each complementing the other. The proposed framework performed well, at 93%, with no Information Rate. Furthermore, adopting the framework supported the objective of an audit, which is to have a more holistic view of records compared to the traditional method.

Keywords Anomaly detection, record data, audit report

INTRODUCTION

One of the recommendations during the Bank for International Settlements (BIS) All Governors meeting held last 2015 (BIS, 2018) was for central banks to focus on projects that assess how data analysis can improve the effectiveness of supervision of banks. This was stressed further in the Irving Fisher Committee (IFC) Report (2020), where it recognized that though financial innovation and digitalization transforms the financial sector, it also opened data gaps in central bank statistics. One of its recommendations for central banks is to ensure that statistical methodologies used to measure financial activities adhere to sound professional and scientific standards. Also, recognizing these data gaps, the Chief Data Officer of the United States Federal Reserve Board, in his presentation during the seventh European Central Bank (ECB) statistics conference, highlighted its increased business risk and pointed out that increased data complexities require new approaches and solutions (Casey, 2014). In response to the challenge, the Philippines, represented by the Bangko Sentral ng Pilipinas (BSP) proactively seeks to innovate its capabilities to adapt in today's rapidly changing environment.

In National Risk Assessment last 2017, the Philippines identified its overall money laundering and terrorist financing threats as high (AMLC, 2017). Monitoring subject threats are being done by the Anti-Money Laundering Council (AMLC) with the support from different agencies such as Bangko Sentral ng Pilipinas (BSP). BSP, being the central bank of the Philippines, is tasked: (i) to provide policy directions in the areas of money, banking, and credit, (ii) to supervise the operations of banks and (iii) to exercise such regulatory and examination powers over banking operations of non-bank financial institutions, money service businesses, credit granting businesses, and payment system operators.

To fulfill its mandate as a regulating body, BSP uses several systems to monitor all banking records not only within the Philippines but also all foreign records coming from and to the Philippines. One of these systems being used by BSP is the International Transaction Reporting System (ITRS). ITRS is a system that collects data from banks at the level of individual records. The ITRS measures: (i) individual cash records that pass through domestic banks and enterprise accounts that pass through foreign banks, (ii) non-cash records, and (iii) stock positions. Statistics are compiled from forms submitted by domestic banks.

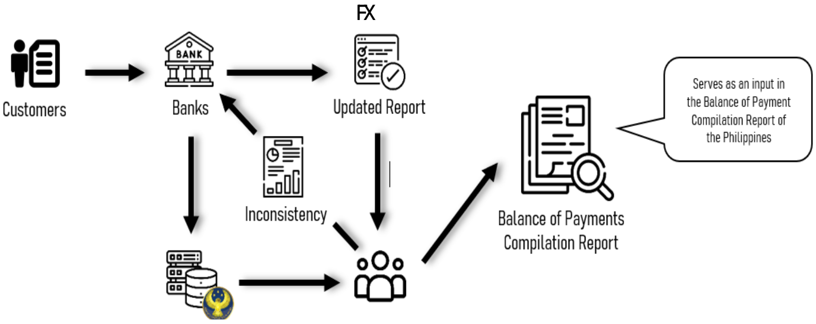


Figure 1. *BSP ITRS Subgroup Process Overview*

In Figure 1, banks submit a report to central banks through an online system. Countries using ITRS among others, are Indonesia, China, Malawi, South Africa, Ghana, Ukraine, Poland, and Hungary. In the Philippines, the report being submitted by banks is called the consolidated report on Foreign Exchange Assets and Liabilities (FX Form 1). It is a report consisting of twelve (12) Schedules, with a different number of items, which varies depending on the nature of the record. For one (1) schedule for example, there are approximately sixty (60) items, which vary in number of records, that need to be monitored. Not all banks are required to submit the said report but only the Authorized Agent Banks (AABs) that has a license to service foreign exchange records. Selling and Receiving of Foreign Exchange (FX) shall be duly reported by the FX selling/remitting/receiving AABs under the appropriate schedules of FX Form 1 based on the instructions of, and declared purpose by, the FX purchaser. All rules and regulations of reporting bank are indicated in the Manual of Foreign Exchange Records (December 2020- an enhanced and complete version of BSP Circular No. 1389, as amended, as it incorporates all amendments made since 1993 and consolidates all regulations on foreign exchange and related records), available in the BSP website.

The ITRS subgroup team consisting of eight (8) Auditors under the Department of Economics Statistics (DES) of the BSP is tasked to monitor the compliance of banks in submitting the FX Form 1 report and all its corresponding schedules on a weekly basis. Their primary goal is to check the correctness of the said report.

The data collected by ITRS is significant in the compilation of Balance of Payments Position (BOP) of a country in making comprehensive analysis to support policy formulation and implementation. BOP is a statistical overview that systematically summarizes the economic records of an economy with other countries of the world during a certain period. With the available data in ITRS, management wants to gain insights from it in a timely manner to be able to make informed policy decisions. Manually checking of huge amount of records causes delays in the production of ITRS report. It wastes valuable time that could have been used for other projects. In effect, the department's development is inhibited.

Currently, the auditors use a common rule-based method for monitoring each item code. Auditors check if the record exceeds a threshold value, which is currently the average amount of all records for the subject item. This works well, but the presence of extreme values can affect the calculation. Furthermore, this approach requires auditors to sift through hundreds of thousands of records every month. The average turnaround time to initialize an audit per auditor and per account is three(3) days. Thus, it takes time to determine anomalous cases. Also, the ratio of auditors to reports are not equally distributed due to variation in the number of records per account. Not all schedules defined in the FX Form 1 are being audited, which should be the ideal set up. Given the voluminous amount of records and resource constraints, auditors are having difficulty in prioritizing which item codes (bank code, record code, dealing code etc.) should come first. There is a need for a formal framework that auditors can use to conduct the audit more efficiently.

The rule-based monitoring method is an important part of any recording system. The method can be improved with the help of machine learning. Through proper implementation of machine learning, audit processing can be done in a small amount of time. As a matter of fact, there are numerous studies that were conducted on the application of machine learning that improved the efficiency of audit processing of some practical examples like banking records, structural defects in goods, medical diagnostics, error detection in texts or the cleaning of data and many others (Chakraborty & Joseph, 2017).

Finding irregularities or detecting "not-normal" instances or sometimes called anomaly detection or outlier detection is the main process of audit processing (Liu, 2019). In this project, a potential anomalous record is defined as a record that deviates from the "normal" behavior of all the records of a bank; a data point that is inconsistent with either the item or customer historical behavior. These records could be a possible indication of errors in the report. This project proposes a framework that will serve as a guide in identifying which audit items are potentially anomalous and need to be prioritized. The proposed methodology seeks to augment the existing audit process and reduce processing time in auditing monthly foreign exchange records and not necessarily replace the current audit process.

Objective of the project

The main goal of this project is to propose an alternative methodology in prioritizing accounts of different records by developing a framework using a combination of statistical and machine learning techniques that will identify potential anomalous records in the monthly report submitted by different companies. The said framework will aid auditors in making the audit process more effective and efficient. Not all items had an available labeled dataset, thus this study will utilize unsupervised machine learning techniques in model development, but the entire framework will be tested using the available labeled data of selected items.

Specifically, this project employed three methodologies that will flag the presence of anomalous records and the outcome was consolidated to categorize a record for audit whether low priority, medium priority, or high priority:

Component A: Identify anomalies per item through time series decomposition

Component B: Identify anomalous records using an Unsupervised Outlier Detection Approach on a Company Type - Level

Component C: Identify anomalous records based on Customer Behavioral segments using clustering and classification techniques

Significance of the Project

The implementation of an intelligent account-prioritization framework may significantly reduce the processing and production time of audited reports. On the average, it usually takes thirty-five (35) hours of manual work per auditor to produce a consolidated ABC audit report. Presently, an investigation is initiated after manual audit has been done. The proposed methodology for identifying anomalous records will allow an earlier launch of investigation and correction. Consequently, the framework may prevent backlogs and result in timely submission of reports. Furthermore, the framework can also be extended to other remaining item groups that are not being audited because only two (2) out of twelve (12) groups are currently being audited.

This framework includes data visualizations and relevant statistics of detected cases of inconsistencies that may help the auditors to easily pinpoint irregularities and prioritize where to spend their time for analysis. It also empowers auditors to proactively conduct validation checks on a more comprehensive list of accounts with more accuracy. Furthermore, this project can serve as a benchmark framework for other regulatory institutions in monitoring and compilation of audit data.

Scope and Limitations of the Project

The report being audited records financial transactions in US dollars. It is in a Microsoft Excel Format consisting of twelve (12) item groups with more than 200 item codes. Items are grouped according to its similarity in nature. For this project, only one (1) item group will be selected to be in scope based on the available volume of records found during exploratory data analysis. Specifically, item group A with item type 1 related records were selected.

Available data covers daily records from 2015 to 2019. BSP has approved to use the dataset provided that the data will be anonymized prior to its use outside BSP network and premises and actual records from selected companies will not be included. For confidentiality purposes, anonymization and consolidation of data will be done by the researcher working in BSP.

Data cleansing was also necessary as reports submitted by companies were inputted manually. After data cleansing and standardization, the customer names were assumed as already correct due to numerous similar customer names with different spellings of the same customer. To correct such fell out of scope of this project, therefore, it would be efficient to simply assume that the customer names are correct. A single customer name database was developed based on the above algorithm. After these, the data files were consolidated to create an analytical base table. All values were in US dollars and rescaled to address confidentiality.

Dataset was further filtered based on the results of exploratory data analysis (EDA). Results of the EDA was used to identify the sample for each component of the proposed methodology. Since the product of this project is a prioritization framework for identifying anomalies, aside from selection of Item Group A-Type 1 records, each component was limited only to the sample selected for each component. Specifically, Component A was limited to the monthly total per item from 2015-2019, while Component B and C were limited to records from 2017-2019. Furthermore, for Component A, removal of patterns was limited to the removal of season and trend components, which are of the interest of the business.

Dataset that will be used for evaluation was limited only to records with available tag. Tag was provided by the auditor in charge of the selected items and was based on records that were audited, which implies that other records without available tag were not included in the scope of manual audit.

A full investigation of the reasons behind flagged data was also considered out of scope for this project for the following reasons:

1. Investigation involves coordination with the concerned company, requiring formal action by the BSP;
2. Company identities were masked, further limiting the context within which inferences could be made; and,
3. Inferences about inconsistencies as either intentional, unintentional, or indicative of business condition changes could potentially influence the perception of the public about the condition of the concerned company or the Philippine banking system.

Instead, this project provides three (3) neutral yet technical definitions of a normal record:

i. per item historical behavior
ii. relative to company type
iii. based on a customer behavior

REVIEW OF RELATED LITERATURE

Audit of Foreign Exchange records in The Bangko Sentral ng Pilipinas

Foreign exchange records form part of the balance of payments of a country, thus the need to have quality data is a must. Any item (goods, services or asset) that is exported from the country – its value should be reported. In the same way, any item imported in the country should be reported accordingly (BSP, 2020). Currently, companies submit subject records through a consolidated report weekly to the BSP through email and system checking is limited only in the report structure of the file. The system consolidates it into a monthly report and auditors manually conducts an initial audit based on different perspectives (i.e., average record, knowledge of the item, counterparties background, etc.). Each item is distributed to different auditors. Initial audit refers to initial checks (i.e., any missing field, names are correct, amount is within threshold, etc.). All questionable records based on the said criteria will be returned to the reporting company for confirmation and correction of report if necessary.

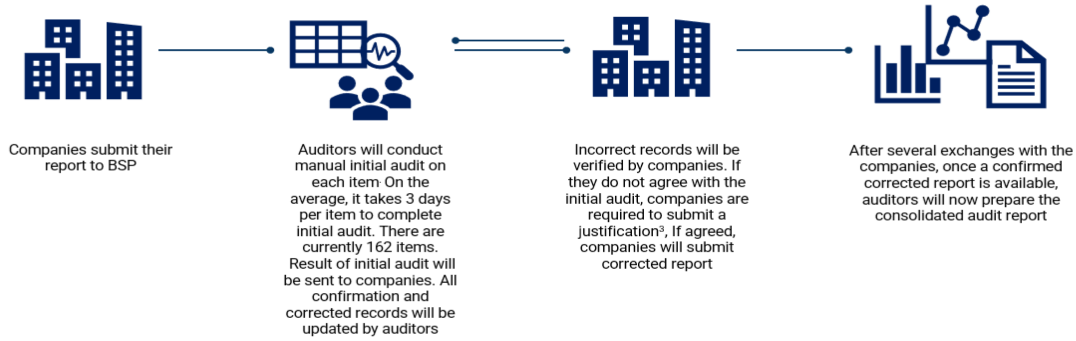


Figure 2: *Current audit process*

Techniques available for Anomaly Detection

Fraud is an uncommon, well considered, imperceptibly concealed, time evolving and often carefully organized crime which appears in many types of forms (Baesens et al., 2015). There are three main categories of algorithms for fraud or anomaly detection, namely supervised, unsupervised, and semi-supervised methods. Supervised methods use a labeled dataset, and the lack of it has led researchers to pay more attention to unsupervised learning methods in recent years (Kasuni et al., 2011). Unsupervised learning is the most flexible setup which does not require any labels. Furthermore, there is also no distinction between a training and a test dataset. The idea is that an unsupervised anomaly detection algorithm scores the data based on intrinsic properties of the dataset. Based on systematic literature research unsupervised outlier or anomaly detection techniques are categorized in: proximity-based techniques, subspace techniques and statistical / probabilistic models. Typically, distances or densities are used to give an estimation of what is normal and what is an outlier (Goldstein & Uchida, 2016). There are numerous applications of anomaly detection techniques for different types of data available. One application is an Autoregressive Integrated Moving Average (ARIMA) model that is fitted on the regular spending behavior of the customer and is used to detect frauds if some deviations or discrepancies appear (Moschini et al., 2020). The model was compared to four anomaly detection approaches such as K-means, Box-plot, Local Outlier Factor and Isolation Forest. The result of the study showed that the ARIMA model presents a better detecting power than the benchmarking model. It noted, however, that the study used a labeled dataset, and is limited for customers with complete daily count of records.

Meanwhile, in 2019, an experiment was conducted in the application of unsupervised outlier detection in financial statement audits (Lenderink, 2019). Isolation Forests (IF), K-Nearest Neighbors (KNN), Histogram-based Outlier Score (HBOS) and Autoencoder Neural Networks were selected in order to conduct experiments with. The selected techniques are evaluated based on their detection rate of the synthetic outliers. All outlier detection techniques have an outlier score as output, providing each journal entry with an outlier score.

Performance is measured based on the proportion of top journal entries that must be selected based on outlier score to obtain a recall of 100% for the synthetic outliers. In other words, sorting journal entries based on their outlier score, how many of these top scoring journal entries are to be included to contain all synthetic outliers. In the case of Isolation Forests, on the average, only the top 2:12% of journal entries include all synthetic outlying journal entries. This makes Isolation Forest the best performing outlier detection technique during these experiments. K-Nearest Neighbors scored a percentage of 19:31%, Histogram-based Outlier Score 3:54% and Autoencoder Neural Networks 56:78%. The experiment concluded that unsupervised outlier detection techniques and more specific, Isolation Forests, are suitable to detect outliers that are of interest during financial statement audits. Isolation Forests has been able to provide auditors with abnormal journal entries that haven't been detected following regular audit procedures. Applying these techniques therefore reduces the risk of missing anomalous journal entries that could be of interest and so improves the quality of financial statement audits.

In 2011, a group of researchers demonstrated the effectiveness of various statistical techniques for discovering quantitative data anomalies (Kasunic et al., 2011). The following tests were found to be effective when used for Earned value management variables that represent cumulative values: Grubbs' test, Rosner test, box plot, autoregressive integrated moving average (ARIMA), and the control chart for individuals. For variables related to contract values, the moving range control chart, moving range technique, ARIMA, and Tukey box plot were equally effective for identifying anomalies in the data. Among anomaly detection methodology, control charts have been considered important technique

As there are many anomaly detection techniques available, there are studies that suggest combining a set of methodologies for anomaly detection. The idea of developing a framework in identifying records for audit priority was inspired by a study presented in the 2017 Staff Working Paper from the Bank of England (Chakraborty and Joseph, 2017) where machine learning was used in predicting regulatory alerts on the balance sheet of financial institutions in an environment of incomplete information. The study created a stylized framework of identifying 3-level alerts and trained machine learning models on a set of supervisory alerts which indicate the need for closer scrutiny of a firm. The target variable was a binary classification if the account has 3-level alerts. The study employed Naïve Bayes classifier, k-nearest neighbor, decision trees and random forest machine learning techniques and found out that advanced machine learning approaches are seen to generally outperform conventional approaches. It concluded that the logic model does not perform considerably better in terms of accuracy than the trivial benchmark of never raising an alert. On the other hand, most models' test performance plateaus at around 92% accuracy, which is an example of the at-maximum effect. It states that there is no substantially best model in many situations, but many different models may show similar performance. A small deviation from the at-maximum effect is the slightly better performance of the random forest classifier. This is an example of an appropriate model choice as the intrinsic working of random forests matches well the data generation process. Namely, by a combination of thresholding, a non-trivial rule of combining three or more thresholds and noise inductions through the removal of variables. Furthermore, the paper pointed out that since balance sheet items are not independent from each other, Naive Bayes classifier showed a relatively poor performance compared to the other models as the data invalidated the "naïve" base model assumption.

The studies presented above served as a guide in identifying the machine learning model techniques that will be applied to address the objectives of the current project. Specifically, the related studies guided the current project in the development of framework in determining anomalous records, and in the application of the selected machine learning techniques.

1. METHODOLOGY

In this section, the following main activities which supports completion of the project will be discussed:

- Exploratory data analysis to determine scope, inclusion criteria and limitations or constraints of the project, feature engineering and building of analytical base table (ABT); and
- Developing, testing, and evaluating machine learning models for identifying priority level.

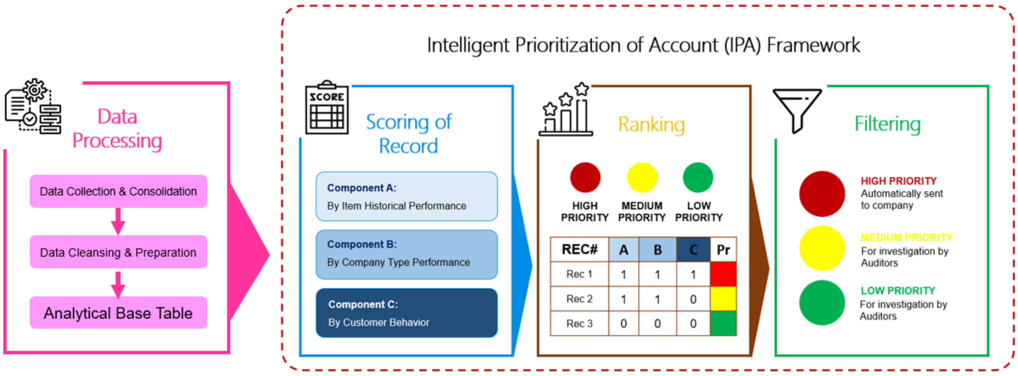


Figure 4: *Overview of the proposed intelligent prioritization of account framework*

1.1 Model Development and Simulation

From the original dataset, three subsets were created and used in the three components of the framework. The development of the framework being proposed is guided by three perspectives: first, an item code is seen to have intrinsic behavior on its own, leading to Component A. Second, using behavior of a company based on its company type guided Component B. Finally, movement of the level of records of a person based on its historical records led to Component C. Each record will be tagged as anomalous based on the priority definition threshold from these three components.

1.1.1 Component A: Identify anomalies per item based on historical behavior using time series decomposition

Records vary on a day to day basis. Currently, companies submit their report weekly with daily information which then is consolidated to a monthly report. Auditors tag an item for priority if the value of the record is above the mean value of all the records for the month for that item. Anomaly detection problem for time series is usually formulated as finding outlier data points relative to some standard or usual signal. While there are plenty of anomaly types, this project focused only on the most important ones from a business perspective, such as unexpected spikes, drops, trend changes and level shifts. Specifically, component A used time series decomposition for the detection of anomalous behavior of an item. Currently, an item is considered for priority of audit if the level of its total amount is more than the average amount for the year. In this project, we focused on removing "normal" patterns from the series and analyzed the irregular series to determine and establish the level of what is acceptable and what is not per item.

The available dataset contains 60 time series data points (equivalent to five (5) years' worth of monthly data). The objective of this component was to use the available data points to identify the presence of pattern (i.e., seasonality and trend) in each series, remove those patterns, then propose rules in identifying anomalies on the residuals for each item that may be used by the auditors for future audit. Rules would be item-specific thus, items that deviates from the rule provided should be considered as priority for audit. This process should be done to all selected items (62 items).

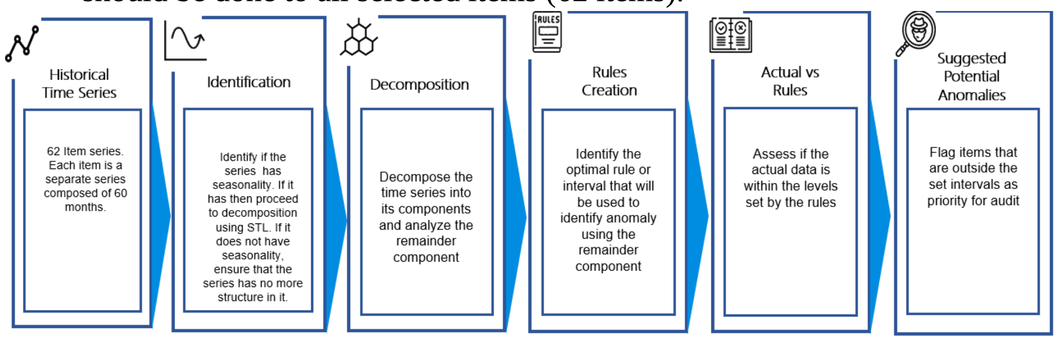


Figure 6: *Component A Process Overview*

The overall process of component A is presented in Figure 6. The first task was to determine if the series is deconstruct-able – meaning if there are patterns present. If there is an identified pattern, then the given time series was deconstructed by removing the identified patterns, such as seasonality and trend, until a residual is achieved. The residual was then analyzed to determine the levels that will define what is suspected anomaly and what is not for each specific item. Identification characteristics (or patterns) of each time series was done through time plots and were further verified by statistical tests.

Seasonality of a series refers to its predictable changes affected by seasonal factors such as time of the year or day of the week. For this project, seasonality and trend were considered as part of the "normal" behavior of each series, thus the need to be taken out from the original series. Specifically, seasonality was checked using the WO overall seasonality test which was developed by Webel and Ollech (2018). By default, the WO-test combines the results of the QS-test and the Kruskal Wallis test, both calculated on the residuals of an automatic non-seasonal ARIMA model. If the p-value of the QS-test is below 0.01 or the p-value of the Kruskal Wallis test is below 0.002, the WO-test will classify the corresponding time series as seasonal.

For series that are found to be seasonal, removing the season and trend of the subject series was done to get its irregular component. Meanwhile, for series that are not seasonal, detrending the series was done to get its irregular component.

Non-seasonal time series consists of trend components and irregular components. Decomposing the subject time series involves trying to separate its trend and irregular component. In this project, it is imposed that a trend structure is present thus it was estimated using simple moving average. After noticing that this resulted to a flat structure in the beginning and last part of the series, spline technique was explored in approximating the trend. Spline, in its simplest sense, is a tool that is used to draw smooth curves between points in a metal. In statistics, splines are used in order to mathematically reproduce flexible shapes. Several weights (or knots) are placed on various positions within the data range, to identify the points where adjacent functional pieces join each other. Smooth functional pieces (usually low-order polynomials) are chosen to fit the data between knots. The type of polynomial and the number and placement of knots is what then defines the type of spline (Perperoglou et al., 2019)

Meanwhile, the seasonal time series assumed to consist of a trend component, a seasonal component and irregular component. To identify the seasonal and trend component of subject series, decomposition of seasonal series was done using the Seasonal and Trend decomposition using Loess (STL) decomposition technique. Unlike high performance machine learning techniques which perform poorly for anomaly detection because of overfitting, seasonal decomposition does very well for this task, removing the right features (i.e., seasonal and trend components) while preserving the characteristics of anomalies in the residuals. In STL, it is assumed that a time series can be decomposed as the sum of trend, seasonality, and remainder components: $\mathbf{y_t} = \boldsymbol{\tau_t} + \mathbf{s_t} + \mathbf{r_t}$, $\mathbf{t} = 1, 2, \cdot\cdot\cdot, N$ where y_t denotes the original observation at time t,

$\boldsymbol{\tau_t}$ denotes the trend,
$\mathbf{s_t}$ denotes the seasonality if the time series is periodic and
$\mathbf{r_t}$ is the irregular component.

The irregular component will be the de-seasonalized detrended series

The irregular component (the residuals), or what is left over in a time series after decomposition, was checked to ensure that the decomposition removed the seasonality and trend, meaning there is no more (or close to none) pattern left. A good decomposition will produce an irregular component that are uncorrelated and has zero mean. Aside from time plot, these two properties were checked using AutoCorrelation Function (ACF) plot, histogram of the residuals (with an overlaid normal distribution for comparison), and Ljung-Box test with the correct degrees of freedom. ACF is an (complete) auto-correlation function which gives us values of auto-correlation of any series with its lagged values. In simple terms, it describes how well the present value of the series is related with its past values. There are several autocorrelation coefficients, corresponding to each panel in the lag plot. For example, r_1 measures the relationship between y_t and y_{t-1} , r_2 measures the relationship between y_t and y_{t-2} , and so on. The value of r_k can be written as

$$r_k = \frac{\sum_{t=k+1}^{T} (y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=1}^{T} (y_t - \bar{y})^2}$$

where T is the length of the time series.

Meanwhile, Ljung Box-test is a more formal test for autocorrelation by considering a whole set of r_k values as a group, rather than treating each one separately. Specifically, Ljung Box-test is based on

$$Q^* = T(T+2)\sum_{k=1}^{h}(T-k)^{-1}r_k^2.$$

where h is the maximum lag being considered and T is the number of observations.

The remainder of all the decomposed series, both the seasonal and not seasonal series, will be identified if white noise or not. If found to be not white noise, the series will be tested for stationarity using the ndiffs and nsdiffs function in r. Said functions uses a unit root test (i.e., kpss test) to estimate the number off differences (non-seasonal and seasonal respectively) required to make the given series stationary. Once the series is stationary and the remainder component is still not white noise then the original series will be modeled through the *auto.arima*() function in R Software.

After decomposition, the remainder or residual was used in establishing the rules for identifying suspected anomalies in the series. Establishing the rules based on the remainder component was done using two methods: the first one is the InterQuartile Range (IQR), which is a measure of variability based on dividing a dataset into quartiles (Dodge, 2008). It takes a distribution and uses the 25% and 75% interquartile range to establish the distribution of the irregular. Detecting anomalies using IQR methods requires setting a decision range, where any data point lying outside this range is considered as anomaly. In this project, the decision range was set to a factor of three (3) times above the 75th inter quartile and there (3) times below the 25th inter quartile range, and any points beyond the limits were considered anomalies.

The next method is the Generalized extreme studentized deviate test (GESD). It is an iterative hypothesis test proposed by Rosner in 1983. In this test, the upper bound or the total number of outlier values is given in the null hypothesis. After that, a separate test is performed by using the Grubbs statistics as given in (Cohn et al., 2013)

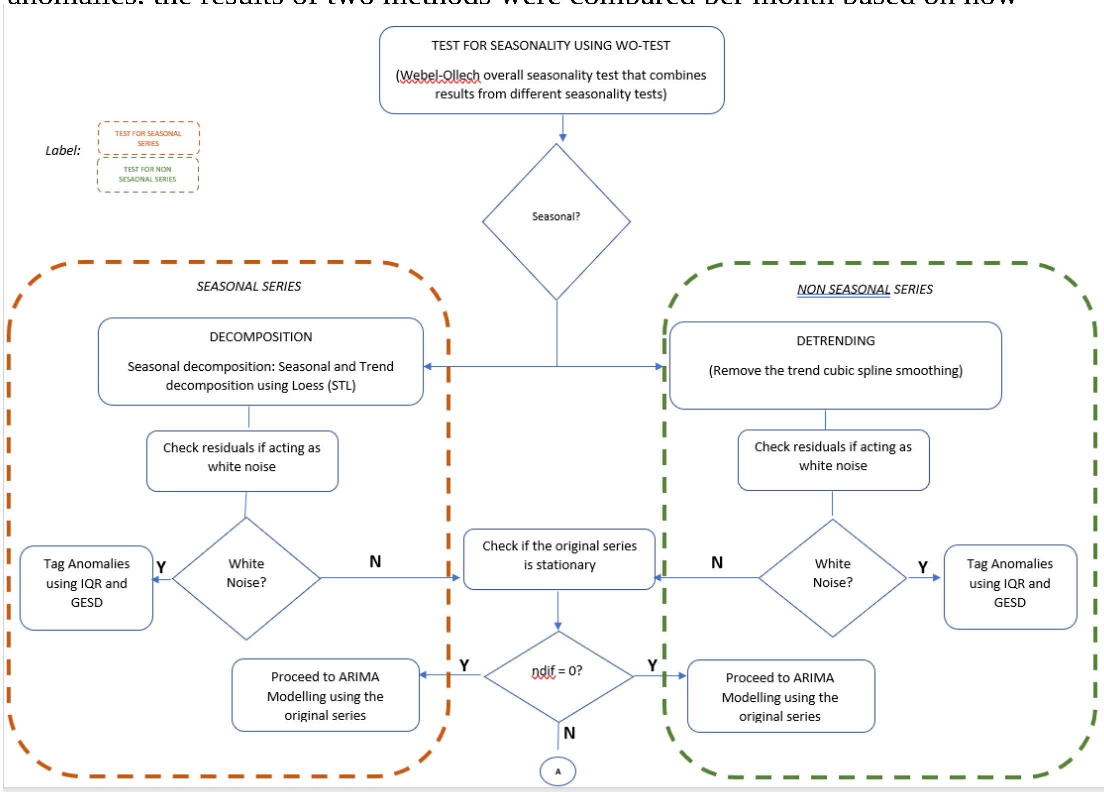
$$T_k = \frac{\max \left| \mathbf{z}_i - M \right|}{\sigma},$$

where M and σ denote the mean and standard deviations in the data. The observation corresponding to

$$\max \left| \mathbf{z}_i - M \right|$$

is removed using Grubbs statistics, and T_2 is computed from the remaining sample. A sample mean and standard deviation are computed for the remaining n-1 data values. This process is repeated until $\mathbf{T_k}$ is determined for a prespecified *k*. Here, k represents the number of outliers in the data set known as the upper bound specified in the null hypothesis (Hyndman & Athanasopoulos, 2018).

Since the objective of the project is prioritization of items by detecting anomalies, the results of two methods were compared per month based on how



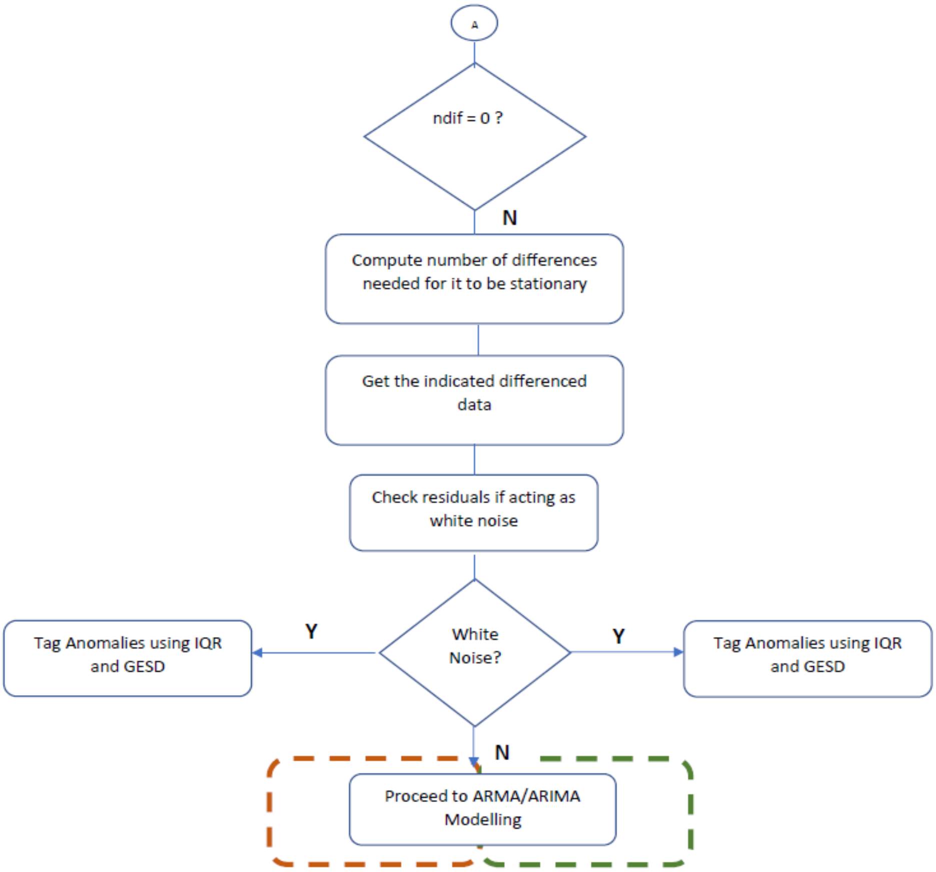


Figure 7: *Component A Decomposition Process*

The flowchart above shows the detailed proposed process of decomposition of component A. Above flowchart was created to ensure proper case handling of each available series. In case the residual is still not acting as white noise after decomposing, either seasonal series through STL or a non seasonal series through cubic spline smoothing, or after differencing, the model for the series will be selected through *auto.arima()* function available in R. Said function uses a variation of the Hyndman-Khandakar algorithm (Hyndman & Khandakar, 2008), which combines unit root tests, minimization of the AIC and MLE to obtain the best possible ARIMA model.

1.1.2 Component B: Identify anomalous records using an Unsupervised Outlier Detection Approach on a Company-Level

Meanwhile, Component B, considered the behavior of the records per company type – and in assessing anomalies, an unsupervised machine learning model called the Isolation Forest was adopted.

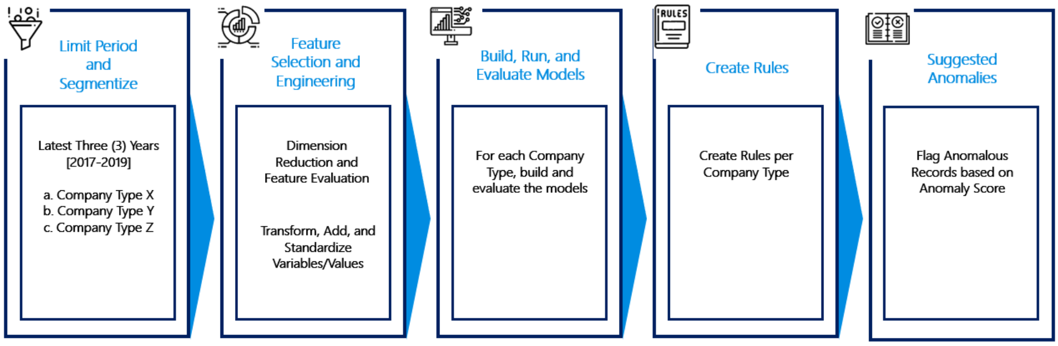


Figure 8: *Process Overview of Component B*

In this component, the data set was limited to the most recent period, that is from 2017-2019 or 3 years, and segmentized it according to company type: X Y or Z. Second, features were selected and engineered from the filtered data set. Third, Isolation Forest Models were built, run, and evaluated for each of the Company Types. Fourth, rules were created based on these models, and lastly, the rules were implemented to flag anomalous transactions.

Records are attributable to the company type's layering and structure particularly in their assets and capacity, that is, for each company type, there are different levels of products and services that the companies under it can offer. For instance, Companies categorized to be in Company Type X are companies with relatively lower capacity and offer a limited number of products and services, while Company Type Z are companies with full capacity to offer an extensive set of products and services. Essentially, at the minimum three Models were created for Component B for each of these Company Types.

Data Preparation

Before the implementation of the models, the variation in magnitude was addressed. Numeric variable was standardized from 0 to 1, so that the data is internally consistent and comparable with other data points. The aim of this step was to standardize the range of amounts so that each item contributes equally to the analysis.

Additionally, since the data is a mixed data set (numeric and categorical), categorical data was converted to ensure that it would be understood by the machine. Initially, Label encoder in sklearn (python) was proposed to be used for the preprocessing of categorical data. However, since subject function only assigns a unique integer based on alphabetical ordering which in turn produces an ordered value. This will result to issues in prediction and poor performance of data modeling thus One-Hot encoding is more appropriate to use. With one-hot, each categorical value is converted into a new categorical column and assign a binary value of 1 or 0 to those columns. Each integer value is represented as a binary vector. One hot encoding makes the data more useful, easily rescalable and better for prediction

Feature Selection and Engineering, Dimension Reduction

Since one-hot encoding produces additional feature for every value in the categorical variable, this expanded the dimensionality of the dataset. The dimensionality of the data set and the predictive importance of both original and engineered features were handled using a dimension reduction technique called Principal Component Analysis (PCA).

PCA is an unsupervised dimensionality reduction technique that can be used to create a compact representation of the dataset while minimizing information loss. For instance, if a data set is represented as vectors in a high-dimensional space, it might be observed that numerous variables are correlated, and that the data closely fits a lower dimensional linear manifold. PCA can be used to find the lower dimensional representation in terms of uncorrelated variables called principal components (Hastie, et al., 2014).

PCA constructs relevant features by transforming correlated features linearly into fewer uncorrelated variables, called principal components, by projecting the original data into the reduced PCA space using the eigenvectors of the covariance/correlation matrix. The resulting projected data are essentially linear combinations of the original data capturing most of the variance in the data (Jolliffe 2002).

PCA is an orthogonal transformation of data into multiple uncorrelated data in a space wherein we can extract information such as the first few components can already explain the most variance in the data with each subsequent component explaining less. The goals of PCA can be summarized below (Hastie, et al., 2014).

1. extract the most important information from the data table;
2. compress the size of the data set by keeping only this important information;
3. simplify the description of the data set; and,
4. analyze the structure of the observations and the variables.

In order to evaluate the predictive importance of the variables and engineered features, a PCA score called Loadings was utilized in this project. Loadings is the correlation between a principal component and a variable and estimates the information they share (Hastie, et al., 2014).

The importance of each feature is reflected by the magnitude of the corresponding values in the eigenvectors (higher magnitude — higher importance). As an example, from the loadings score, information on how important features 1, 2 and 3 are for the first component can be acquired. Similarly, determining which features are most important for the second component, and so on, can be revealed.

In short, the absolute values of the eigenvectors' components corresponding to the m largest eigenvalues can be checked to determine the more important feature. The larger the absolute values, the more important the feature is in contributing to a particular principal component.

This information was used in assessing whether to keep or drop the features before running it to the model. The result of the feature selection was evaluated through trials by variation, and by comparing its effects to the recall rate of the framework using the available labeled dataset.

Build, Run and Evaluate Model using Isolation Forest

In nature, anomalies are difficult to identify due to the following:

1. Severe Class Imbalance: Anomalies or outliers in general are much fewer than our normal records
2. Severe Class Overlap: The reason why we audit is difficult is because there is a small gap between legal and fraudulent activities
3. Concept Drift: Anomalies can be done in different ways and evolves and changes in time
4. Complexity and volume of our data

From the above characteristics, there are two distinct properties of an anomaly (Liu, Ting, & Zhou, 2008):

a. They are the minority consisting of fewer instances
b. They have attribute-values that are very different from what we consider normal

In other words, anomalies are "few and different". One logical way to identify it is to isolate it from the rest and this is where the idea of "Isolation" comes in. Isolation means separating an instance from the rest of the instances. After which, data-induced random tree will be produced to partition instances that are few and different from the rest. In doing so, it can be noted that the random partitioning will produce noticeable shorter paths, or in laymen, isolates sooner, for anomalous instances.

It was Lui, Ting & Zhou who first proposed a tree-based unsupervised outlier detection technique. Lui describes the term 'isolation' as *'separating an instance from the rest of the instances'*. The researchers note that iForest shares intuitive similarity to random forest, another tree-based algorithm but is mainly used for classification problems.

iForest functions under the assumption that it is more likely to be able to isolate outliers. Hence, when a forest of random trees collectively produces shorter path lengths for some points, those points are likely to be anomalous :

i. In a single isolation tree, the data is recursively partitioned with axis-parallel cuts at randomly chosen partition points in randomly selected attributes (features).

ii. This is done for n data points to isolate the points into nodes with fewer and fewer points until they are isolated in singleton nodes containing one instance.
iii. The intuition behind the technique is that tree branches containing outliers are noticeably less deep, because these data points are located in sparse locations.
iv. The distance of the leaf to the root is used as the outlier score.
v. Since iForest creates multiple trees (n estimators) the average path length for each data point is calculated over the different trees in the isolation forest.

Using this average path length, an "Anomaly Score" will be computed.

Create Rules: Model Algorithm and Parameter Tuning

Isolation Forest is black-box methodology. Its algorithm is illustrated in a simplified pseudocode below:

1. Randomly select two features or set of features
2. Split the data points by randomly selecting a value between the maximum and the minimum
3. Repeat step 2 iteratively until fewer and different data points are isolated
4. Termination point is until everything is split, or the data points are completely duplicate
5. Calculate the "anomaly score" for each tree and average across. Outliers with lower path length will have higher anomaly scores, and thus, tagged as anomalous.

In implementing the model, scikit-learn in Python was used for the black-box algorithm and model parameters such as n_estimators, max_sample, Contamination, and max_features, were tuned.

Key Parameters	Description
n_estimators int, default=100	The number of base estimators in the ensemble.
max_samples "auto", int or float, default="auto"	The number of samples to draw from X to train each base estimator.
Contamination 'auto' or float, default='auto'	The amount of contamination of the data set, i.e. the proportion of outliers in the data set.
max_features int or float, default=1.0	The number of features to draw from X to train each base estimator.

Table 3: *Isolation Forest Key Parameters*

Based on research, iForest model is quite sensitive to the parameter Contamination relative to its other parameters. Contamination is the proportion of outliers or anomalies in the dataset which in this project, based on business domain understanding, to be around 10%. For this project, contamination was set and tested from seven and a half percent to twelve percent (7.5% to 12.5%).

Suggested Anomalies: Flag Anomalous Records using Anomaly Score

In creating iTrees, a data point x in a sample size n was used to predict an output called anomaly score using the formula (Liu, Ting, & Zhou, 2008):

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$

E(h(x)) is the expected value of the average path length or search height of x, meaning how soon can the data point x be separated, c(n), the denominator, answers what is the average path length that it would take to find any general node, not just the data point x, across all of the trees

The score shows how long it takes to isolate the particular point x relative to isolating every other data point. If E(h(x) is much lower than c(n), then the anomaly scores(x,n) will be nearer to 1. For example, if E(h(x)) is 2 and the sample average c(n) is 5, then the anomaly score is 2 raised to negative 2 divided 5, which is 0.76 and is nearer to 1

And if E(h(x)) is about the same as c(n), then the anomaly score s(x,n) will be lower, or if both are exactly the same, it will be exactly 0.5. For example, if E(h(x)) is 5 and our c(n) is 5, then the anomaly score is 2 raised to negative 5 divided 5, which is 0.5.

Notice that the score will become higher, if E(h(x)) is much lower than the average path length, meaning, if x isolates much sooner compared to the isolation of other data points, then that point is considered as anomalous.

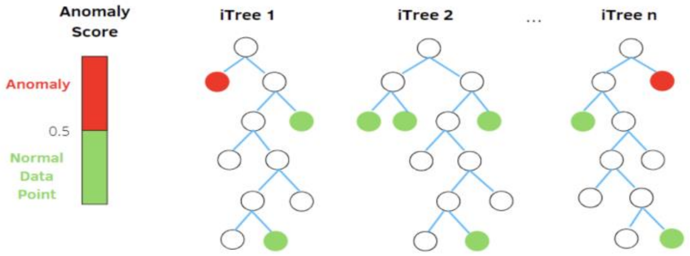


Figure 9: *Anomaly Detection using Isolation Forest**

*Illustrated by E. Anello (betterprogramming.pub)

1.1.3 Component C: Determine the behavior of a transactor using customer segmentation

Component C is a new perspective in terms of the current audit process. The objective of this method is to determine the customer behavioral segments that transacts on different bank types and detect any anomalies on their transactions. In order to achieve it, Data Pre-processing and Clustering Technique were applied. Data Pre-processing is already defined in Section 3.2 thus the discussion will proceed on how clustering technique was applied.

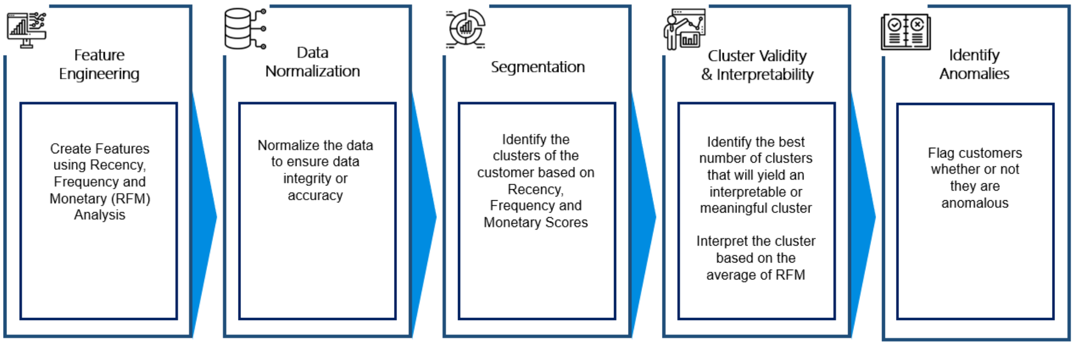


Figure 10. *Component C Process Overview*

i. ***Feature Engineering***. To identify the most effective subset of the original features to use in clustering, Feature Engineering was done. For Feature Engineering, Recency, Frequency and Monetary (RFM) (Cullinan, 1977) were used as attributes of concern. In order to compute the Recency, Frequency and Monetary, the following formula were used:

Recency

Recency is the number of days between the first date of the period examined (1/1/2017) and the date of the customer's last record. It answers the question, how recent was the customer's last record? For example, a customer who has conducted his last record on 03/15/2019 is characterized by R=803

Frequency (F)

Frequency is defined as the count of financial records the customer did within the period of interest (1/1/2017 to 12/31/2019). It answers the question, how often did this customer make a record in a given period?

Monetary (M)

Monetary is the total value of financial records the customer made within the examined period. It answers the question, how much money did the customer spend in a given period?

RFM Score (RFM Factor)

It is calculated using the formula

$$RFM_{Score} = R + F + M$$

ii. ***Data Normalization***. Normalization of data or Feature Scaling is an important step prior to the actual clustering because it enables the reduction of the scale of the variables which affects the statistical distribution of the data. Based on the unit of measurement of the RFM data in this project, monetary has a larger scale compared to Recency and Frequency. To do the Feature Scaling, Min-Max Normalization was applied. The data values were scaled between a range of 0 to 1 only. Consequently, the effect of outliers on the data suppresses. Also, it generates a smaller value of the standard deviation of the data scale. The formula for Min-Max Scaling is as follows:

$$\mathbf{M} = (X - X_{min}) / (X_{max} - X_{min})$$

Where:

M is our new value

X is the original cell value

$\mathbf{X}_{min}$ is the minimum value of the column

$\mathbf{X}_{max}$ is the maximum value of the column

iii. ***Segmentation***. Clustering identifies which observations are alike, and potentially categorize them therein. For this project, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) was used. The DBSCAN algorithm uses two parameters:

minPts: The minimum number of points grouped together for a region to be considered dense. This will be the threshold.
eps (ε): A dissimilarity measure that will be used to locate data points in the neighborhood of any datapoint.

It can be more explained using the terms Density Reachability and Density Connectivity.

Reachability in terms of density establishes a point to be reachable from another if it lies within a distance (eps) from it.
Connectivity, on the other hand, involves a transitivity-based chaining-approach to determine whether points are in a cluster.

For comparison, a centroid-based clustering, specifically k-medoids, was also utilized. With the objective of minimizing the dissimilarity of all the observations to the nearest medoid, the Clustering Large Applications based upon Randomized search (CLARANS) was employed in this project. CLARANS is a partitioning method of clustering that searches a graph where every node, k medoids, is a potential solution.

iv. ***Cluster Validity and Interpretability***. To identify the most optimal number of clusters we will use Dunn Index (DI) (Yang et al., 2014). Dunn Index (DI) is calculated based on the following equation:

$$D_{nc} = \frac{min}{i = 1, \dots, nc}\left[min\ j = i + 1, \dots, nc\ \left(\frac{d(c_i, c_j)}{max_{k=1,\dots,nc} diam(c_k)}\right)\right]$$

Where $d(c_i, c_j)$ is different function between cluster $c_i and\ c_j$ defined as:

$$d(c_i, c_j) = \frac{min}{x \in c_i, y \in c_j} d(x, y)$$

and $diam(c)$ is cluster diameter probably considered as cluster dispersion size. Cluster diameter of C can be defined as flows:

$$diam(C) = \frac{max}{x, y \in C} d(x, y)$$

v. ***Identify Anomalies***. In this project there are two assumptions that can be considered as anomalies in using clustering.
Noise is considered as anomalous (Ester et al., 1996). Example in below figure, Cluster 1 and Cluster 2 are clusters containing normal instances A1 and A2 are considered anomalous. This can be detected using DBSCAN Clustering.

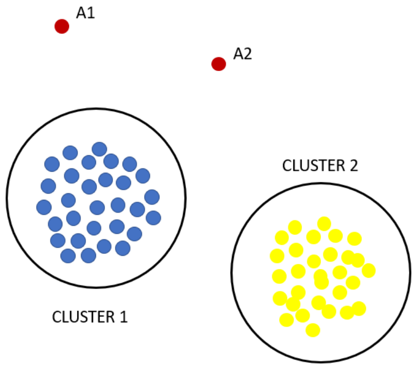


Figure 11(a). *Noise is considered as anomalous*

Anomalies are far away from the centroid. Under this assumption, anomalous events are detected using a distance score. This can be detected using CLARANS Clustering. See below Figure.

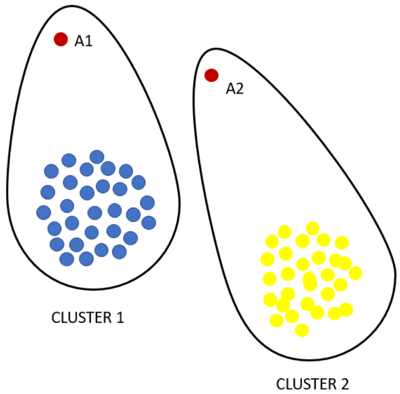


Figure 11 (b). *Anomalies are far away from the centroid*

The data points or records identified as anomalous will be considered as a Priority for Component C and will be added in other priorities done in Component A and B for Priority Ranking and Filtering which will be discussed in the next section.

1.1.4 Priority Ranking and Filtering

The combined results from the three components was used to identify the priority level of a record. A sample priority matrix result can be seen in below table:

Record No.	A	B	C	Priority Level
Record 1	Priority	Priority	Priority	HIGH
Record 2	Priority	Priority	Not Priority	MEDIUM
Record 3	Priority	Not Priority	Priority	MEDIUM
Record 4	Not Priority	Not Priority	Not Priority	LOW
Record 5	Not Priority	Priority	Priority	MEDIUM
Record 6	Not Priority	Priority	Not Priority	MEDIUM
⋮	⋮	⋮	⋮	⋮
Record N	Not Priority	Not Priority	Priority	MEDIUM

Table 4: *Sample Priority Matrix Result*

Records were ranked according to its level of priority for audit and were filtered accordingly. Table 5 presents the Priority rating definition that was set for this project and can be used by the auditors.

HIGH PRIORITY	Records with high priority for audit are those that violated the priority definition (threshold) of the 3 components. These records need to be verified further by the reporting company immediately. Auditors will just consolidate identified records and require companies for justification.
MEDIUM PRIORITY	Records with medium priority for audit are those that violated the priority definition of 1 or 2 components. Auditors will conduct its analysis of the record and will apply domain knowledge if the record needs further justification or not.
LOW PRIORITY	Records with low priority for audit are those that fall within the normal priority definition of the 3 components and need minimal verification of the auditors.

Table 5: *Priority Rating Definition*

1.1.5 Model and Framework Evaluation

The effectiveness of the framework being proposed was evaluated against the actual anomalous records dataset for 2018-2019 (with an average of two thousand records per month) provided by the BSP, and the following measures were used in determining the performance of the proposed priority framework:

- Classification Accuracy is the percentage of correctly classified observations.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- Sensitivity, or recall measures how many of the anomalies are correctly labeled as anomalies

$$\text{Recall} = \frac{TP}{TP + FN}$$

- Precision indicates how many of those tagged as anomalous are anomalies

$$\text{Precision} = \frac{TP}{TP + FP}$$

- F-measure is a measure of accuracy

$$F = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

- Mcnemar’s Test of Con

$$\chi^2 = \frac{(b - c)^2}{b + c}.$$

RESULTS AND DISCUSSIONS

Component A: Identify anomalies per item based on historical behavior using time series decomposition

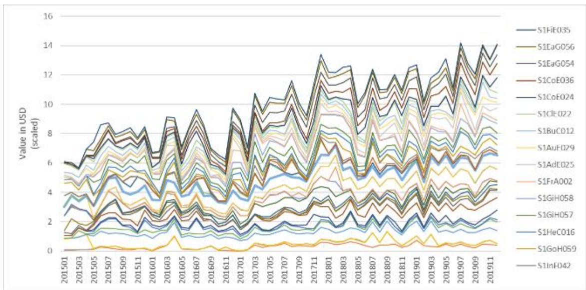
Component A uses time series decomposition to analyze each item series and establish the rules that can be used by the auditors in identifying anomalies. This approach mimics the current steps being done by the auditors to assess the consistency of the behavior of an item series by looking into its historical performance. In this project, out of 162 items available in the dataset, Component A was limited to item group A - item type 1 record, 62 in total, from January 2015 to December 2019 (60 available data points per item), as most of the volume of records fall in this item group.

Each item follows different behaviors in terms of complete reporting and can be classified into following categories:

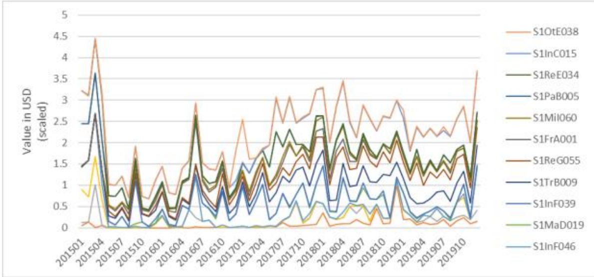
No.	Category based on completeness	Items
1	With records every month	56%
2	With less than 10 months of no records	23%
3	With more than 10 months but less than 30 months without records	8%
4	With 30 or more months without records	13%

Table 7: *Categories of items based on months with transactions*

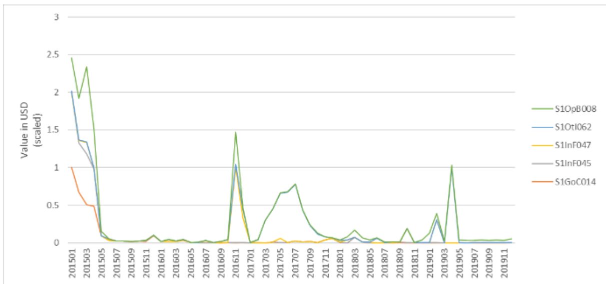
As seen in the initial exploratory data analysis, values of each item vary greatly. To address this in further illustration, scaling the values for each item was done to ensure that each item series will be compared in equal footing.



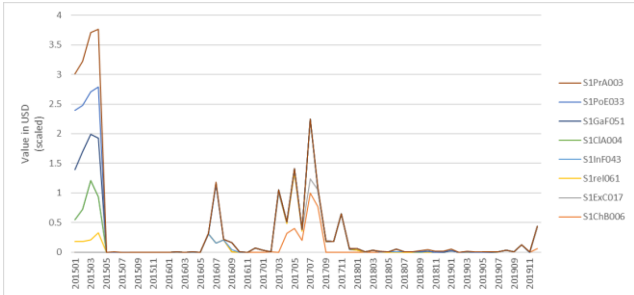
(a) : Intermittent Time Series: Historical Series of items under category 1



(b) : Intermittent Time Series: Historical Series of items under category 2



(c) : Intermittent Time Series: Historical Series of items under category 3



(d) : Intermittent Time Series: Historical Series of items under category 3

Figure 18 : *Comparison of historical series of each item*

In some of the observations, zero values can be found either in consecutive months (Figure 18 c and d), while for other series, zero values can be found intermittently (Figure 18 b). Based on this, it can be observed that item series that falls under the same category almost follow the same behavior and has the same nature of fluctuations over time. As an example, it is shown in Figure 18 a, that item series without zero value for any time period almost follow the same trend. This observation of intermittent behavior of zero values in the records may be reflective of a change in economic condition or business status which may serve as a springboard for further investigation of the auditors. As such, this observation led to the organization of items into two (2) major categories:

i. Items with consecutive zero values – Static Item Series

In business perspective, each item is expected to have a record for at least three consecutive months. In this project, it has been found that it is not the case. Some items do not have a reporting for the entire year (value and volume is zero), and then will have a record for just 1 month in the next year, such that this record is either static (no value) during the first period then has a record in between or has record at first then static (no value) onwards. This behavior signals a need for investigation as it may indicate either a new business for close monitoring or an intentional error. As such, items having that kind of behavior were considered static and will be flagged as anomaly for that particular month, if that item deviates or is different from its value (zero value and volume) for the last three months.

ii. All other items – Non Static Items

Some items has an intermittent behavior which may be reflective of error in the report. Thus, the categorization is important for the auditors to determine which item should be focused on.

Static Items	12 items series
Non Static Items	50 item series

Table 8: *Category of items based on behavior*

4.2.1 Identification of seasonality

Time series exhibit a variety of patterns and decomposing its components will be helpful as each pattern represents and underlying pattern category (Heinze, 2018). In this project, patterns being considered as “normal” components of each series were seasonality and trend. The first pattern that was identified was the seasonality of each item. It was commonly believed that foreign exchange transactions were affected by seasonality. Seasonality is a component of time series which data is affected by regular and predictable seasonal factors such as time of the year or day of the week. Presence of seasonality in each 62 series were tested prior to decomposition of the series using the WO overall seasonality test, which combines the result of the QS-test and Kruskal Wallis test. In the said test, if the p-value of the QS-test is below 0.01 or the p-value of the Kruskal Wallis test is below 0.002, the WO-test will classify the corresponding time series as seasonal – this is the default setting of the test.

Contrary to the belief that foreign exchange transactions, regardless of purpose, are seasonal in general, based on the test out of the 62 items series only one item series (S1ReG055) was found to have a seasonal component, meaning only one item is influenced by changes in time factors. Said item series is related to earnings of residents working in supranational companies. During 2015 to 2017, the value of the subject item series has its peak during the last quarter of the year. This changed in 2018 to 2019, which moved the peak to the first quarter of the year, which maybe reflective of a change in policy in giving earnings of workers amongst supranational companies (i.e., United Nation, International Monetary Fund, etc.).

It was noted that the WO-test for seasonality has a very stringent test, with *p-value* at .01 of the QS-test while the *p-value* of the Kruskal Wallis test at 0.002. Thus, the default *p-value* was changed to 0.05, and all item series was retested for seasonality. This increased the number of item series that were to be seasonal from one item series to seven item series.

For the remaining series which were found not to have a seasonal component, trend was identified. Trend is a general direction in which the series is moving. It is characterized to be whether increasing or decreasing (Gurung & Perlman, 2018). In this project, items that were not seasonal and non-static were considered to be non-seasonal items with trend. It is believed that observations that are nearby in time are likely to be close in value, thus taking the moving average eliminates randomness in the data, leaving the trend component of the series. Initially, trend was identified using the *ma()* function in r which determines the trend of a series using moving average but after observing that this produces a flat line in the beginning and end part of the series, the *splinef()* function in R Software was explored. *splinef()* function fits a model using cubic spline smoothing which provides a smooth historical trend as well as linear forecast function (Hyndman et al., 2005).

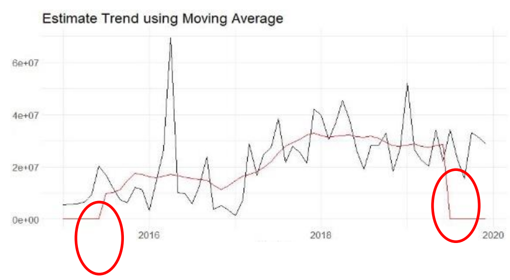


Figure 23 : *Sample item series with trend estimated using moving average*

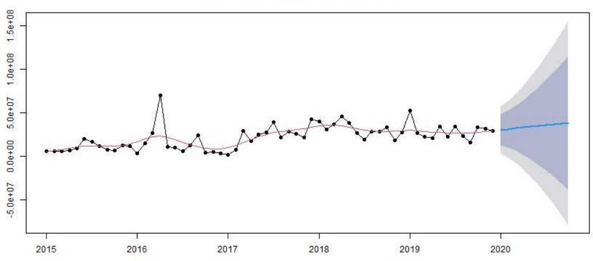


Figure 24 : *Sample item series with trend estimated using cubic spline smoothing*

It can be observed that the trend is better captured by cubic spline smoothing than the moving average. Thus after noting the limitations of the moving average in capturing the trend at the beginning and end of the series, spline smoothing was selected.

4.2.2 Decomposition of components

Decomposition was done for 62 item series, except for items that were considered as static. For item series found to have a seasonal component, decomposition was done using the STL technique. There are other traditional methods available in decomposing time series but for this project, STL technique was used in decomposing seasonal series to make the process more efficient and less time consuming. Meanwhile for non-seasonal items, the *ma()* and *splinef()* function in R Software were used to determine the trend and subtract it from the original series to get the remainder component. Consequently, the auto-correlation function (ACF) of the residuals of each item was checked. This is done to verify that there is no more (or close to none) pattern left in the series. ACF generates values of auto-correlation of any series with its lagged values. In simple terms, it describes how well the present value of the series is related from its past values.

Furthermore, each series was tested using the Ljung-Box test which is a more formal test for autocorrelation. Ljung- Box test is one of the statistical tests that checks if autocorrelation exists in a given series. For the remainder of series which still does not behave like a white noise after the proposed decomposition, either seasonal or not seasonal, subject series were transformed to be stationary. Subsequently, it was tested again for white noise and series which are still found to be "not white noise" was modeled through *auto.arima()* function in R Software.

For clarity, below are the sample cases that may be present in the data with corresponding recommended model:

CASE	SERIES TYPE	MODEL
1	Seasonal Series – Decomposed using STL – Remainder White Noise	STL MODEL
2	Seasonal Series – Decomposed using STL – Remainder not white noise – Stationary Series – Remainder White Noise	SEASONAL DIFFERENCED MODEL
3	Seasonal Series – Decomposed using STL - Seasonal Series– Remainder not white noise – Stationary Series – Remainder not White Noise	ARIMA MODEL
4	Non seasonal Series – Decomposed using detrending methods (i.e., Moving average, Spline) – Remainder White Noise	DETRENDED MODEL
5	Non seasonal Series – Decomposed using detrending methods (i.e., Moving average, Spline) – Remainder not white noise – Stationary Series – Remainder White Noise	NON-SEASONAL DIFFERENCED MODEL
6	Non seasonal Series – Decomposed using detrending methods (i.e., Moving average, Spline) — Remainder not white noise – Stationary Series – Remainder not White Noise	ARMA MODEL

Table 10: *Case Handling Scenarios*

4.2.3 Flagging potential anomalies

To establish the rules of a normal level for each series, the Inter-Quartile Range (IQR) test was explored. The IQR is a measure of variability based on dividing a dataset into quartiles. Specifically, it uses the 25th and 75th interquartile range to establish the distribution of the irregular or the decomposed series. Detecting anomalies using IQR method requires setting a decision range, where any data point lying outside this range is considered as anomaly. In this project, the decision range of what is considered as normal behavior of the series was set to a factor of three (3) times above the 75th inter quartile and three (3) times below the 25th inter quartile range and any data point beyond the limits were considered as potential anomalies.

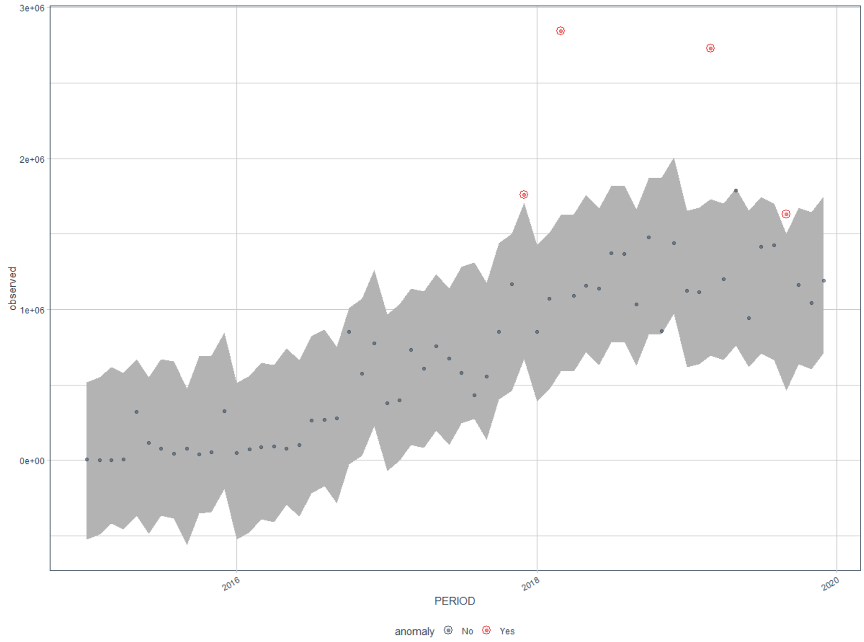


Figure 29 : *Sample illustration of potential anomaly tagging using IQR*

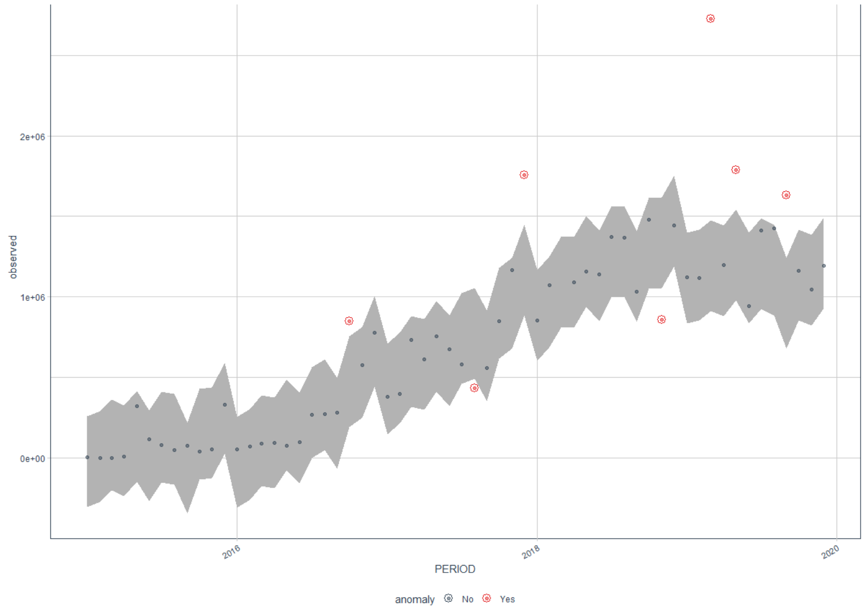


Figure 30 : *Sample illustration of potential anomaly tagging using GESD*

Since the objective of this component is the prioritization of items by detecting anomalies, the rules established using IQR was compared to the rules discovered applying the GESD. As illustrated, rules or the established normal value level of an item using GESD is more stringent than the IQR which has a wider range of limits. That is, it is easier for a value to be tagged as potential anomaly under the rules of GESD than the IQR. As GESD tends to be the better performing method in outlier removal (Rosner, 1983), potential anomalies tagged through the GESD method were selected.

4.2.4 Sample Results

Each item series was decomposed depending on the result of its seasonality test. After decomposition, residuals were checked as it is useful in determining if the decomposition has adequately captured the information in each item series. After which, each series was modelled according to its case, then tagged the potential anomalies, accordingly.

Cases	No. of Series
STATIC	12
S-STL-WHITE NOISE	7
NSDET-SPLINE-WHITE NOISE	29
NS-DIFFERENCING-WHITE NOISE	6
NS-ARIMA(0,0,0)	1
NS-ARIMA(0,0,1)	2
NS-ARIMA (0,1,0)	1
NS-ARIMA(0,1,1)	2
NS-ARIMA(0,2,0)	1
NS-ARIMA(0,3,1)	1

Table 13: *Component A: Overall model category of each item*

The table above shows the model used for each item series to cull out the remainder from the original series which was used in tagging potential anomalies. Note that the for the 12 item series found to be static, once its value is above/below its value = 0, for the past consecutive month, automatically, this series and all its records, will be tagged as potential anomaly. Meanwhile, for the item series included in the dataset with available priority tagging, below table shows the chosen model according to its case:

IDENTIFIED MODEL	INCLUDED
STATIC	S1ChB006
	S1ClA004
	S1PrA003
S-STL-WHITE NOISE	S1TrB009
NSDET-SPLINE-WHITE NOISE	S1FrA002
	S1FrA001
	S1PaB005
	S1PoB007
	S1OtB010
	S1OpB008

Table 14: *Component A: Model identified for the selected item series*

Static item series (S1ChB006, S1CIA004,S1PrA003) are records related to merchandise shipments, item series with NSDET-SPLINE type of model are related to freight payment, and operational related payments(port payment, lease, etc). S1TrB009 item series is related to transportation commission and fees. This is an interesting insight in business perspective as this may be used in further analyzing and determining its effect in the movement of trade statistics.

The remainder of all seasonal series decomposed using STL were found to be white noise. For the sample seasonal series above, months that were tagged as potential anomaly are as follows: 201610, 201708, 201712, 201803, 201811, 201903, 201905, 201909. Periods that have the most number of tags from the seasonal item series are 201505 and 201512 (both periods were tagged as anomaly for 3 item series). Transactions that fell on subject flagged items and months should be verified by the auditors.

For non-seasonal items, four (4) different case handlings were done to extract the remainder as white noise. Periods with the most number of potentially anomalous tags from the nonseasonal item series are the following: 201506 – with 13 items tagged as anomaly, 201505 – with 11 items tagged as anomaly, 201712 – with 10 items tagged as anomaly. As the objective of Component A is really to suggest a priority item for auditors, the results showed that instead of checking 62 items, for example in period 201712, auditors will check only ten (10) items that are tagged as priority giving them more time to analyze subject records and provide more in-depth insights on each record. Detailed monthly tag for all item series can be found in Appendix 1.

4.3 Component B: Identify anomalies per item based on company type

After framework rules are created on the accounts level in Component A, the behavior of the records per company type was considered. In assessing anomalies, an unsupervised machine learning model called Isolation Forest was used.

To discuss the results for Component B, recall the following Process Overview.

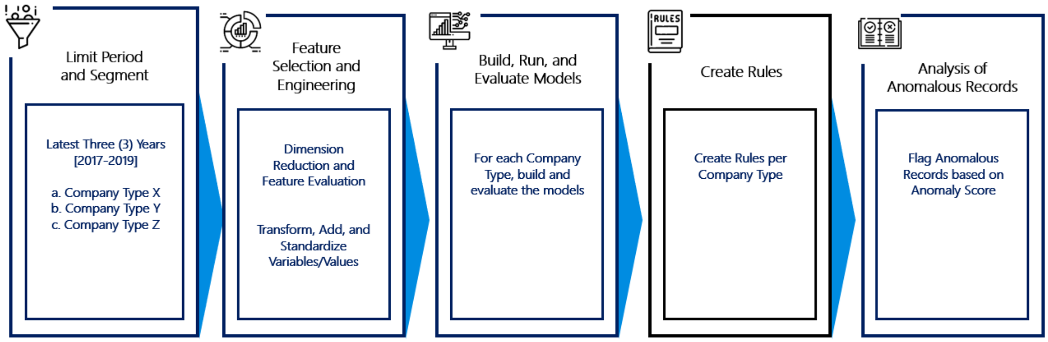


Figure 35: *Process Overview of Component B series*

Figure 37: *New Engineered Features*

4.3.1 Build, Run and Evaluate Model

i. Model Diagram

Using iterations from the engineered features and varying model parameters on Contamination Rate and Number of Trees, the best model per Company Type was evaluated. To have a baseline model, dataset with original features and default parameters were used in identifying potential anomalous records per company type. After which, effect of using different model parameters in the original dataset was inspected. The said process was repeated using the dataset with engineered features. The steps in model building for Component B is detailed below:

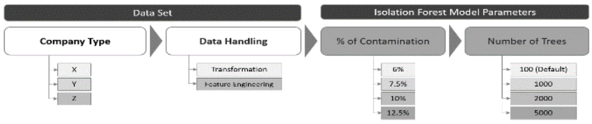


Figure 39: *Model Building flow per company type*

ii. Model Metrics

In order to evaluate each model iterations, a confusion matrix was utilized.

		Actual Labelled Data	
		Not Priority	*Priority*
Isolation Forest Label	*Not Priority*	TRUE Not Priority	FALSE Not Priority
	Priority	FALSE Priority	TRUE Priority

In the confusion matrix, there are four (4) cases where records can fall into:

- TRUE Priority (TP): The model labeled the record as Priority, and it is actually a Priority
- TRUE Not Priority (TNP): The model labeled the record as Not Priority, and is actually Not Priority
- FALSE Priority (FP): The model labeled the record as Priority, but it is actually Not Priority
- FALSE Not Priority (FNP): The model labeled the record as Not Priority, but it is actually Priority

In analyzing the classification of records of the model, the following metrics was used from the confusion matrix**:**

Metrics	Formula	Definition	Answers
Accuracy	$= \frac{TP + TNP}{TP + TNP + FP + FNP}$	represents the ratio of correctly classified records over total records	How many records did we correctly classify out of all records?
Recall (Sensitivity)	$= \frac{TP}{TP + FNP}$	represents the ratio of the correctly classified priority records over actual priority records	Of all the records that are priority, how many did we correctly predict?
Precision (or Positive Predictive Value)	$= \frac{TP}{TP + FP}$	represents the ratio of the correctly classified priority records over all labeled priority records	How many of those we classified as priority are actually priorities
F1 Score	$= 2 * \frac{Recall * Precision}{Recall + Precision}$	Helps measure both recall and precision by using weighted average	This shows the balance between precision and recall (low score mean poor precision and recall)
Specificity	$= \frac{TNP}{TNP + FP}$	represents the ratio of the correctly classified not priority records to all	Of all the records that are not priority, how many did we correctly predict?

Table 17: *Confusion Matrix Definition*

Although accuracy is a good measure and the most intuitive one, it is only best for symmetric datasets wherein our FNP and FP are almost close. Based on the results, this is not the case for the available dataset in this project. Thus, additional metrics will be used.

Precision signifies how certain the model provides a True Priority result while recall indicates how much "Priority" records were not missed. Recall is used if having False Priority is more acceptable than having False Not Priority, while Precision is used if a True Priority is more of the concern. Meanwhile, a high F1 Score indicates that the model provided a good mix of recall and precision. Lastly, Specificity is chosen if the intention is to cover all True Not Priority.

For this project, Recall and Precision is emphasized, which will give the auditors a good level of certainty that (1) the model is tagging actual anomalous records correctly, and (2) the model is not missing actual anomalous records. These are two of the intended results for having the framework, which will contribute in making the auditing process more efficient.

iii. Model Building

In discussing how the models were developed, this section will begin with the baseline model for Company Type X. Only the AMOUNT and COMPCODE were used for the initial modelling, and the results are shown below:

Looking at the results of the iteration for the Company Type X data for different variable combinations and contamination rates, the Isolation Forest produced varying results. It can be observed that (1) the Amount and COMPOCODE variables are enough to provide relatively good Accuracy vis-a-vis other models, and (2) as the contamination rate increases, the recall rate also increase, however, the increase negatively impacts the accuracy and precision of the model.

Using the results of iteration from Company Type X, it was observed that out of the eight iterations, the optimal contamination rate is 11%, as highlighted in iteration viii above. This rate is near to actual rate of anomalous records that is being experienced in the business domain by the auditors (at 10%).

Now, although the accuracy of the model is relatively good at 86.6%, the Recall and Precision are low at 32.5% and 20.2%, respectively. This result suggests that additional parameter tuning might be necessary.

Provided these low results, adding more in the number of trees parameter was explored. In the below results, it can be observed that there were no improvements in the metrics when the number of trees increased, thus the use of the default number of trees of 100 was

COMPANY TYPE X				
Data Set				
Variables	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding
No. of Data Columns	17	17	17	17
Model Iteration	viii FINAL TYPE X MODEL	ix	x	xi
Isolation Forest Parameters				
Contamination Rate	11%	10%	10%	10%
No. of Trees (Default)	100	1000	2000	5000
Evaluation Metrics				
No. of Records Tagged as Anomaly:	5168	4707	4703	4706
Accuracy	86.6%	87.1%	87.1%	87.0%
Recall	32.5%	28.6%	28.3%	28.1%
Precision	20.2%	19.5%	19.3%	19.2%
Specificity	90.6%	91.4%	91.4%	91.3%
F1 Score	24.9%	23.2%	23.0%	22.8%

Table 19: *Company Type X Final Model*

Given the results of our metrics from the different iterations, the it was concluded concluded that the model with the baseline features (AMOUNT and COMPCODE) at 11% contamination and at 100 trees is the most optimal as well as ideal for Company Type X model.

But, despite establishing a good accuracy level the Recall, Precision, and F1 Score remained to be low after multiple model iterations. Below is the summary of interpretation for the first Component B model:

- **Accuracy**: At good level of 86.6%, this tells us that our Company Type X model is intuitively classifying the True Priority and True Not Priority records correctly over all records
- **Recall**: At low level of 32.5%, this tells us that our Company Type X model is classifying a lot of False Priority records
- **Precision**: At low level of 20.2%, this tells us that our Company Type X model is not precise enough to classify the True Priority records
- **Specificity**: At high level of 90.6%, this tells us that our Company Type X model can properly specify the True Not Priority
- **F1 Score**: At low level of 24.9%, this further validate the poor level of recall and precision

Given that our framework is to improve the auditing process, we emphasize the importance of having a good Recall and Precision. The researcher highly recommends further improvements of the Company Type X Model by resolving the following limitations:

- Acquire more original and numeric variables from the ITRS Record and external figures for better feature selection for model building
- To increase Recall, introduce more features and validate potentially increasing the contamination rate by acquiring more labelled data and align it with the current experienced anomaly rates by the auditors
- To increase Precision and F1 score, explore other possible iterations in the variables and model parameters using a super computer

Furthermore, the researchers still propose to utilize the Component B Company Type X model for the initial operationalization of our framework. This will be accounted by the overall nature of our consolidated models and the final priority ranking with Component A and Component C scores.

Meanwhile, above process was repeated for company type Y and Z. Inspecting the evaluation metrics, below are the results and final models:

Table 20: *Company Type Y Model Iterations*

COMPANY TYPE Y									
Data Set									
Variables	-Amount -Company Code with One-Hot Encoding	- Date Interval - Amount ALL CATEGORICAL with One-Hot Encoding: - BookCD - ItemCode - CompCode - LocCode Excl PARTY1 and 2	- Date Interval - Amount - Company Code with One-Hot Encoding - Engineered Means and Delta - Engineered Date Variables	- Date Interval - Amount - Company Code with One-Hot Encoding - Engineered Means and Delta - Engineered Date Variables	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding
No. of Data Columns	17	179	22	22	17	17	17	17	17
Model Iteration	i	ii	iii	iv	v	vi	vii	viii FINAL TYPE Z MODEL	ix
Isolation Forest Parameters									
Contamination Rate	10%	10%	10%	15.0%	6%	7.5%	12.5%	15.0%	20.0%
No. of Trees (Default)	100	100	100	100	100	100	100	100	100
Evaluation Metrics									
No. of Records Tagged as Anomaly:	619	619	619	928	371	464	771	928	1231
Accuracy	87.5%	86.4%	87.7%	83.2%	90.2%	89.2%	85.5%	83.6%	79.1%
Recall	28.8%	20.5%	30.7%	34.8%	18.3%	22.9%	33.2%	38.5%	41.5%
Precision	17.3%	12.3%	18.4%	13.9%	18.3%	18.3%	16.0%	15.4%	12.5%
Specficity	91.2%	90.7%	91.3%	86.3%	94.8%	93.5%	91.2%	86.5%	81.5%
F1 Score	21.6%	15.4%	23.0%	19.9%	18.3%	20.4%	21.5%	22.0%	19.2%

Based on the summary metrics for Company Type Y it was concluded that the model with the baseline features (AMOUNT and COMPCODE) at 15% contamination and at 100 trees is the most optimal as well as ideal model for this type.

However, it also fell to the same challenges with Company Type X, for having high accuracy, but low Recall and Precision. It is suggested that the same recommendation as Company Type X to further improve Type Y. Similarly, it was proposed to utilize the Company Type Y model for the initial operationalization of our framework.Lastly, below are the results for Company Type Z model building:

COMPANY TYPE Z								
Data Set								
Variables	-Amount -Company Code with One-Hot Encoding	- Date Interval - Amount ALL CATEGORICAL with One-Hot Encoding: - BookCD - ItemCode - CompCode - LocCode Excl PARTY1 and 2	- Date Interval - Amount - Company Code with One-Hot Encoding - Engineered Means and Delta - Engineered Date Variables	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	-Amount -Company Code with One-Hot Encoding	- Date Interval - Amount - Company Code with One-Hot Encoding - Engineered Means and Delta - Engineered Date Variables
No. of Data Columns	17	179	22	17	17	17	17	17
Model Iteration	i	ii	iii	iv	v	vi FINAL TYPE Z MODEL	vii	viii
Isolation Forest Parameters								
Contamination Rate	10%	10%	10%	6%	7.5%	12.5%	15.0%	12.5%
No. of Trees (Default)	100	100	100	100	100	100	100	100
Evaluation Metrics								
No. of Records Tagged as Anomaly:	7	7	7	4	5	9	10	9
Accuracy	92.424%	83.3%	92.424%	90.9%	89.4%	92.4%	90.9%	92.4%
Recall	66.667%	16.7%	66.667%	33.3%	33.3%	83.3%	83.3%	83.3%
Precision	57.143%	14.3%	57.143%	50.0%	40.0%	55.6%	50.0%	55.6%
Specificity	95.000%	90.0%	95.000%	96.7%	95.0%	93.3%	91.7%	93.3%
F1 Score	61.538%	15.4%	61.538%	40.0%	36.4%	66.7%	62.5%	66.7%

Table 21: *Company Type Z Model Iterations*

Given the results of our metrics for Company Type Z, it was concluded that the model with the baseline features (AMOUNT and COMPCODE) at 12.5% contamination and at 100 trees is the most optimal as well as ideal model for this type:

- **Accuracy**: At high level of 92.4%, this tells us that our Company Type Z model is intuitively classifying the True Priority and True Not Priority records correctly over all records
- **Recall**: At good level of 83.3%, this tells us that our Company Type Z model classify low False Not Priority records
- **Precision**: At good level of 55.6%, this tells us that our Company Type Z model is precise enough to classify the True Priority records
- **Specificity**: At high level of 93.3%, this tells us that our Company Type Z model can properly specify the True Not Priority
- **F1 Score**: At low level of 66.7%, validating a relatively good mix of recall and precision

4.3.2 Create Rules

Given these findings, below is the summary of the proposed rules for Component B:

Company Type	Variables	Dimension Reduction	Contamination Rate	Number of Trees
If record under Type X	Baseline Features		11%	
Y	Baseline Features	No	15%	100
Z	Baseline Features		12.5%	

Table 22: *Proposed Rules for Component B*

Figure 44: *Company Type Z – Average amount of transaction for potentially anomalous transaction per company*

4.4 Component C: Identify anomalies per item based on customer behavior

In parallel of the analysis of company type level in Component B, the behavior of the customers was analyzed. To identify anomalies, Clustering techniques such as Density-based spatial clustering of applications with noise (DBSCAN) and Clustering Large Applications based on RANdomized Search (CLARANS) will be used. R Software using R Studio (*v 1.2.5033*) was used for the analysis.

To discuss the results for Component C, recall the following Process Overview.

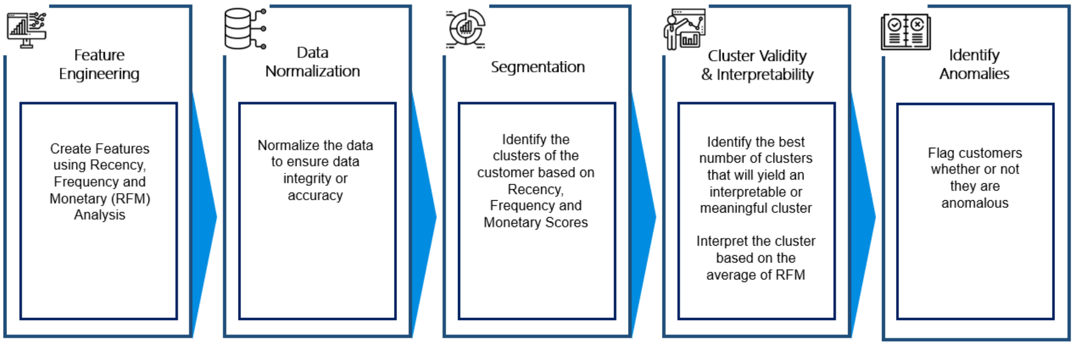


Figure 10: *Process Overview of Component C*

4.4.1 Feature Engineering Using Recency, Frequency and Monetary (RFM) Analysis

i. **Calculate Recency, Frequency and Monetary values for every customer**

Based from the definition of Recency (R), Frequency (F) and Monetary (M), the data was processed accordingly

Recency (R) : difference between the analysis date and the most recent date, that the customer has transacted. The analysis date here has been taken as the maximum date available for the variable REFDTE.

Frequency (F) : Number of transactions performed by every customer (PARTY1).

Monetary (M) : Total money spent by every customer (PARTY1)

To determine the RFM value of the customer (PARTY1) using the RFM value, the symbol the following symbols was coined

Symbol	Description
_Up	is a value higher than the average value
_Down'	is a value lower than the average value

Table 23: *Symbols for RFM Score*

This means that the higher the value it will be better for the company and the lower of the average it will get worse for the company. But for Recency (R), the symbol _Down means the lower of the average then the better for the company and the symbol _Up means higher than average then the value is not good for the company.

4.4.2 Data Normalization

Normalization of data aims to manage data between one attribute to another attribute does not have a great distance. This study needs to be normalized because the data, Recency (R), Frequency (F) are very different from Monetary (M). M is the amount of money issued by customers. Data that has been identified as RFM will be normalized by using min-max method using R Software. Min-Max Normalization was chosen as the method of data normalization because it preserves the relationships among the original data values thus it guarantees that all the features will have the exact same scale (guaranteed to reshape the features to be between 0 and 1) compared to Z-score Normalization which is also helpful in the normalization of the data but not with the exact same scale (normalized values can have different ranges). It is also easier to compute compare to Z-Score Normalization. It can be useful in algorithms that do not assume any distribution of the data like K-Nearest Neighbors which is vital in determining the best number of clusters.

4.4.3 Segmentation

To understand the behavior of the customers, there is a need to segment it using the RFM Scores that were derived and define it using the symbols ***_Up*** and ***_Down*** per quantities – Recency, Frequency and Monetary

Segments	Description	Behavior
R_Down F_Up M_Up or R_Down F_Up M_Down	Frequent Customer	This customer group is customers who has recently made a transaction with a high number of transactions and the amount of money spent is either high or low
R_Up F_Up M_Up or R_Up F_Up M_Down or R_Up F_Down M_Down or R_Up F_Down M_Up	Inactive or Lost Customer	Group of customers who have not made a purchase with the number of transactions and the total money spent is either higher or lower than the average in the past.
R_Down F_Down M_Down or R_Down F_Down M_Up	New Customer	This customer group is the customer has just made a transaction with a low number of transactions and the money is either low or high

Table 24: *Customer Segmentation using RFM Scores*

4.4.4 Cluster Validity & Interpretability

Now that the data is normalized and behavior of the customers was determined, the clustering technique was performed to identify the potential anomalies. There were two (2) clustering technique that was performed namely Density-based spatial clustering of applications with noise (DBSCAN) and Clustering Large Applications based on RANdomized Search (CLARANS)

i. Clustering Large Applications based on RANdomized Search (CLARANS)

Clustering Large Applications based on RANdomized Search (CLARANS) was run in R Software for the same normalized dataset. To run it, there is a need to determine the optimal value of k, or the number of clusters. To derive the value of k, an elbow method was used by running the *fviz_nbclust()* function in R software. The results were as follows:

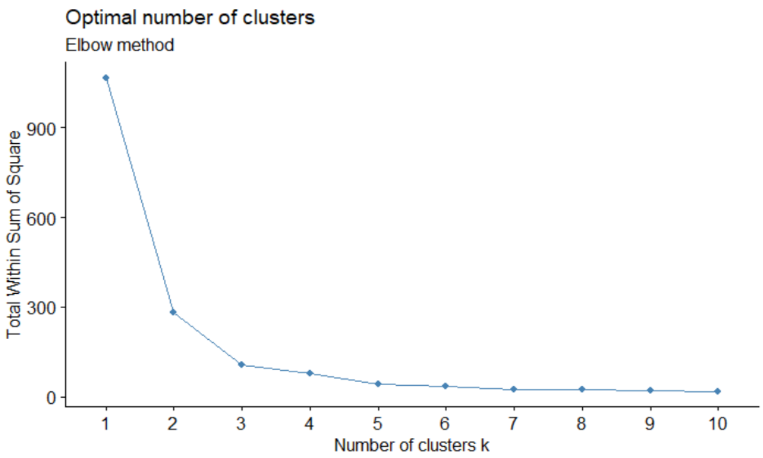


Figure 54: *Elbow Method*

Based from the graph the optimal number of k = 2. This k was used in running CLARANS. To run CLARANS, *clara()* function was performed in R by setting by setting pamLike = FALSE and samples = n = 12001, the sample size.

```
Call:    clara(x = rfm_df_norm, k = 2, samples = 12001, pamLike = FALSE)
Medoids:
       Recency   Frequency     Monetary
[1,] 0.0718232 0.004573171 0.0001923414
[2,] 0.5856354 0.001524390 0.0001850288
Objective function:  0.1238343
Clustering vector:   int [1:12001] 1 2 1 1 2 2 2 2 1 2 1 2 1 1 2 2 1 1 ...
Cluster sizes:           5651 6350
Best sample:
 [1]   419   648   843   857   920  1813  2098  2103  2227  2449  2567  2625  2740  2897  3360  3403  3534  3953  3988  4088
4239  5914
[23]  5967  6099  6621  6892  7011  7178  7182  7785  7854  8097  8219  8246  8408  8480  8650 10197 10940 11619 11727 11833
11841 11882

Available components:
 [1] "sample"     "medoids"    "i.med"      "clustering" "objective"  "clusinfo"   "diss"       "call"       "silinfo"    "d
ata"
```

Figure 55: *CLARANS Result*

To validate the cluster, Dunn's Index was performed in R Software using *cluster.stats()* function

```
# Dun index
CLARANS_stats$dunn

[1] 0.004257649
```

Figure 56: *Dunn's Index of CLARANS*

A higher Dunn Index will indicate compact, well-separated clusters, while a lower index will indicate less compact or less well-separated clusters so for this case since our Dunn's Index is 0.00425, it means that the cluster generated is less compact or not well-separated from other clusters therefore it is poor clustering. Given that a poor clustering was concluded, another clustering technique were performed for comparison which is the Density-based spatial clustering of applications with noise (DBSCAN)

ii. Density-based spatial clustering of applications with noise (DBSCAN)

For DBSCAN, the parameters ε and *minPts* are required and must be specified and determined. Ideally, the value of ε is given by the problem to solve (e.g. a physical distance), and minPts is then the desired minimum cluster size.

As a rule of thumb, a minimum minPts can be derived from the number of dimensions D in the data set, as minPts ≥ D + 1. The low value of minPts = 1 does not make sense, as then every point on its own will already be a cluster. With minPts ≤ 2, the result will be the same as of hierarchical clustering with the single link metric, with the dendrogram cut at height ε. Therefore, minPts must be chosen at least three (3). However, larger values are usually better for data sets with noise and will yield more significant clusters. Also, minPts = 2·dim can be used, but it may be necessary to choose larger values for very large data, for noisy data or for data that contains many duplicates. In this project, minPts = 2*dim = 2*3 dimensions = 6 was used since there are three (3) dimensions namely Recency, Frequency and Monetary.

On the other hand, the value for ε can then be chosen by using a k-distance graph, plotting the distance to the k = minPts-1 nearest neighbor ordered from the largest to the smallest value. Good values of ε are where this plot shows an "elbow". if ε is chosen much too small, a large part of the data was not be clustered; whereas for a too high value of ε, clusters will merge, and the majority of objects will be in the same cluster. In general, small values of ε are preferable, and as a rule of thumb only a small fraction of points should be within this distance of each other.

The function *kNNdistplot()* using R Software was used to draw the k-distance plot. The aim is to determine the "knee", which corresponds to the optimal eps parameter. This knee corresponds to a threshold where a sharp change occurs along the k-distance curve.

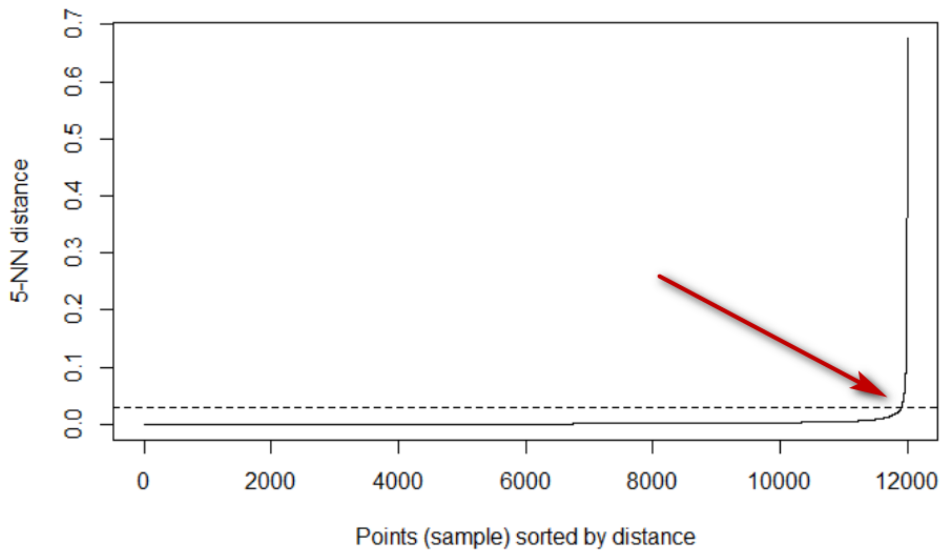


Figure 57: *K-NN distance plot*

It can be seen that the optimal eps value is around a distance of 0.03.

Once MinPts and optimal eps value were determined, DBSCAN was performed using *dbscan()* function in R software.

```
DBSCAN clustering for 12001 objects.
Parameters: eps = 0.03, minPts = 6
The clustering contains 1 cluster(s) and 69 noise points.

   0     1
  69 11932

Available fields: cluster, eps, minPts
```

Figure 58: *DBSCAN Results*

4.4.5 Identify Anomalies

i. Density-based spatial clustering of applications with noise (DBSCAN)

From the DBSCAN Results, it already explicitly shows that there are 69 noise points, these noise points were considered as potentially anomalous. To further verify, cluster plot is needed to have a visualization how far these points. *fviz_cluster()* function was used

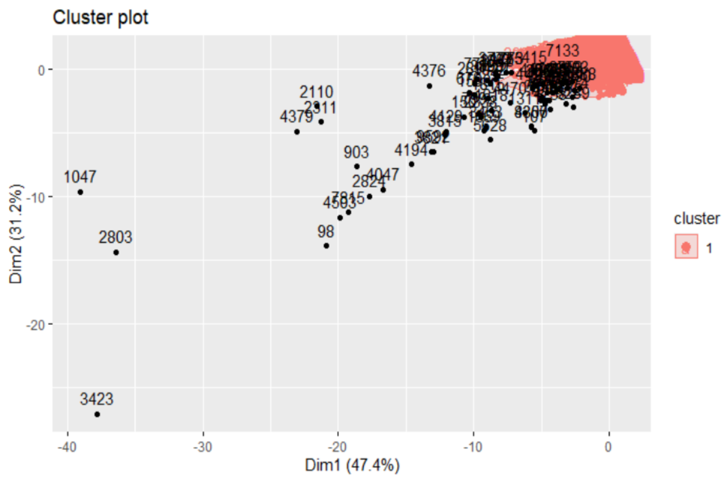


Figure 64: *Cluster Plot of DBSCAN*

The black points on the cluster plot corresponds to the noise points. These noise points were extracted from the dataset to tag it as potentially anomalous customers.

```
Frequent Customers Inactive or Lost Customers                New Customers
                53                          12                           4
```

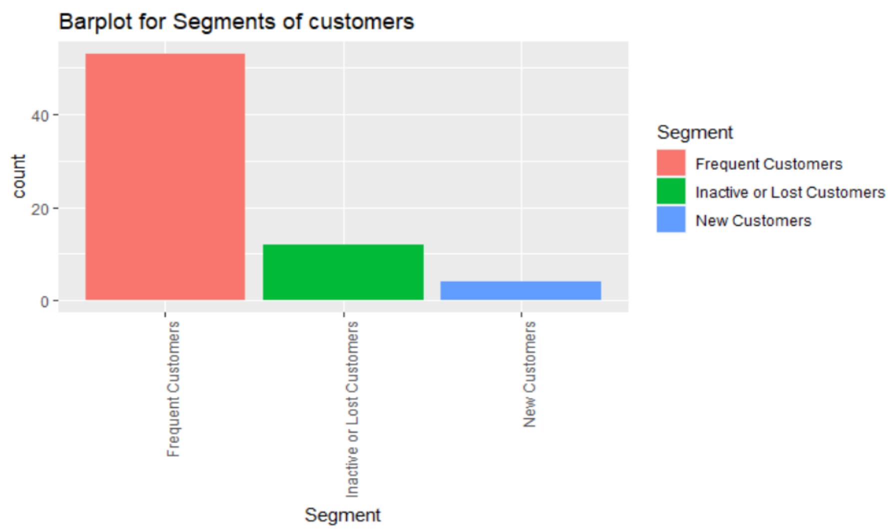


Figure 66: *Frequency Plot of Noise Points per Customer Segment*

To further analyze the behavior of these noise points, the difference of cluster means from overall means was computed

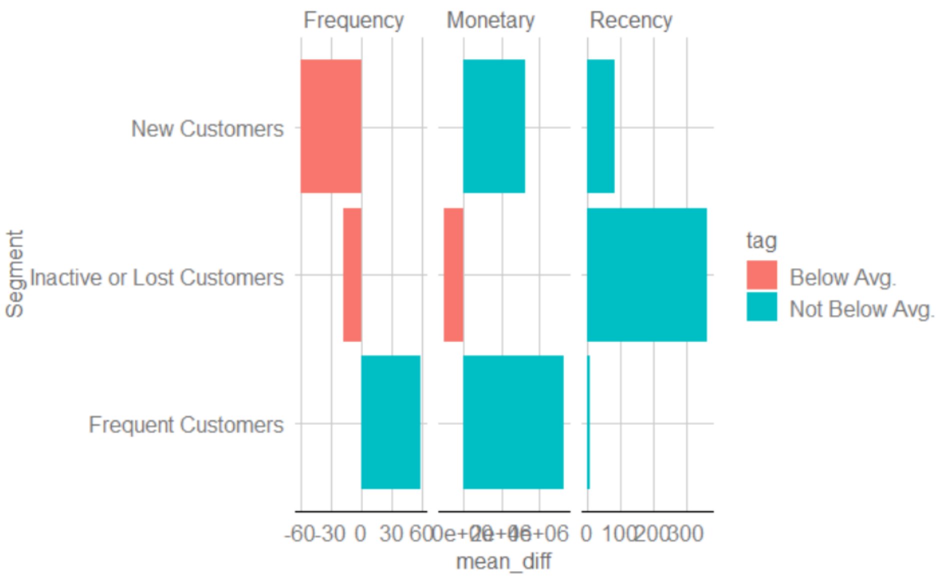


Figure 67: *Mean Difference Plot of Noise Points*

Based from the Mean Difference Plot, it can be concluded that for New Customers has low frequency of records and above average on the amount they spent on their records. For Inactive or Lost Customers, their frequency of records and the amount they spent is low. Lastly, For Frequent Customers, they have a high frequency and amount in their records.

Now that the behavior of the potentially anomalous customers was determined, tagging them as Priority in the dataset was done.

REFDTE	AMOUNT	BOO…	REMA…	ITEMCODE	LOCC…	COMPC…	PARTY2	PARTY1	Component C
<dbl>	<dbl>	<dbl>	<chr>	<chr>	<chr>	<chr>	<dbl>	<dbl>	<chr>
20191016	35877.02	1	P	S1ChB006	C0006	U0010	1000317830	1000734810	Priority
20191016	21267.58	1	P	S1ChB006	C0006	U0010	1000317830	1000784707	Priority
20190206	174960.00	2	NP	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190208	199960.00	2	NP	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190220	162458.00	2	NP	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190222	76883.00	2	NP	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190114	199960.00	2	P	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190107	199960.00	2	P	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190116	94960.00	2	P	S1ChB006	C0003	U0005	1000161375	1000287519	Priority
20190125	219960.00	2	P	S1ChB006	C0003	U0005	1000161375	1000287519	Priority

Figure 68: *Priority Tagging for Component C*

Based from the results of DBSCAN and CLARANS, DBSCAN is an easier cluster technique to identify potential anomalies given that it is more sensitive to outliers. Potential anomalies can also be easily tagged which answers the objective of this study.

4.5 Priority Ranking and Filtering

In this study, identifying the priority level of a record for audit was based on three (3) components as proposed in the outlined framework. Component A investigated the items' historical behavior, Component B evaluated the company type, while Company C explored the specific customer behavior. Each component has delved into a record using different methodologies having the same objective of the framework, that is to evaluate a record in more than one aspect. Different rules of what is normal were established, and records which deviate from this were tagged as priority Table 25 presents a summary of the components.

	Component A	Component B	Component C
Perspective	Item's own historical behavior	Company Type behavior	Customer Behavior
Feature Selection	n/a	PCA	RFM Analysis
Algorithms	STL, Moving Average, Cubic Spline Smoothing, ARMA/ARIMA	Isolation forest	DBSCAN, CLARANS
Priority rule	GESD: Outside of 95% of the critical value	At 10 percent contamination	Noise Points

Table 25: *Summary of each component*

Since each component investigated different aspect that influences a record, scoring of a record was done equally. While results of each component was produced independently, results were designed to be read as one output, each complementing the other.

		Actual Tag		
		Priority (P)	Not Priority (NP)	Total
IPA Tag	High	91	128	219
	Medium	544	4064	4608
	Low	2965	45537	48502

Table 26: *Results of Priority Rating using the IPA Framework*

High Priority Records

Based on above table, there are 219 records tagged as High Priority. Records with High priority rating implies that the record reported was unexpected on three different perspectives of evaluation: the total item value deviated from its known historical behavior; the company reported the record failed to follow its usual movement in relation with same company type; and the customer who owns the record acted differently from its usual behavior. Records with high priority need immediate attention and suggested to be sent directly to the banks for confirmation or eventual correction of record. In this way, any significant deviation can be efficiently identified, and eventual explanation will be secured.

Medium Priority Records

There are 4,608 records that were rated as Medium Priority. These records require auditor to scrutinize and assess if a confirmation or correction is still necessary. With the availability of priority rating, auditors will have enough time to investigate records requiring attention.

Low Priority Records

Records rated as Low priority can be considered as a normal record in terms of the three angles used for its evaluation, but this does not limit the auditors to audit the said transactions.

The second dataset provided has a priority tag for each record. Note that the **Not Priority (NP)** tag for this dataset does not necessarily mean that a record is not a priority. There are cases that the record was not audited due to the volume of records being handled by the auditor and was eventually tagged as NP. Furthermore, records tagged as **Priority (P)** does not necessarily mean that the record is erroneous. Subject dataset was used to evaluate the result of the entire framework. For the purpose of evaluation, once a record is tagged as a priority in one component, its final tag will be Priority.

		Predicted Tag		
		Priority (P)	**Not Priority (NP)**	Total
Actual Tag	**Priority (P)**	1734	1866	3600
	Not Priority (NP)	21190	28539	49729
	Total	22924	30405	53329

Table 27: *Confusion Matrix of the results*

The confusion matrix above shows the actual tag versus the resulted tag of a record when the framework was used in evaluating the priority level of a record. For the auditors, this table shows that there are more records tagged as priority by the framework than the ones manually tagged by the auditors, that is 22,924 records tagged as Priority compared to just 3,600 originally tagged. Though type 1 errors (or any type of error) is best to be avoided, in this case, this implies that the framework broadens the scope of records being audited which aids the job of an auditor in the initial screening of records.

4.6 Model and Framework Evaluation

Computing the metrics to evaluate the framework, the following are the results:

Metric	Percentage
Sensitivity or Recall or True Positive Rate	48.2%
Specificity, Selectivity or True Negative Rate	57.4%
Precision or Positive Predictive Value	8%
Accuracy	57%

F-score	13%

Table 28: *Metrics used to evaluate the results*

Sensit d by the framework. In other words, a highly sensitive test is one that correctly confirms the true nature of the subject. Meanwhile, Specificity or Selectivity refers to True Negative Rate. Precision refers to the positive predictive values which signifies the probability that the test will produce true positive.

```
Confusion Matrix and Statistics

          Reference
Prediction    NP     P
        NP 28539  1866
        P  21190  1734

               Accuracy : 0.5677
                 95% CI : (0.5634, 0.5719)
    No Information Rate : 0.9325
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0159

 Mcnemar's Test P-Value : <2e-16

              Precision : 0.07564
                 Recall : 0.48167
                     F1 : 0.13075
             Prevalence : 0.06751
         Detection Rate : 0.03252
   Detection Prevalence : 0.42986
      Balanced Accuracy : 0.52778

       'Positive' Class : P
```

Figure 69: *Confusion Matrix for Evaluation of Framework*

In evaluating the framework, it is important to note that the available labeled dataset has a limitation to what was manually covered by the audit. In table 28, recall, which is at 48 percent, and precision which is at 7 percent alone, seems low, but since specificity is already above 50 percent and given the limitation of the available dataset that we have, the performance of the framework is a good benchmark compared to the manual process. This is supported by the Confusion Matrix above, which shows that at No information Rate, the performance of the framework is at 93 percent already. In addition to this, to evaluate if the Framework really brought improvement, McNemar’s test was used. McNemar’s test is being used to determine if there was a statistically significant difference in the proportion of items with priority rating before and after the implementation of the framework.

Conclusion: At $\alpha = 0.05$ level of significance, there is a sufficient statistical evidence to conclude that that the proportion of records tagged as priority under the proposed framework is significantly different from that of a “no information framework”. Specifically, it is observed that a total of 22,924 records were tagged as priority under the proposed framework while 3,600 records were tagged as priority under a “no information framework”. This implies that the proposed framework is better than the no information framework.

5 CONCLUSION

The proposed Intelligent Prioritization of Account (IPA) framework offers a solution that can augment the current manual audit process. The IPA framework ensured that all records will be part of the scope of audit and will be checked accordingly. Adopting the IPA framework supported the objective of the auditors to have a more holistic view of the record through its evaluation in different perspectives that support each other. Deeper analysis and more informed action recommendations can be focused on based from data discovery.

In summary, the implementation of the framework is seen to provide the following benefits for the BSP:

- will increase process efficiency through the immediate prioritization of records, earlier launch of investigation (as needed);
- will strengthen data quality through the addition of procedures that validates against other aspects not currently routinely considered;
- will enhance audit process through a more statistically based validation process;
- will empower decision making through the insights presented in the different sections of this study and the proof-of-concept offered by the study in the application of artificial intelligence in central banking.

6 RECOMMENDATIONS

Based on the experience of this study on handling BSP data, the following recommendations are presented:

1. **Maintain a central archive of ITRS records pre- and post- audit.**
   - The availability of labeled data set was found to be a limitation of the study. It is recommended that all records under ITRS be labeled and collected in a central archive. Having a labeled dataset allowed for supervised algorithms to learn from these data.

2. **Standardize entries in report (i.e., customer names)**
   - In addition to labeled dataset, during the exploration and cleanup of data, it was noted that data quality, specifically customer names, are not standard. Ensuring quality of each data entry increases the accuracy of any algorithm to learn the data.

3. **Explore using different parameter levels used in this study**
   - In component A for example, all parameters adopted were based on the default setting in R Software. Exploring different level parameters may provide different results

4. **Periodic retraining of the algorithms used in the study**
   - Periodic retraining might be needed to redefine historical behavior, capture changes in regulations and incorporate pattern evolution

5. **Framework Expansion or Model Improvement**
   - The proposed framework is currently limited to the available fields in the ITRS report. External factors can be explored depending to the item being investigated, for example foreign exchange rates, inflation rate, Gross Domestic Product, oil prices, etc. Identification of external factors should be done with subject matter expert to ensure appropriate factors will be included in model development.
   - Models are currently based on nominal amount of a record. Future work to complement the current framework may be done using models build on a different measure. Also, models in component A are based on month on month changes. An exploration in the day to day fluctuations may also complement the current framework.

6. **Model expansion to cover all items in the ITRS Report**
   - Due to data availability, current models are limited to a number of items preselected in the study. Expanding the models to incorporate all items in the ITRS report might provide different insight and might open for new opportunities.

Furthermore, below are the list of recommendations specific per component.

Component	Recommendation
Component A: Identify anomalies per item based on historical behavior using time series decomposition	Explore using different seasonal decomposition technique Use different detrending method in decomposing non-seasonal series Conduct parameter tuning in GESD and IQR
Component B: Identify anomalies per item based on company type	Higher processor specifications (e.g. super computer) to run larger data with more avenue for parameter tuning (e.g. increasing number of trees) and feature engineering exploration Further deep dive on the tagged anomalous records, to qualify accuracy and effectiveness of Component B anomaly detection using the built Isolation Forest models Acquire more labeled and updated data to train the iForest models Explore other feature importance methods and variable transformation techniques to properly select variables and improve model
Component C: Identify anomalies per item based on customer behavior	Use DBSCAN for clustering instead of CLARANS given that it is the more viable technique and is more sensitive to outliers Conduct further RFM runs to derive new segments for analysis and model improvements

Table 29: *Recommendations per Component*

In terms of operationalizing the Intelligent Prioritization Account (IPA) Framework, the following strategy can be proposed to ITRS

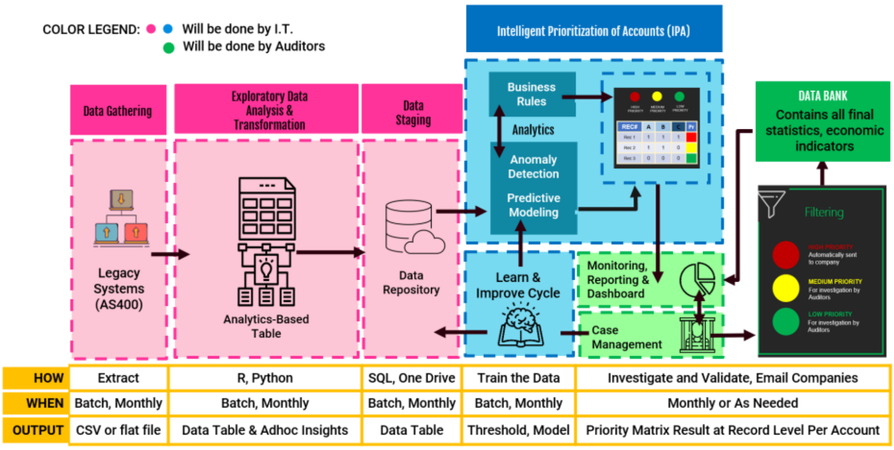


Figure 70: *Recommendations for Operationalizing the Framework*

The IT Team of BSP will do the Data Gathering on a monthly basis and conduct Exploratory Data Analysis and Transformation on it and stored it in a cloud or secured server. They will also do the model building, running and evaluation and tagging of priorities that will yield a Priority Matrix Result to be sent to the auditors for their investigation. Recalibration on the tuning parameters can be done on quarterly basis in cooperation with the business experts and auditors.

To fully operationalize the IPA in the aspect of people and technology, the following items must be considered but not limited to:

People	Job Title	Core Skill Set
IT	Infrastructure Admin Support	Administers and manages infrastructure systems including but not limited to: threat detection; Servers, Cloud etc
	Project Manager	Planning the activities Organizing a project team to perform work Delegating the teams Controlling time management Managing deliverables Monitor progress
	Application Developer	builds application that interact with data and implements data models
Data Team	Data Engineer	Collect, structure and analyze data builds and maintains a scalable data infrastructure to make relevant data available to teams

	Data Scientist	gets deep into the data to draw hidden insights and influence business decisions Develop statistical models and algorithms
	Business Analyst	approaches data from different angles and applies analytics models to confidently support or dispel assumptions answer business questions with data Identify gaps, pain points and opportunities for growth
	Data Governor	to manage and control the ever-growing amount of data in order to improve business outcomes
Auditor	Auditor	User of IPA Framework Monitors the records Performs Case Management of tagged anomalies

Technology	Requirement	Purpose
Hardware (Infrastructure) - Application Server - Database Server	Cloud or On-premise	Acts as a central archive or database of records
Audit Management System - Software Licenses - Subscription	Software as a Service (SaaS) Licenses for Software access	For Case Management of tagged anomalous records
Dashboard and Reporting Tool	Software as a Service (SaaS) Licenses for Software access	For rendering real-time Dashboards and reports
Others	Digital Certificate Web and Internet Security	For protection against Cybersecurity attacks

Table 30: *Roles and Technology in the Recommended Operationalization of the*

The figure below is a sample product that can be designed as the User Interface of the IPA Framework to be used by auditors so they can have a real-time dashboard or view of the number of records tagged as priority and can already act on it.

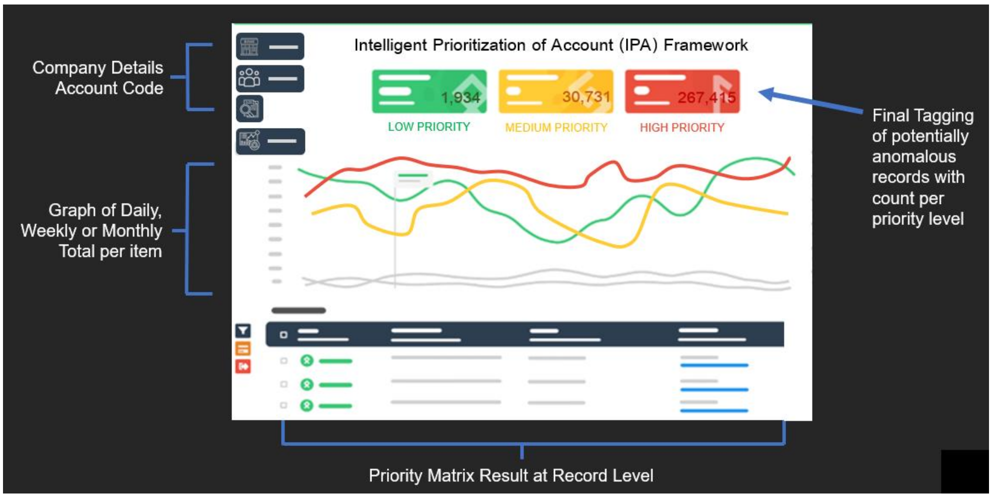


Figure 71: *Sample Product of IPA Framework*

7 REFERENCES

1. Abdi, H., & Williams, L. J. (2010). Principal Component Analysis. Wiley Interdisciplinary Reviews: Computational Statistics, 2: 433-459. https://doi.org/10.1002/wics.101
2. Ahmed M., Mahmood A.N. (2013). A novel approach for outlier detection and clustering improvement. in: 8th IEEE Conference on Industrial Electronics and Applications, ICIEA, pp. 577–582.
3. Anti-Money Laundering Council (2017). A risk assessment on the Philippines exposure to external threats based on submitted suspicious transaction reports. http://www.amlc.gov.ph/images/PDFs/AMLC%20EXTERNAL%20THREATS%20STUDY.pdf
4. Asian Development Bank (2019). Strenghening Anti-Money Laundering and combating the Financing of terrorism in the Philippines . https://www.adb.org/news/videos/strengthening-anti-money-laundering-and-combating-financing-terrorism-philippines
5. Baesens, B., Van Vlasselaer, V., & Verbeke, W. (2015). Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection. . Wiley and SAS Business Series
6. Bangko Sentral ng Pilipinas (2020). Manual of Foreign Exchange Transactions (MORFXT). https://www.bsp.gov.ph/Regulations/MORFXT/MORFXT.pdf
7. Bank for International Settlements (2019). The use of big data analytics and artificial intelligence in central banking. IFC Bulletin Number 50, Proceedings of the IFC - Bank Indonesia International Workshop and Seminar on Big Data in Bali, 23-26 July 2018.
8. Blazquez-Garcia,A., Conde,A., Mori,U., & Lozano, J. (2021). A Review on Outlier/Anomaly Detection in Time Series Data. ACM Comput. Surv. 54, 3, Article 56, https://doi.org/10.1145/3444690
9. Casey, M. (2014). Emerging opportunities and challenges with central bank data. In Proceedings of seventh ECB statistics conference. https://www.ecb.europa.eu/events/pdf/conferences/141015/presentations/Emerging_opportunities_and_chalenges_with_Central_Bank_data-presentation.pdf?6074ecbc2e58152dd41a9543b1442849
10. Chakraborty, C. & Joseph, A. (2017). Machine Learning at Central Bank. Bank of England Working Paper No. 674, Available at SSRN: https://ssrn.com/abstract=3031796
11. Chandola, V., Banerjee, A. & Kumar, Vipin (2009). Anomaly Detection: A Survey. ACM Comput. Surv. 41, 3,Article 15 (July 2009), 58 pages. DOI = 10.1145/1541880.1541882
12. Chaundy, T., & Bullard, J. (1960). John Smith's Problem. The Mathematical Gazette, 44(350), 253-260. doi:10.2307/3614890
13. Cleveland, R. B., Cleveland, W. S., McRae, J. E. & Terpenning, I. (1990). STL: A Seasonal-Trend Decomposition Procedure Based on Loess (with Discussion). Journal of Official Statistics, 6, 3--73.
14. Cohn,T. A. , England,J. F. , Berenbrock, C., Mason,R., Stedinger,J., & Lamontagne, J. (2013). A generalized Grubbs-Beck test statistic for detecting multiple potentially influential low outliers in flood series. Water Resources Research, vol. 49, no. 8, pp. 5047–5058.
15. Cullinan, G.J. (1977). Picking Them by Their Batting Averages: Recency-Frequency-Monetary Method of Controlling Circulation. New York: Direct Mail/Marketing Association.
16. Diday, E., & Simon, J.C. (1976). Clustering analysis. In Digital Pattern Recognition. K. S. Fu, Ed. Springer-Verlag, Secaucus, NJ, 47–94.
17. Dodge, Y. (2008). InterQuartile Range. The Concise Encyclopedia of Statistics.. https://doi.org/10.1007/978-0-387-32833-1_200
18. Espenilla, N. (2018). Central Bank Evolution in the Digital Age. A speech, https://www.bis.org/review/r180814j.htm

19. Ester M., Peter Kriegel H., Sander J., Xu X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. AAAI Press, 1996, pp. 226–231
20. Financial Stability Board (2017). Artificial intelligence and machine learning in financial services: Market developments and financial stability implications. http://www. fsb .org/wpcontent/uploads/P011117.pdf.
21. Gao, J., Song, X., Wen, Q., Wang, P., Sun, L., & Xu, H. (2020). RobustTAD: Robust Time Series Anomaly Detection via Decomposition and Convolutional Neural Networks.. ArXiv, abs/2002.09545.
22. Goldstein, M. & Uchida, S. (2016). A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. . PloS one. 11. e0152173. 10.1371/journal.pone.0152173.
23. Greis,R., Ries,T., & Nguyen, C. (2018). Comparing Prediction Methods in Anomaly Detection: An Industrial Evaluation . MiLeTS ’18, August 2018, London, United Kingdom
24. Guo,Y., Xu,Q.,Sun,S.,Luo,X., & Sbert, M. (2016). Selecting video key frames based on relative entropy and the extreme studentized deviate test. Entropy, vol. 18, no. 3, p. 73.
25. Gurung, N. & Perlman, L. (2018). Use of Regtech by Central Banks and Its impact on Financial Inclusion. Evidence from India, Mexico,Nigeria, Nepal and Philippines. . http://dx.doi.org/10.2139/ssrn.3285985
26. Heinze, E. (2018). Anomaly detection techniques and their applicability in univariate time series. Seminar Paper, Traveltainment GmBH
27. Hung, P. & Dat, D. (2020). Customer Behavior Clustering Based on Balance History Using Dynamic Time Warping Distance. International Journal of Machine Learning and Computing. 10. 87-92. 10.18178/ijmlc.2020.10.1.903.
28. Hyndman, R.J., King, M.L., Pitrun, I., & Billah, B. (2005). Local linear forecasts using cubic smoothing splines. Australian and New Zealand Journal of Statistics, 47(1), 87-99.
29. Hyndman, R.J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2
30. Irving Fisher Committee on Central Bank Statistics (2020). Towards Monitoring financial innovation in central bank statistics. IFC Committee Report number 12
31. Jain, A. K., & Dubes, R. C. (1988). Algorithms for Clustering Data.. Upper Saddle River, NJ Prentice-Hall, Inc.
32. Kapoor, K. (2020). A Novel Algorithm for Optimized Real Time Anomaly Detection in Timeseries. arXiv preprint arXiv:2006.04071.
33. Kargari, M., & Eshghi, A (2018). A Model Based on Clustering and Association Rules for Detection of Fraud in Banking Records. 10.11159/mvml18.104.
34. Kasunic, M., McCurley, J., Goldenson, D., & Zubrow, D. (2011). An Investigation of Techniques for Detecting Data Anomalies in Earned Value Management Data Software Engineering Measurement and Analysis (SEMA). Carnegie Mellon University. Report. https://doi.org/10.1184/R1/6571940.v1
35. Larik,A. & Haider, S. (2011). Clustering based Anomalous Transaction Reporting. Procedia Computer Science, Volume 3, 2011, Pages 606-610, ISSN1877-0509, https://doi.org/10.1016/j.procs.2010.12.101.
36. Rosnerd, B. (1983). Percentage Points for a Generalized ESD Many-Outlier Procedure. Technometrics, Vol. 25, No. 2, May 1983, pp. 165-172.
37. Lenderink ,R. J. (2019). Unsupervised outlier detection in financial statement audits. Master’s thesis, University of Twente.
38. Liu, F., Ting, K.M. & Zhou, Zh (2008). Isolation Forest. In ICDM ’08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining. IEEE Computer Society

39. Liu, Q. (2019). An Application of Exploratory Data Analysis in Auditing – Credit Card Retention Case. Emerald Publishing Limited, Bingley, pp. 3-15. https://doi.org/10.1108/978-1-78743-085-320191001
40. Moschini, G., Houssou, R., Bovay, J., & Robert-Nicoud, S. (2020). Anomaly and Fraud Detection in Credit Card Records Using the ARIMA Model.. ArXiv, abs/2009.07578
41. Murugavel, P., & Punithavalli, Dr. (2011). Improved Hybrid Clustering and Distance-based Technique for Outlier Removal. International Journal of Computer Science and Engineering (IJCSE), Vol. 3, No. 1, 2011, pp. 333-339.
42. Ounacer, S. ,El bour, H., Oubrahim, Y., Ghoumari, M.Y., & Azzouzazi, M. (2018). Using Isolation Forest in anomaly detection : the case of credit card transactions. Periodicals of Engineering and Natural Sciences Vol.6, No. 2, p394-400
43. Perperoglou, A., Sauerbrei, W., Abrahamowicz, M. et al. (2019). A review of spline function procedures in R. BMC Medical Research Methodology, https://doi.org/10.1186/s12874-019-0666-3
44. Shipmon, D.T., Gurevitch, J.M., Piselli, P., & Edwards, S.T. (2017). Time Series Anomaly Detection; Detection of anomalous drops with limited features and sparse examples in noisy highly periodic data. Google Inc. , ArXiv, abs/1708.03665
45. Siti, Monalisa (2018). Analysis Outlier Data on RFM and LRFM Models to Determining Customer Loyalty with DBSCAN Algorithm. International Symposium on Advanced Intelligent Informatics (SAIN), 2018, pp. 1-5, doi: 10.1109/SAIN.2018.8673380.
46. Vallis,O. ,Hochenbaum, J. , & Kejariwal, A. (2014). A novel technique for long-term anomaly detection in the cloud.. Proceedings of the 6th USENIX conference on Hot Topics in Cloud Computing (HotCloud'14). USENIX Association, USA, 15.
47. Vendramin L, Campello RJGB, , Hruschka ER. (2009). On the Comparison of Relative Clustering Validity Criteria. In: Apte C, Park H, Wang K, Zaki MJ, editors. Proceedings of the 2009 SIAM International Conference on Data Mining. Philadelphia, PA. Society for Industrial and Applied Mathematics. 2009: 733–44.
48. Webel, K. & Ollech, D. (2018). An overall seasonality test based on recursive feature elimination in conditional random forests. Proceedings of the 5th International Conference on Time Series and Forecasting, pp. 20-31.
49. Hastie, Trevor & Tibshirani, Robert & Friedman, Jerome & Franklin, James. (2004). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Math. Intell.. 27. 83-85. 10.1007/BF02985802.
50. Yang,Y., Lian,B., Li,L., Chen, C., & Li, P. (2014). DBSCAN Clustering Algorithm Applied to Identify Suspicious Financial Transactions. International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2014, pp. 60-65, doi: 10.1109/CyberC.2014.89.
51. Zhang, C., Cao, B., Li, T. (2010). Fuzzy Information and Engineering Volume 2. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg. pp. 797.

8 LIST OF TABLES

9 List of Figures