National College of Ireland

Detection of Non-Contemporaneous Activity in an Electronic System Using Unsupervised Machine Learning

MSc Research Project
Data Analytics

Chris Miller
Student ID: x20166788

School of Computing
National College of Ireland

Supervisor: Mohammed Hasanuzzaman

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Chris Miller
Student ID:	x20166788
Programme:	Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Mohammed Hasanuzzaman
Submission Due Date:	01/02/2023
Project Title:	Detection of Non-Contemporaneous Activity in an Electronic System Using Unsupervised Machine Learning
Word Count:	7346
Page Count:	23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	1st February 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Detection of Non-Contemporaneous Activity in an Electronic System Using Unsupervised Machine Learning

Chris Miller
x20166788

Abstract

Some industries, such as the pharmaceutical industry require their employees to record manufacturing related activities in real-time as imposed by the global regulatory agencies. Deficient electronic system design means that these international requirements can be violated, therefore questioning the integrity of the manufactured products. An external solution is therefore required to identify these non-contemporaneous and therefore anomalous activities.

The objective of this research project is therefore to build an information and communications technology (ICT) solution that can be used to identify these potentially anomalous activities. Synthetic datasets that mimic this use case are initially created as there are no real-world datasets available. These datasets are then used as inputs for the K-means clustering (KM), Isolation Forest (IF), Restricted Boltzmann Machines (RBM), and Adaptive Resonance Theory (ART) unsupervised methods, and the statistical methods of the interquartile range (IQR) and Z-score (ZS), to identify potential anomalous activities. The developed models are evaluated, and the best-performing method is integrated into the final ICT solution.

Based on the evaluation of the experiments completed in this research project the RBM method was integrated successfully into the final ICT solution due to its consistent performance and ability to be fine-tuned by the anomaly investigators.

1 Introduction

The ALCOA+ framework (Kopp; 2019) has been adopted by the pharmaceutical industry to ensure the data integrity of batch-related information. The ALCOA+ framework stipulates that the associated records must be Attributable, Legible, Contemporaneous, Original, Accurate, complete, consistent, enduring, and available. This research project focuses on the contemporaneous requirement, whereby batch-related records should be documented in real-time.

Adhering to international regulations is a critical activity for companies operating in the pharmaceutical space, and breaches of these regulations can come at a high cost, both financial and reputational in nature (McDowall, 2020). They also state that the Federal Drug Agency (FDA)[1] is losing patience with the pharmaceutical industry due to a failure to take these regulations seriously.

[1] `https://www.fda.gov/`

The number of noncompliance findings regarding non-contemporaneous activity is also increasing with the Medicines and Healthcare products Regulatory Agency (MHRA)[2] making 141 citations in 2018 (Unger; 2020). This trend is also noted by Kopp (2019) citing that deficient system design means that there are gaps when it comes to compliance to regulations as the electronic systems have not been designed to meet regulatory requirements. Schniepp (2019) also support the increase in citations stating that the European Medicines Agency (EMA)[3], World Health Organization (WHO)[4] and other regulatory agencies are also registering an increase in citations globally. The need for an information and communications technology (ICT) solution to automatically detect these anomalous activities is further highlighted by Kopp (2019), as the need for controls using technical solutions is noted.

To meet the contemporaneous regulatory requirement, operators and lab technicians in the pharmaceutical industry must record data into the various electronic systems, used to support manufacturing, in real-time. These electronic systems include Manufacturing Execution Systems (MES), Lab Control Systems (LCS), and Learning Management Systems (LMS). Entering the data in real time will ensure that the products are manufactured and tested in line with the product specifications.

The following research question is therefore raised by this research problem:

- *Can an ICT solution be developed using unsupervised machine learning, and visualization techniques to detect potential non-contemporaneous activities in electronic systems, to provide industry, regulators, and consumers assurances that critical batch manufacturing activities have been recorded in real-time?*
- *Can this solution be developed and deployed in an ethical manner?*
- *Can this solution be evaluated using standard methods?*

To answer this research question, the first objective, and major contribution of this research project, will be to complete in-depth experiments using the unsupervised machine learning methods of K-means clustering (KM), Isolation Forest (IF), Restricted Boltzmann Machines (RBM), and Adaptive Resonance Theory (ART). These methods have been selected as anomaly detection will be used to detect the non-contemporaneous activities. Anomaly detection has primarily been used in intrusion and financial fraud detection systems so this research project will be the first to use anomaly detection to detect non-contemporaneous activities in a pharmaceutical electronic system.

The next step and objective will be to evaluate these methods, and the best-performing model will be integrated into a novel information and communications technology (ICT) solution. This ICT solution will be used to ensure activities recorded while manufacturing pharmaceutical products will be completed in a contemporaneous manner benefiting the regulators, industry, and product consumers. This ICT solution will include a graphical user interface (GUI) and PowerBI report which will allow the anomaly investigators to complete further analysis of the identified potential anomalies.

The final objective, and minor contribution of this research project, will be to create a synthetic dataset to mimic the recording of operators' and technicians' activity in a

[2] `https://www.gov.uk/government/organisations/medicines-and-healthcare-products-regulatory-agency`

[3] `https://www.ema.europa.eu/en`

[4] `https://www.who.int/`

Manufacturing Execution System (MES) due to the absence of publicly available real-world data.

In terms of limitations, the absence of real-world datasets related to Manufacturing Execution Systems presents a challenge for this research project, this limitation, however, has been mitigated by the creation of three synthetic datasets, one simple, and two complex datasets. The approach of allowing the system user to enter a complex query into the ICT solution, which can then be used to retrieve the required data from the source system, also provides additional flexibility for this ICT solution. It has been assumed that this complex query can also retrieve the start and end times or duration of each activity or transaction that needs to be analysed to determine if the activity is indeed normal or anomalous.

Another challenge presented by this research project was the trend that most unsupervised learning methods generally used labeled data in order to train the associated models, the challenge was therefore to develop models and solutions that did not use labeled data.

The ART model was very challenging due to the absence of an ART2 distribution in order to complete the experiments. The execution time for this model was also protracted which resulted in a subset of the data being used for experimental purposes.

The delivery of this ICT solution will benefit numerous key stakeholders including the pharmaceutical industry, regulators, and most importantly the consumer of the manufactured products. The pharmaceutical industry will be able to identify non-contemporaneous activities in near or real-time depending on the size of the dataset being queried and the selected frequency of extraction and analysis. The tracking, reporting, and review of these anomalous records will also have time and cost savings for the business when completed using the proposed ICT solution when compared to completing this activity manually. The implementation of this ICT solution will also mitigate the requirement to build safeguards into the electronic systems to prevent non-contemporaneous activity which would be expensive, timely, and complicated to implement. The result from the ICT solution can then be shared with the regulators in a pro-active manner as opposed to the regulators finding the anomalous activities, therefore, raising doubts as to the efficacy of the manufactured products. The end users of these products will also benefit as the products they consume will have been manufactured in line with the original specifications from a recording of data in a real-time perspective.

Section 2 “Related Work”, will critically review and analyse related works in anomaly detection, and related works in the area of synthetic datasets. Next, section 3 “Research methodology”, will describe the research methodology process flow that was followed, the materials and equipment used in the experiments and building of the ICT solution, and an overview of the synthetic datasets that were created and used in this project. Section 4 “Design specification”, describes the system design that was followed to deliver the ICT solution. Next, section 5 “Ethics”, will describe the ethical concerns raised by the project. Section 6 “Implementation” will describe how the selected models were implemented including how the models were developed. Next, section 7 “Evaluation”, will provide an analysis of the results from each model and dataset. Section 8, “Deployment” describes the deployment of the selected model into the final ICT solution. And finally, section 9, “Conclusion and Future Work” will critically summarise and assess the problem, research question, solution, and benefits of this research project and will also consider potential future work pertaining to this research project.

2 Related Work

This section of the project report will be structured with the following sub-sections. Sub-section 2.1 "Anomaly detection", will introduce and critically assess other works completed in this area. Next, sub-section 2.2 "The need for synthetic data", will describe the requirement for synthetic data when real world data is not readily available. This section will then close with sub-section 2.3 "Related work conclusion", which will summarise the literature review and provide further justification for this research project.

2.1 Anomaly detection

Anomaly detection is used in the financial, information technology (IT), and manufacturing sectors to detect financial fraud, security intrusions, and manufacturing defects (Alla and Adari; 2019). Unsupervised anomaly detection is used where there is an absence of labeled data, and in cases when you don't know the pattern of the fraudulent user behavior (Parisi; 2019), both of which are applicable to this research project. Conversely, both supervised and semi-supervised learning require labeled data to be available for model training, with the former requiring labeled data for model testing as well (Alla and Adari; 2019). Unfortunately, labeling data can be a time-consuming task (Lazarevic and Kumar; 2005) and is impractical when it comes to large dynamic data sets where the variables can change over time. For example, if a new product is added to the manufacturing product portfolio, or if an unseen cyber-security attack is launched against a network. This is why unsupervised learning stands out as the best method for this research project, whereby thresholds are used to detect anomalies as opposed to labels.

According to Gong et al. (2022), IQR, also known as the boxplot method, can also be used to identify outliers in a univariate dataset. The ZS method is based on the attributes of normal distribution (Wicklin; 2013), and uses the mean and standard deviation to detect outliers. By grouping the step durations by product and step combinations, a univariate dataset is created, which can therefore use IQR and ZS for outlier or anomaly detection.

A number of unsupervised methods exist to complete anomaly detection including IF (Alla and Adari; 2019; Bonaccorso; 2019; Zadafiya et al.; 2022; Zhong et al.; 2019), and RBM (Alla and Adari; 2019; Demertzis et al.; 2022; Do et al.; 2016); artificial neural networks (ANN) (Parisi; 2019); KM (Dunning and Friedman; 2014; Parisi; 2019; Yoseph and Heikkila; 2019; Kamra et al.; 2008); fuzzy c-means (FCM) clustering and local outlier factor (LOF) (Brahma et al.; 2020); and finally ART (Bielecki and Wójcik; 2021; Jones et al.; 2018; Liu et al.; 2015; Mejía-Lavalle; 2010).

IF is an unsupervised method for detecting anomalies that performs well on datasets with a high number of variables (Alla and Adari; 2019). Zadafiya et al. (2022) compared the performance of IF to LOF to detect fraudulent credit card transactions with IF providing superior performance. Zhong et al. (2019) used IF to monitor gas turbines health. A high level of accuracy was achieved using a small dataset and with unlabelled data, the latter being applicable to this research project. While the example detailed by Alla and Adari (2019) uses test and training datasets with labeled outcome variables, according to the scikit-learn documentation[5], and as illustrated by Zadafiya et al. (2022), the predict method can be used to return whether a record is an anomaly or not, returning

[5] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html`

1 for a normal record, and -1 for an anomalous record, therefore allowing the package to be executed in a fully unsupervised mode, that is, without having to train the model. Alla and Adari (2019) recommend that categorical data is encoded which is applicable to this research project. In terms of hyperparameter tuning, the n_estimators parameter can be used to set the number of trees in the forest, and the contamination parameter can be used to set the number of anomalies that exist in the dataset (Zadafiya et al.; 2022; Zhong et al.; 2019)

Demertzis et al. (2022) developed Automatic Differentiation Variational Inference (ADVI) RBM to complete anomaly detection of industrial infrastructure in real-time to identify security events. Do et al. (2016) developed Mixed-variate RBM to handle mixed data types using both real-world and synthetic datasets, the latter being used by this research project. They found that this new model competed well against state of the art solutions. Alla and Adari (2019) and Do et al. (2016) also recommend that numeric variable standardisation, and label encoding for categorical values take place to improve model performance, additionally Demertzis et al. (2022) normalised the data during their pre-processing stage. In terms of hyperparameter tuning, the learning_rate parameter can be tuned (Do et al.; 2016; Zadafiya et al.; 2022).

Yoseph and Heikkila (2019) used KM and Fuzzy C-means clustering techniques to identify anomalous queries in database systems with KM providing better performance. They used the cluster-based method to identify the anomolous records as the number of clusters wasn't known, for this research project the cluster distance-based approach will be used as the number of clusters is known. Kamra et al. (2008) used KM for role-based access control (RBAC) database intrusion detection using both real and synthetic databases with results that would suggest that their solution would work well in the real world. Hyperparameter tuning involves selecting the best value of k, which is unknown for most datasets. Conversely, the number of clusters, k, for the datasets created for this research project is equivalent to the number of product/step combinations.

ART was first introduced in the seminal paper by Grossberg (1976). Jones et al. (2018) used FuzzyArt, the combination of ART and fuzzy logic, to detect intrusions on a building control system, the source code for which is available on GitHub[6]. While this model was an unsupervised solution, it requires the data to be split into training and test datasets which does not meet the requirements of this research project. ART1 was used to create a new ART framework, ART-E, with an explanation feature to detect fraudulent activity in a financial database by Mejia-Lavalle (2010) and was selected due to its simplicity and performance. The unsupervised ART1 architecture was developed to cluster binary variables and has been implemented in the neural python library[7], this model is unsuitable for this research project as the dataset for this research project uses a continuous variable. The ART2 model, on the other hand, has been designed to accept continuous or analog input variables (Carpenter et al.; 1991). The vigilance parameter can be used to determine the reset threshold (Carpenter and Grossberg; 1987).

In terms of evaluation metrics, false positive rate (FPR) and false negative rate (FNR) were the most popular evaluation metrics used to evaluate model performance (Alla and Adari; 2019; Bielecki and Wójcik; 2021; Brahma et al.; 2020; Dunning and Friedman; 2014; Fadolalkarim et al.; 2020; Gaikwad and Thool; 2015; Jesus et al.; 2021; Kamra et al.; 2008; Parisi; 2019; Ronao and Cho; 2016; Vanerio and Casas; 2017). These metrics were also used by Zhong et al. (2019) whereby they were referred to as false and true

[6] `https://github.com/cbirkj/art-python/`

[7] `https://pypi.org/project/neural-python`

alarm rates. Brahma et al. (2020) state that a high TPR and low FPR are an indication of a performant model. Accuracy was also considered a popular evaluation metric by Alla and Adari (2019); Demertzis et al. (2022); Gaikwad and Thool (2015); Islam et al. (2015); Liu et al. (2015); Zadafiya et al. (2022); Zhong et al. (2019).

The ability to fine-tune anomaly detection thresholds was also considered an important feature, (Dunning and Friedman; 2014; Fadolalkarim et al.; 2020; Parisi; 2019) where the detection rate versus cost of detection needs to be taken into consideration. Such control will be implemented into the final ICT solution to allow the anomaly investigator to fine-tune the accuracy of the selected model.

2.2 The need for synthetic data

Synthetic data has been used by a number of industries especially the financial industry due to the sensitivity and lack of availability of real-world financial data, this sensitivity problem applies equally to the pharmaceutical industry upon which this research project is based. Use cases for synthetic data include fraud detection in the mobile payments industry (Gaber et al.; 2013) and anti-money laundering (Lopez-Rojas and Axelsson; 2012b) and (Lopez-Rojas and Axelsson; 2012a). Using synthetic data has its benefits including protecting sensitive company and customer data, making the data available to other researchers who are interested in the problem domain, and finally, different anomaly-based scenarios can also be investigated (Lopez-Rojas and Axelsson; 2012a). Lopez-Rojas and Axelsson (2012a) also states that the downside of using synthetic data is the fact that the data can be biased, or that the dataset does not reflect the real world. Ringberg et al. (2008) also argues that you need to simulate the normal and anomalous activity, and environment, to ensure that anomaly detectors are effective. They also state that to have an accurate probability of false positives and false negatives, you need a list of all anomalies to establish the ground truth, which is very difficult to achieve with real-world data due to the nature of the anomalous activity.

2.3 Conclusion

Based on the preceding literature review it is apparent that unsupervised anomaly detection is applicable to this research project and an overview of unsupervised anomaly detection has been provided. The need for synthetic data for this research project has been justified based on other works in anomaly detection where synthetic data was used. The next section will describe the implemented research methodology.

3 Research Methodology

3.1 Process overview

The CRoss Industry Standard Process for Data Mining (CRISP-DM) methodology was used as a baseline for the research methodology used during this research project (Chapman et al.; 2000) as outlined in Figure 1. The idea for the research project was initially generated. The next stage was to understand the research project definition which was to use unsupervised machine learning methods to identify anomalous activities in an electronic system, and to incorporate the best-performing model into an ICT solution. A

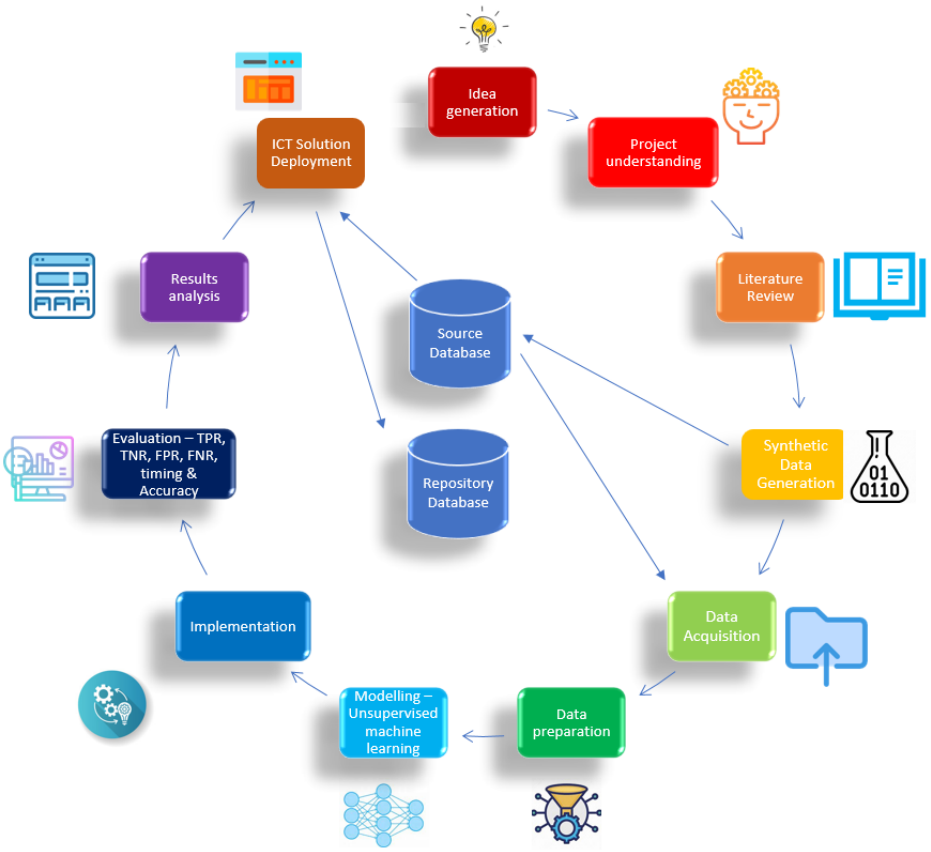

Figure 1: Research Methodology Process Flow

literature review was completed to build upon related works in this domain. Three synthetic data sets were used for this research project due to an absence of publicly available data. These synthetic data sets were generated using python[8] and stored in a database and comma-separated value (CSV) files for further processing. The data was then acquired from the source database using structured query language (SQL)(Badia; 2020). The next stage was then to understand and prepare the data for modeling including one hot encoding (OHE) and standardization. The statistical methods of IQR and ZS; and the unsupervised machine learning methods of KM, IF, RBM, and ART were then used to detect the anomalies in an unsupervised manner, that is, without labeled data. The models were then implemented and evaluated against a labeled data set using true positive rate (TPR), true negative rate (TNR), false positive rate (FPR), false negative rate (FNR), accuracy, and execution time. The best-performing model was then selected following the results analysis stage. And finally, the best-performing model was deployed by integrating within the final ICT solution which showcases the capability of the solution end to end.

3.2 Materials and equipment

A standard laptop running the Microsoft Windows 10 Pro 10.0.19044 Build 19044 operating system[9] was used to build and run this research project. The laptop used an

[8] `https://www.python.org/`
[9] `https://www.microsoft.com/en-gb/software-download/windows10`

Table 1: Table structure

Field name	Data type	Example
batch_number	Numeric	1234
product_name	String	Product A
step_number	Numeric	1
step_name	String	Step 1
username	String	User A
start_time	Datetime	01-Dec-2022 10:00:00
end_time	Datetime	01-Dec-2022 11:00:00

Intel(R) Core(TM) i5-3320M central processing unit (CPU) running at 2.60GHz with 16.0 GB random access memory (RAM), and a 240 GB solid state disk device (SSD). The primary programming language was Python[10] 3.8.12, and Jupyter[11] notebooks 6.4.3 running in Anaconda[12] navigator 2.1.1 was used as the primary development environment. PostgreSQL[13] 14 was used for the database layer and pgAdmin[14] was used for database management. Microsoft PowerBI desktop Version: 2.109.1021.064-bit (September 2022)[15] was used for the visualisation layer.

3.3 Dataset overview

The data for this research project was created using python as there is no publicly available data for this use case. The table structure can be seen in Table 1.

Three separate datasets were created for this research project with increasing degrees of complexity, referred to throughout this report as SIMPLE, COMPLEX1, and COMPLEX2. Anomalous user activity was injected into each of the 3 datasets, following a level of complexity in line with the complexity of each of the datasets. Sub-section 6.1 describes the process of creating these datasets.

4 Design Specification

The research project design consists of 3 layers as per Figure 2, namely, the presentation layer, business logic layer, and data layer. The presentation layer consists of a GUI and PowerBI dashboard which support user interaction with the system. The business logic layer includes, query design, system configuration, anomaly detection, and notification. And finally, the data layer includes data storage and retrieval.

The system users can use the GUI (1) to specify the data extraction query, source database details, repository database details, notification email address, and fine-tune model parameters. The data extraction query (2) will then be executed against the source database (5). The system configuration parameters (4) will be stored in the repository database. The extracted data will be processed by the anomaly detection system (ADS) (7), and stored in the repository database (6). The potential anomalies identified by the

[10] https://www.python.org/
[11] https://jupyter.org/
[12] https://www.anaconda.com/
[13] https://www.postgresql.org/
[14] https://www.pgadmin.org/
[15] https://powerbi.microsoft.com/

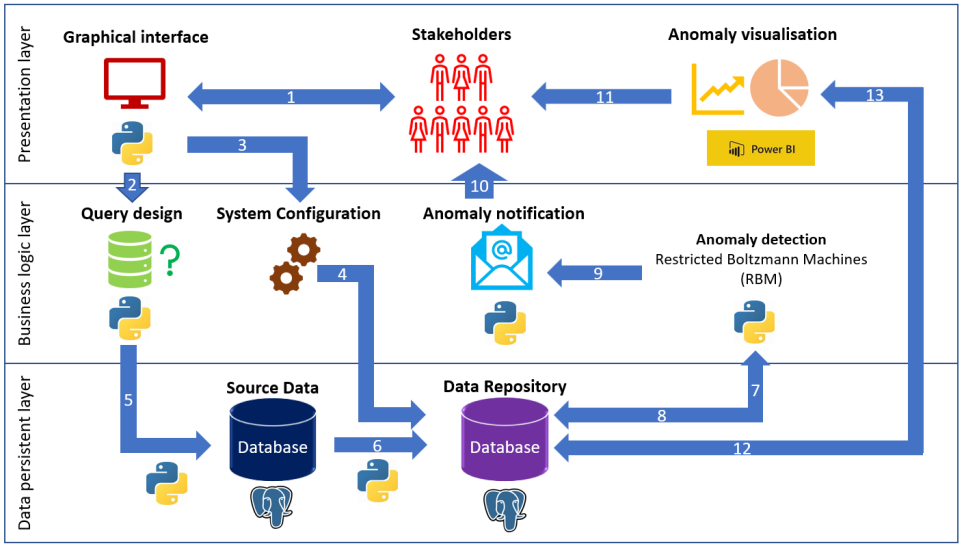

Figure 2: Design Specification Overview

ADS model, will be stored in the repository database (8). The anomalies will be circulated by the ADS to the configured email address in CSV format (9)(10). The system users can then use the PowerBI report to complete anomaly analysis (11). The PowerBI report retrieves data from the repository database (12) for self-service visualisation in the report (13).

5 Ethics

Figure 3 illustrates the 3 challenges, 4 themes, and 10 questions Saltz et al. (2019) proposes are reviewed when considering the ethical implications of a research project using machine learning. While there are no ethical concerns regarding this project in the research stage as synthetic data is being used, each of the questions will be reviewed in turn to determine the ethical impact of this research project if it was commercialised.

Regarding question 1, General Data Protection Regulation (GDPR)[16] could possibly apply to this solution if the user who recorded the anomalous activity can be determined from the assigned username stored in the database, although this will be contrary to security best practices which would use a system generated username. Question 2 raises the fact that users could be identified and subject to human resource (HR) processes if they have been identified as repeatedly breaching the contemporaneous requirement. Regarding question 3, it is envisaged that the legal rights of the company will not be impacted as the data would be used in a proactive manner to ensure product efficacy. Is it also envisioned that the rights of the individual will be protected by the company's HR processes. Question 4 will be addressed by ensuring that the username stored in the ADS will be hashed out to protect the identity of the user, therefore protecting personally identifiable information (PII) (Robinson; 2017). Regarding question 5, system owner approval would be needed in order to gain access to the operational databases and to approve the usage of the acquired data. Question 6 would be mitigated by ensuring that the system is validated for its intended use. Regarding question 7, bias has been minimized by injecting anomalies in a random fashion, and creating step durations that

[16] `https://gdpr-info.eu/`

Challenge	Theme	Questions
Oversight related challenges	Accountability & Responsibility	1. Which laws and regulations might be applicable to this project?
		2. How is ethical accountability being achieved?
Data Related Challenges	Data Privacy and Anonymity	3. How might the legal rights of organizations and individuals be impinged by our use of the data?
		4. How might an individuals' privacy and anonymity be impinged via aggregation and linking of the data?
	Data Availability and Validity	5. How do you know the data is ethically available for its intended use?
		6. How do you know the data valid for its intended use?
Model Related Challenges	Model and Modeler Bias	7. How have you identified and minimized any bias in the data or the model?
		8. How was any potential modeler bias identified, and then if apprioriate, mitigated?
	Model Transparency & Interpretation	9. How transparent does the model need to be and how is that transparency achieved?
		10. What are likely misinterpretations of the results and what can be done to prevent those misinterpretations?

Figure 3: Ethical considerations (Saltz et al.; 2019)

follow a normal distribution with random standard deviations. For question 8, modeler bias has been avoided by ensuring that all relevant variables have been retained for modeling purposes. Regarding question 9, the models selected for this research project are well-documented and known. Finally, question 10 will be mitigated by displaying a message on the GUI that the users of the ICT solution should use the source system to verify the reported anomalies before making any decisions.

6 Implementation

The following sub-sections describe the creation of the synthetic datasets, the implementation of the selected models, namely, KM, IF, RBM, and ART, and the statistical methods of IQR and ZS.

6.1 Synthetic dataset creation

Three synthetic datasets were created for this research project with varying levels of complexity, namely, SIMPLE, COMPLEX1, and COMPLEX2. Each of the datasets was created with the same structure as per Table 1.

For each dataset, a product configuration table, which was stored in CSV format, was used as a baseline for determining the median step duration. This CSV file had ten products, each with ten steps, and the associated step durations, which were selected arbitrarily. The username was incremented by 1 for each product.

The SIMPLE dataset was created by initially looping through 10,000 iterations of the product configuration data set. Anomalies were injected randomly, resulting in 999 anomalies being created. The anomalous duration was set to between 1 and 10 percent of the product/step duration, and was selected randomly using the randint()[17] function. The start and end times for each step were then calculated using this anomalous step duration. The resulting dataset was written to a CSV file, and to the SOURCETABLE table in the PostgreSQL source database.

[17] `https://numpy.org/doc/stable/reference/random/generated/numpy.random.randint.html`

Similarly, the COMPLEX1 dataset was created by looping through the product config CSV file contents for 10,000 iterations. The random randint() function was used to randomly select a standard deviation that was between 0.01 and 0.05 of the associated product/step duration. The random normal function was then used to create a normal distribution (Wicklin; 2013) of durations for each product/step combination, as the assumption is that these durations will follow a normal distribution. A Bernoulli distribution (Wicklin; 2013) was then used to inject 1 percent anomalous records into the dataset. The anomalous durations were randomly set at 1 to 50 percent of the minimum of each product/step duration. The username was incremented by 1 for each product and cycled between 1 and 10. The resulting dataset was written to a CSV file and to the SOURCETABLE_ND table in the PostgreSQL source database.

Finally, the COMPLEX2 dataset was created in a similar manner to the COMPLEX1 dataset, the difference being that the standard deviation was selected between 0.01 and 0.10 of the associated product/step duration, therefore resulting in potentially wider distributions. The anomalous durations were also randomly set at 1 to 90 percent of the minimum of each product/step duration, the intention, therefore, to make it more difficult for the anomaly detectors to distinguish between normal and anomalous records. The resulting dataset was written to a CSV file and to the SOURCETABLE_ND2 table in the PostgreSQL source database.

The anomalous records were labeled for evaluation purposes for all three datasets, but these labels were not saved to the resulting CSV file or PostgreSQL database table to ensure the unsupervised learning models could not use this labeled data. The batch numbers were also incremented by 1 for each change in product. The resulting durations for each dataset were then used to calculate the start and end times for each step using the datetime timedelta()[18] function, commencing with the current timestamp.

6.2 IQR and Z-score

The IQR is the difference between the first (Q1) and third quartiles (Q3). Outliers typically lie in the range Q1-1.5 and Q3+1.5. ZS of greater than 2.5 and less than -2.5 indicate outliers, although 3 is usually used as the cut-off point, meaning the outliers reside 3 standard deviations outside the mean value.

The following process was completed for each of the three generated synthetic datasets. Each of the datasets was initially read from CSV files as described in sub-section 6.1, and stored in a pandas[19] dataframe. The data was then grouped by product and step names, therefore, isolating the duration variable, and effectively creating a univariate dataset. The records which fell within the range Q1-1.5 were identified as being potentially anomalous. These anomalous records were then given a binary predicted label of 1 for anomalies and 0 for normal records and stored in another dataframe. The values in this dataframe had to be re-sorted and the index reset in order to be joined to the labeled anomaly dataset to determine the performance metrics for this method. The same process was then repeated for the ZS statistical method using the scipy stats zscore statistical function[20], with a ZS of 3 being used as the cut off point as specified above.

[18] `https://docs.python.org/3/library/datetime.html`

[19] `https://pandas.pydata.org/`

[20] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.zscore.html`

6.3 Data preparation

The following data preparation process was completed for each of the three generated synthetic datasets, and for each of the selected models. Each of the datasets was initially read from CSV files, as described in sub-section 6.1, and stored in a pandas dataframe. A new variable called prod_step was generated by concatenating the product_name and step_name fields together. The pandas get_dummies[21] function was then used to convert this categorical variable into dummy variables, therefore one hot encoding this variable. The duration variable was then standardised using the sklearn preprocessing minmax-scaler[22] class using the default feature length of 0 to 1. The unneeded variables were then dropped from the dataframe.

The enumeration technique[23] was also used by the IF, RBM, and ART models, whereby a new dataframe was created, with the prod_step variable and a new variable called prod_step_enum which was assigned a unique integer for each prod_step value. This dataframe was then joined to the duration variable using the pandas merge class[24]. This resulted in a dataframe with 2 variables being passed to the appropriate model.

A grouping technique was also used by the IF and RBM models, whereby the duration variable was isolated by grouping by product_name and step_name. The appropriate model was then run against each subset of data and the duration and predicted variables were saved to a new dataframe. The original index was retained to allow the new dataframe to be re-sorted by the index value, which would ultimately be used to join this dataframe to a dataframe containing the anomaly labels.

6.4 K-means

KM is an unsupervised algorithm that groups the input dataset into k clusters using the Euclidean distance from the cluster centre, which is called the centroid (Parisi; 2019; Yoseph and Heikkila; 2019). The formula for the KM objective function can be seen in Figure 4.

The KM model was built using the sklearn cluster KMeans class[25], specifying a k value, the number of clusters parameter, n_clusters, equal to the number of product/step combinations, and with the random_state parameter set to ensure repeatable centroid initialisation. The cluster_label, cluster_center, and calculated cluster_distance, Euclidean distance[26], attributes were then appended to the source dataframe. IQR and ZS methods were then used on the cluster_distance to determine whether the record was anomalous or not as described in sub-section 6.2, and label the record accordingly. The values in this dataframe had to be re-sorted and the index reset in order to be joined to the labeled anomaly dataset to determine the performance metrics for this method. The same process was then repeated for the ZS statistical method, with a ZS of 3 being used as the cut-off point. In instances where the default model didn't achieve an accuracy metric of 100 percent, the n_init hyperparameter was tuned using values in the range of 100, 200, and

[21] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.get_dummies.html

[22] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

[23] https://www.ga-ccri.com/outliers-and-categorical-data

[24] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.merge.html

[25] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

[26] https://towardsdev.com/outlier-detection-using-k-means-clustering-in-python-214188fc90e8

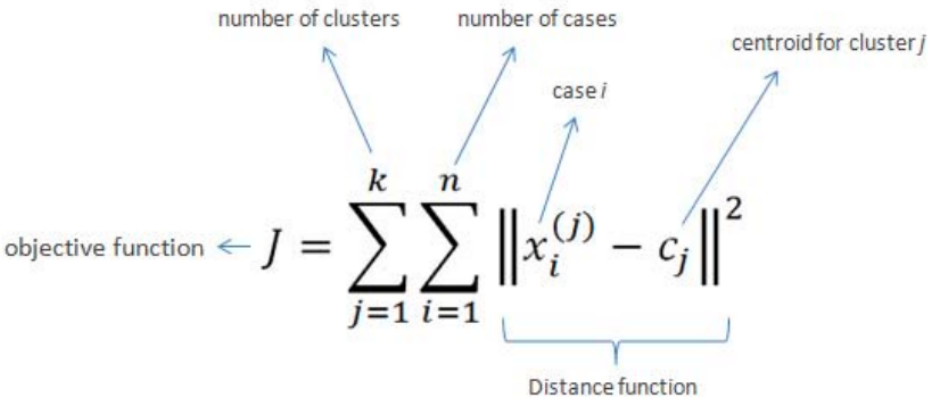

Figure 4: K-means - objective function (Yoseph and Heikkila; 2019)

300, to determine whether the performance of the model could be improved, as tuning this parameter avoids the issue of local minima.

6.5 Isolation Forest

IF continually partitions the dataset until an individual tree has been formed. Anomalies can be identified as those records that are closest to the root of the tree (Alla and Adari; 2019; Zhong et al.; 2019) as illustrated in Figure 5.

The IF model was built using the sklearn ensemble IsolationForest class[27]. The model was initially run using the default parameters, the random_state parameter was set to ensure repeatable selection of the feature and split values for each branching step and each tree in the forest. Subsequent runs specified a contamination parameter with a value equal to the number of anomalies injected into each dataset, and a max_features parameter set to the number of features in the dataset, that is, 101 features. The number of estimators in the model, n_estimators, was then tuned by specifying values in the range 1-30, and 40-100, in increments of 10, to determine which model could best detect the number of injected anomalies.

6.6 Restricted Boltzmann Machines

RBM is a two-layered unsupervised, stochastic generative deep learning model. The two layers consist of an input and hidden layer and it is similar to the first two layers of an artificial neural network (ANN). The restricted term comes from the fact that none of the nodes within a layer can be connected to each other. Each node also outputs a binary value, and the probability of each data point can be used to identify the anomalous records (Alla and Adari; 2019). An example of an RBM structure is illustrated in Figure 6.

The RBM model was built using the sklearn neural_network BernoulliRBM class[28]. The model was run specifying the number of binary hidden units, n_components, set to a value of 1, the random_state parameter was set to ensure repeatable generation of Gibbs

[27] `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html`

[28] `https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.BernoulliRBM.html`

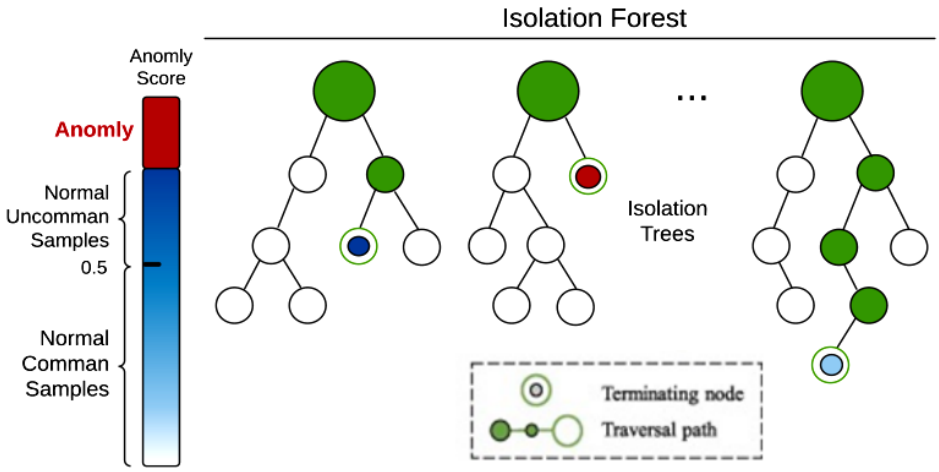

Figure 5: Isolation Forest overview (Zadafiya et al.; 2022)

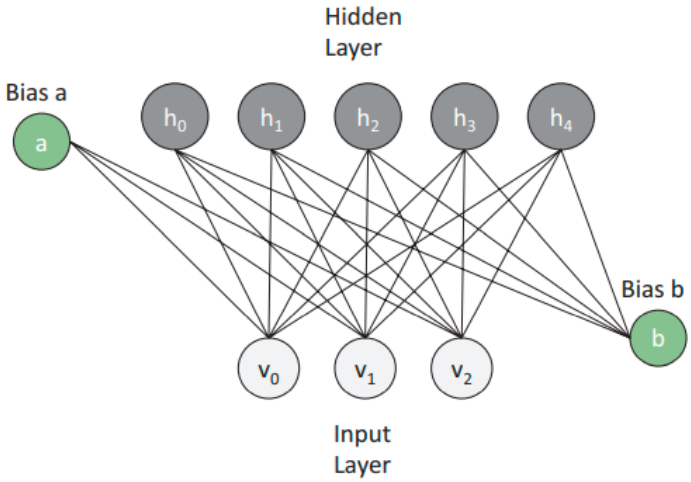

Figure 6: RBM structure (Alla and Adari; 2019)

sampling from visible and hidden layers. The learning_rate parameter was specified in the range 0.0001, 0.001, 0.01, 0.1, and 1, this parameter specifies the learning rate for weight updates. Once the best learning_rate was identified, the batch_size was tuned with the range of values of 50, 100, and 200. The best-performing model was then run to compute the hidden layer activation probabilities. The activation probabilities were appended to the source dataframe as the variable rbm. The resulting dataframe was reordered using the original index variable and the index was reset.

6.7 Adaptive Resonance Theory

ART networks use unsupervised learning to assign inputs to a cluster, the cluster assignments and weights are continually adjusted as new inputs are processed (Fausset; 2014). The ART architecture comprises three groups of neurons, an input layer, cluster units layer, and a reset layer. The main elememts of the ART2 model are illustrated in Figure 7.

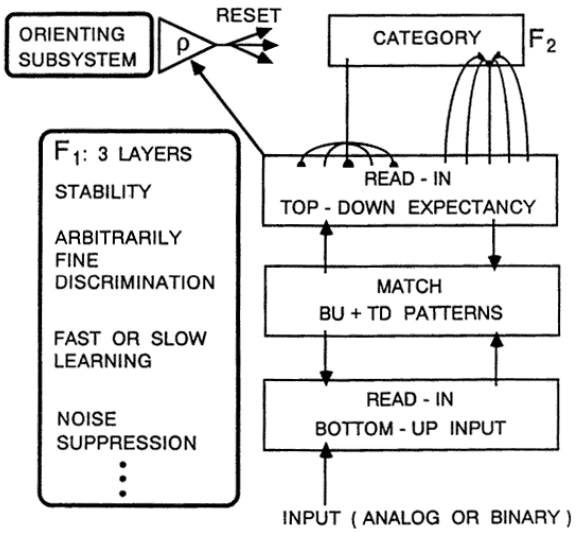

Figure 7: ART2 components(Carpenter et al.; 1991)

The ART2 model was built using the GitHub repository Art2Py[29] as there was no ART2 distribution available for use. The number of cluster units parameter, n, was set to the depth of the source dataframe, with the number of input units set to the width of the dataframe. The vigilance parameter was set to 0.9987, and parameter e, which prevents division by zero when the normalisation of a list of variables is zero, was set to 1E-6 as per the GitHub example. The result from the ART2 model is a cluster number assignment. The number of clusters was then aggregated and merged with the cluster assignment using the pandas merge class and specifying the cluster as the joining column. Any clusters that had less than 100 members were then identified as being potentially anomalous and were labeled accordingly. Due to the protracted execution time for the complete dataset, which ran into several days, a subset of 100 thousand records was used to complete the associated experiments.

7 Evaluation

In order to calculate FPR and FNR, false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN), must be determined. A TP is where the actual and predicted value are true, a FP is where the actual value is false and predicted value is true, a FN is where the actual value is true and the predicted value is false, and finally, a TN is where both the actual value and predicted value are false. The FPR, FNR, true positive rate (TPR), true negative rate (TNR), and accuracy can then be calculated based on the formulae included in Table 2.

The best performing model for each technique as per section 6, that is, the model which detected the number of anomalies closest to the number injected into the dataset, was then determined. The results from each of the applicable techniques, one hot encoding, enumeration, and grouping, were then used to determine which model performed the best by comparing the predicted variable to the anomaly variable by loading in the CSV file which contained the anomaly label. The dataframes were joined and the above performance metrics were captured and evaluated in the following sub-sections.

[29] `https://github.com/ASTARCHEN/ART2py/blob/master/ART2.py`

Table 2: FPR, FNR, TPR, TNR, and accuracy formulae (Alla and Adari; 2019)

Formula
TPR = TP/(TP + FN)
TNR = TN/(TN + FP)
FPR = FP/(FP + TN)
FNR = FN/(FN + TP)
Accuracy = (TP + TN)/(TP + TN + FP + FN)

Dataset = SIMPLE	time	TP	FP	TN	FN	Accuracy	Precision	Recall	TPR	FPR	TNR	FNR
IQR (25/75)	5.46	1000	0	999000	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
Z score +/-3	9.45	1000	0	999000	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
RBM - OHE + IQR default parameters	216.23	1000	0	999000	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
K means + IQR/Z score	362.45	1000	0	999000	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
IF - grouped (default) n_estimators= default	98.09	640	0	999000	360	0.999640	1.000000	0.640000	0.640000	0.000000	1.000000	0.360000
IF - enumeration n_estimators= 15	8.43	1	544	998456	999	0.998457	0.001835	0.001000	0.001000	0.000545	0.999455	0.999000
IF - one hot encoding n_estimators= 40	120.03	8	945	998055	992	0.998063	0.008395	0.008000	0.008000	0.000946	0.999054	0.992000
ART (100K) - OHE	12096.81	71	0	99900	29	0.999710	1.000000	0.710000	0.710000	0.000000	1.000000	0.290000
ART (100K) - enumeration	3644.41	0	0	99900	100	0.999000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000

Figure 8: Results of the experiment completed using the SIMPLE dataset

7.1 Experiment 1 - SIMPLE dataset

The first experiment was to run all the aforementioned models and techniques against the SIMPLE dataset. Reviewing Figure 8 it can be seen that the IQR, ZS, RBM, using OHE and ZS of the hidden layer activation probabilities with learning_rate set to 0.0001 and batch_size set to the default value of 10, and KM model with default parameters using ZS on cluster distance, have the best accuracy score of 1. These models also exhibit an FPR and FNR of 0, with the IQR solution exhibiting the best execution time followed by ZS, RBM and KM.

7.2 Experiment 2 - COMPLEX1 dataset

The second experiment was to run all the aforementioned models and techniques against the COMPLEX1 dataset. Reviewing Figure 9 it can be seen that the ZS, RBM using OHE and ZS of the hidden layer activation probabilities, with learning_rate set to 0.0001 and batch_size set to the default value of 10, and KM model with default parameters using ZS on cluster distance, have the best accuracy score of 1. These models also exhibit a FPR and FNR of 0, with ZS exhibiting the best execution time followed by RBM and KM.

7.3 Experiment 3 - COMPLEX2 dataset

The third and final experiment was to run all the aforementioned models and techniques against the COMPLEX2 dataset. Reviewing Figure 10, which includes the top 10 models

Dataset = COMPLEX1	time	TP	FP	TN	FN	Accuracy	Precision	Recall	TPR	FPR	TNR	FNR
Z score of +/-3	21.46	9931	0	990069	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
RBM - OHE + Z-score lr: 0.0001 batch_size = default	58.84	9931	0	990069	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
K means + Z-score default parameters	383.42	9931	0	990069	0	1.000000	1.000000	1.000000	1.000000	0.000000	1.000000	0.000000
IF - grouped n_estimators= default	99.96	9507	346	989723	424	0.999230	0.964884	0.957305	0.957305	0.000349	0.999651	0.042695
IQR (25/75)	7.79	9931	3046	987023	0	0.996954	0.765277	1.000000	1.000000	0.003077	0.996923	0.000000
RBM - OHE + IQR lr: 0.0001 bs: 200	58.84	9931	3316	986753	0	0.996684	0.749679	1.000000	1.000000	0.003349	0.996651	0.000000
K means + IQR default parameters	383.42	9931	13475	976594	0	0.986525	0.424293	1.000000	1.000000	0.013610	0.986390	0.000000
IF - one hot encoding n_estimators= 90	243.93	106	9401	980668	9825	0.980774	0.011150	0.010674	0.010674	0.009495	0.990505	0.989326
IF - enumeration n_estimators= 23	13.18	89	9842	980227	9842	0.980316	0.008962	0.008962	0.008962	0.009941	0.990059	0.991038
ART (100K) - OHE	12255.90	193	0	98960	847	0.991530	1.000000	0.185577	0.185577	0.000000	1.000000	0.814423
ART (100K) - enumeration	3768.01	0	60	99840	100	0.998400	0.000000	0.000000	0.000000	0.000601	0.999399	1.000000

Figure 9: Results of the experiment completed using the COMPLEX1 dataset

out of 17, ordered by accuracy and then execution time, it can be seen that the RBM model using the grouping technique with the ZS of the hidden layer activation probabilities, with learning_rate set to 0.0001 and batch_size set to 800, RBM model using the OHE technique with the ZS of the hidden layer activation probabilities, and the ZS statistical grouping technique, have the best accuracy scores. These models also exhibit the best FPR and FNR results. ZS exhibits the best execution time followed by the two RBM methods.

7.4 Discussion

While the ZS and IQR methods demonstrate some of the best execution times and results, due to their simplicity, these methods have limited tuning options, and are dependent on the data being grouped prior to anomaly detection, that is, in the form of a univariate dataset. Their effectiveness also wanes gradually as the complexity of the dataset increases. It can therefore be assumed that these methods would be less effective against a potentially more complex real-world dataset.

The KM method also performs favourably against all 3 datasets achieving an accuracy of 1 for both the SIMPLE and COMPLEX1 datasets, and placing 5th for the COMPLEX2 dataset, due primarily to the execution time.

The IF model performs well when using the grouping technique for the SIMPLE dataset but has a high number of TP's and FN's for both the OHE and enumeration techniques. The number of FP's and FN's increases further when the COMPLEX1 and COMPLEX2 datasets are processed.

As stated in sub-section 6.7, the number of records processed by the ART model was reduced to 100 thousand records due to protracted execution time. The results however have been included for the SIMPLE and COMPLEX1 datasets for completeness. The ART model performs very well using the OHE technique, but fails to perform using the enumeration technique. Similar results with protracted execution times can also be seen

Dataset = COMPLEX2	time	TP	FP	TN	FN	Accuracy	Precision	Recall	TPR	FPR	TNR	FNR
RBM - grouped + Z-score lr: 0.0001 bs: 800	150.37	9737	42	990056	165	0.999793	0.995705	0.983337	0.983337	0.000042	0.999958	0.016663
RBM - OHE Z-score lr: 0.0001 bs: 800	50.48	9737	45	990053	165	0.999790	0.995400	0.983337	0.983337	0.000045	0.999955	0.016663
Z score of +/-3	18.16	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
RBM - enumerated + Z-score lr: 1 bs: 50	2.51	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
K means + Z-score n_init=100	191.24	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
K means + Z-score default parameters	354.57	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
K means + Z-score n_init=200	6777.16	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
K means + Z-score n_init=300	10208.84	9737	46	990052	165	0.999789	0.995298	0.983337	0.983337	0.000046	0.999954	0.016663
IF - grouped	111.21	9479	401	989697	423	0.999176	0.959413	0.957281	0.957281	0.000405	0.999595	0.042719
IQR (25/75)	5.30	9902	3027	987071	0	0.996973	0.765875	1.000000	1.000000	0.003057	0.996943	0.000000

Figure 10: Top 10 results of the experiment completed using the COMPLEX2 dataset

for the COMPLEX1 dataset. Due to space limitations, the ART results were not included in the top 10 COMPLEX2 results.

Based on the above results, RBM will therefore be selected as the best performing model as it exhibits top 3 performance for all 3 datasets, coupled with the fact that the RBM model can be tuned using the learning_rate and batch_size parameters. The performance of the RBM method is consistently high despite the increase in complexity in the datasets, this would imply that RBM could be well placed to handle potentially more complex real-world datasets. This flexibility will allow the operator to fine-tune the process of finding potential anomalies and will also future-proof the solution against more complicated datasets. While the grouped technique marginally exhibited the best performance, the OHE technique will be integrated into the ICT solution as the data will not need to be grouped in advance. This will also future-proof the solution for multivariate real-world datasets where grouping might not be possible.

8 Deployment

The RBM model was then integrated into the final ICT solution which has been aptly named: Anomaly Detection System (ADS). The graphical user interface (GUI), as illustrated in Figure 11, was developed using the PySimpleGUI[30] library. When the GUI is launched, the user can enter, or verify the current configuration settings, and must enter the source database and repository database passwords, which are not stored for security reasons, and can fine-tune the parameters for the RBM model if desired. The user can then select the "Start" button to start the process of collecting the data from the source database, finding the anomalies, and storing the results in the repository database for

[30] `https://www.pysimplegui.org/`

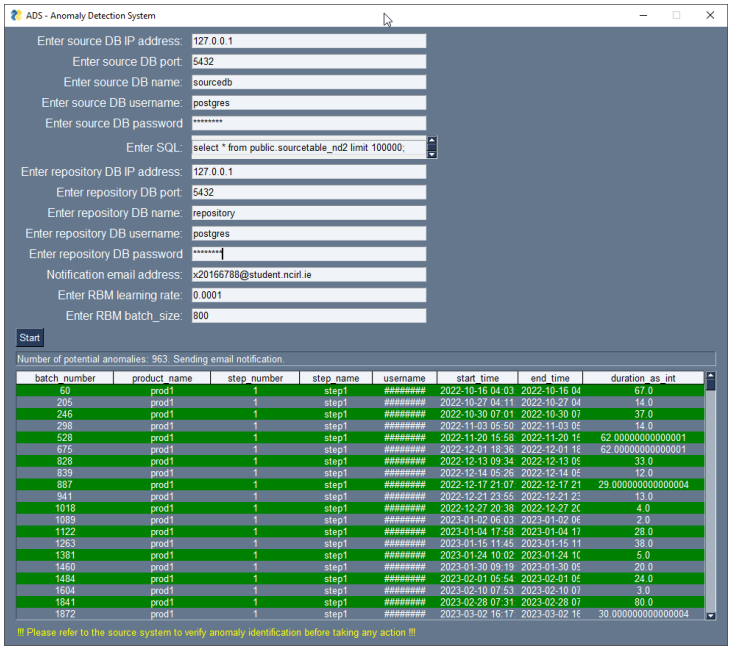

Figure 11: ADS - Graphical User Interface

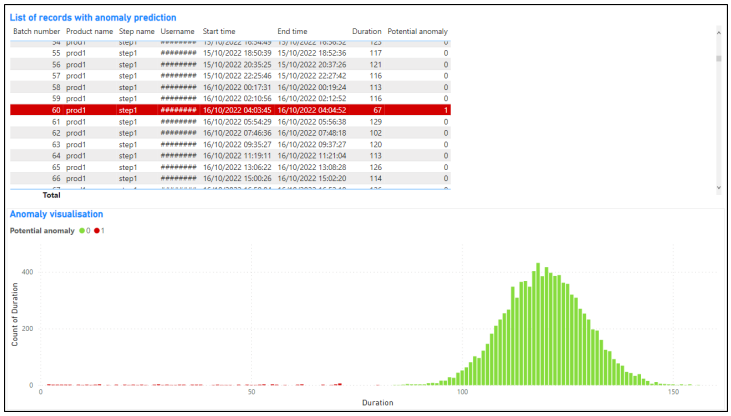

Figure 12: ADS - Power BI report

future tracking and reporting. An email notification is also sent to the address specified in the configuration settings. This email could be a distribution list to reach a wider audience.

The users responsible for reviewing, tracking, and reporting the detected anomalies can then view the PowerBI report, as per Figure 12, which illustrates the selection of one product and step combination. The data and distribution of the step duration are displayed for the users to complete their analysis, and the presence of possible anomalies is highlighted in red in the stacked bar chart and tabular chart. A table visualisation was used to display the original records with a potential anomaly variable added to indicate whether the record is anomalous, 1, or normal, 0. The conditional formatting feature was used to highlight anomalous records in red. The stacked bar chart feature was then used to visualise the distribution of the duration variable, again with 1 and 0 indicating whether the record was anomalous or normal, and represented by red and green respectively.

9 Conclusion and Future Work

This research project report demonstrates that the original research question and major contribution have been met by integrating the RBM unsupervised machine learning method into the final ICT solution to identify potential anomalous activity, by implementing a GUI to configure and fine-tune the ADS, and by implementing a visualisation of the anomalous records using PowerBI to allow the anomaly investigators to pinpoint their analysis, as per section 8. The Ethics section, section 5, also demonstrates that this solution could be commercialised in an ethical manner. The creation of the 3 synthetic datasets, as per sub-section 6.1, has also met the minor contribution of this research project.

While the performance of several of the unsupervised methods was comparable when the associated performance metrics were evaluated, as per section 7, the RBM model was selected due to its consistent performance against all 3 datasets and the potential for this method to be fine-tuned to identify potential anomalous records in potentially more complex real-world multivariate datasets.

As stated in the Introduction section, section 1, businesses would currently have to identify these non-contemporaneous activities following a manual process, by navigating through each batch and step iteratively within the associated electronic system, reviewing each of the start and end times for each activity manually to determine if there were any non-contemporaneous activities. This could involve transcribing the start and end times into a spreadsheet for future analysis which would be a huge data gathering exercise and prone to human error. This process would obviously be time-consuming, labor-intensive, and error-prone due to the volume and complexity of the data being reviewed. Innovation is therefore offered by this solution by automating the data gathering and non-contemporaneous activity detection process.

Businesses would also currently need to implement safeguards, as referenced in the Introduction section, section 1, or business rules into the existing electronic system to prevent users from completing non-contemporaneous activities. This would require an upgrade of the application, which would obviously incur a hefty cost from the application vendor and could take a protracted time to implement, if the vendor can even technically implement this feature in the first place. The fact that these systems are also used in the pharmaceutical industry means that the updates need to be validated following regulatory processes. The implementation of such an upgrade would be logistically challenging to implement. Therefore, implementing this solution in a standalone manner will be very beneficial to the business. Innovation is therefore offered by this solution by circumventing the need to upgrade or update the electronic system in order to prevent the the non-contemporaneous activity.

The developed ICT solution can therefore be used as a prototype for a commercial solution that could be used by industries requiring the recording of user activity in a contemporaneous manner to identify potential anomalous activities. This solution could therefore provide assurances to the product consumers and regulatory agencies that the associated manufacturer is taking the proactive identification of such activities seriously.

Regarding future work, the performance of the ART model could be improved and bench-marked against the RBM solution to determine whether it could be implemented in a future ICT solution. The acquisition of a real-world labeled dataset, and a repeat of the preceding experiments to verify the performance of this solution, would also be a worthwhile research project in the future.

Acknowledgement

I would like to thank my project supervisor, Mohammed Hasanuzzaman, for his support and guidance in completing this research project. Most importantly, I would also especially like to thank Shirley; my wife, and both of my sons; Hugh and Christopher, for all their support throughout the last 2 years.

References

Alla, S. and Adari, S. K. (2019). *Beginning Anomaly Detection Using Python-Based Deep Learning*, Apress.

Badia, A. (2020). *SQL for Data Science Data Cleaning, Wrangling and Analytics with Relational Databases*, Springer.

Bielecki, A. and Wójcik, M. (2021). Hybrid ai system based on art neural network and mixture of gaussians modules with application to intelligent monitoring of the wind turbine, *Applied Soft Computing* **108**.

Bonaccorso, G. (2019). *Hands-on unsupervised learning with Python : implement machine learning and deep learning models using Scikit-Learn, TensorFlow, and more*, Packt.

Brahma, A., Panigrahi, S. and Mahapatra, J. (2020). Anomaly detection in database using bat algorithm., *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Computer Science, Engineering and Applications (ICCSEA), 2020 International Conference on* pp. 1 – 5.

Carpenter, G. A. and Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine, *Computer Vision, Graphics and Image Processing* **37**.

Carpenter, G., Grossberg, S. and Rosen, D. (1991). Art 2-a: an adaptive resonance algorithm for rapid category learning and recognition, *IJCNN-91-Seattle International Joint Conference on Neural Networks*, Vol. ii, pp. 151–156 vol.2.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000). Crisp-dm -cross-industry standard process for data mining- 1.0 step-by-step data mining guide., *CRISP-DM Consortium* .

Demertzis, K., Iliadis, L., Pimenidis, E. and Kikiras, P. (2022). Variational restricted boltzmann machines to automated anomaly detection, *Neural Computing and Applications* **34**.

Do, K., Tran, T., Phung, D. and Venkatesh, S. (2016). Outlier detection on mixed-type data: An energy-based approach, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 10086 LNAI.

Dunning, T. and Friedman, E. (2014). *Practical Machine Learning: A New Look At Anomaly Detection*, O'Reilly.

Fadolalkarim, D., Bertino, E. and Sallam, A. (2020). An anomaly detection system for the protection of relational database systems against data leakage by application programs, *Proceedings - International Conference on Data Engineering* **2020-April**.

Fausset, L. (2014). *Fundamentals of Neural Networks: Architectures, Algorithms And Applications*, Vol. 1, Pearson.

Gaber, C., Hemery, B., Achemlal, M., Pasquet, M. and Urien, P. (2013). Synthetic logs generator for fraud detection in mobile transfer services, *Proceedings of the 2013 International Conference on Collaboration Technologies and Systems, CTS 2013*.

Gaikwad, D. P. and Thool, R. C. (2015). Intrusion detection system using bagging with partial decision tree base classifier, *Procedia Computer Science* **49**.

Gong, X., Zhang, F., Lu, T. and You, W. (2022). Comparative analysis of three outlier detection methods in univariate data sets, *2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Electronic Communication and Artificial Intelligence (IWECAI), 2022 3rd International Conference on, IWECAI* pp. 209–213.

Grossberg, S. (1976). Adaptive pattern classification and universal recoding: Ii. feedback, expectation, olfaction, illusions, *Biological Cybernetics* **23**.

Islam, M. S., Kuzu, M. and Kantarcioglu, M. (2015). A dynamic approach to detect anomalous queries on relational databases, *CODASPY 2015 - Proceedings of the 5th ACM Conference on Data and Application Security and Privacy* .

Jesus, G., Casimiro, A. and Oliveira, A. (2021). Using machine learning for dependable outlier detection in environmental monitoring systems, *ACM Transactions on Cyber-Physical Systems* **5**.

Jones, C. B., Carter, C. and Thomas, Z. (2018). Intrusion detection response using an unsupervised artificial neural network on a single board computer for building control resilience, *Proceedings - Resilience Week 2018, RWS 2018*.

Kamra, A., Terzi, E. and Bertino, E. (2008). Detecting anomalous access patterns in relational databases, *VLDB Journal* **17**.

Kopp, S. (2019). Guideline on data integrity, *WHO Drug Information* **33**.

Lazarevic, A. and Kumar, V. (2005). Feature bagging for outlier detection, *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* .

Liu, S., Jia, C. and Tao, Y. (2015). An art2-based pumping unit fault diagnosis, *Proceedings of the 2015 27th Chinese Control and Decision Conference, CCDC 2015*.

Lopez-Rojas, E. A. and Axelsson, S. (2012a). Money laundering detection using synthetic data, *The 27th workshop of (SAIS)*.

Lopez-Rojas, E. A. and Axelsson, S. (2012b). Multi agent based simulation (mabs) of financial transactions for anti money laundering (aml), *Nordic Conference on Secure IT Systems*.

Mejia-Lavalle, M. (2010). Outlier detection with innovative explanation facility over a very large financial database., *2010 IEEE Electronics, Robotics and Automotive Mechanics Conference, Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2010* pp. 23 – 27.

Mejía-Lavalle, M. (2010). Outlier detection with innovative explanation facility over a very large financial database, *Proceedings - 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference, CERMA 2010* .

Parisi, A. (2019). *Hands-on artificial intelligence for cybersecurity : implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies*, Packt.

Ringberg, H., Roughan, M. and Rexford, J. (2008). The need for simulation in evaluating anomaly detectors, *Computer Communication Review*, Vol. 38.

Robinson, S. C. (2017). What's your anonymity worth? establishing a marketplace for the valuation and control of individuals' anonymity and personal data, *Digital Policy, Regulation and Governance* **19**.

Ronao, C. A. and Cho, S. B. (2016). Anomalous query access detection in rbac-administered databases with random forest and pca, *Information Sciences* **369**.

Saltz, J., Skirpan, M., Fiesler, C., Gorelick, M., Yeh, T., Heckman, R., Dewar, N. and Beard, N. (2019). Integrating ethics within machine learning courses, *ACM Transactions on Computing Education* **19**.

Schniepp, S. J. (2019). Alcoa+ and data integrity, *Pharmaceutical Technology* **43**.

Unger, B. (2020). The 10 most-cited mhra gmp inspection deficiencies by annexchapter. **URL:** *https://www.pharmaceuticalonline.com/doc/the-most-cited-mhra-gmp-inspection-deficiencies-by-annex-chapter-0001*

Vanerio, J. and Casas, P. (2017). Ensemble-learning approaches for network security and anomaly detection, *Big-DAMA 2017 - Proceedings of the 2017 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks, Part of SIGCOMM 2017* .

Wicklin, R. (2013). *Simulating data with SAS*, SAS Institute.

Yoseph, F. and Heikkila, M. (2019). A clustering approach for outliers detection in a big point-of-sales database, *Proceedings - International Conference on Machine Learning and Data Engineering, iCMLDE 2019* .

Zadafiya, N., Karasariya, J., Kanani, P. and Nayak, A. (2022). Detecting credit card frauds using isolation forest and local outlier factor - analytical insights, *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 1588–1594.

Zhong, S., Fu, S., Lin, L., Fu, X., Cui, Z. and Wang, R. (2019). A novel unsupervised anomaly detection for gas turbine using isolation forest, *2019 IEEE International Conference on Prognostics and Health Management, ICPHM 2019*.