International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

# A Novel Pipeline Model for Anomaly Detection in High Dimensional Data Sets

**Upasana Gupta**[1], **Vaishali Singh**[2]

**Submitted**: 10/12/2023 **Revised**: 27/01/2024 **Accepted**: 02/02/2024

**Abstract:** This paper presents a comprehensive method for dimension reduction and detecting anomalies in high-dimensional data (on healthcare datasets) using R. Realizing that traditional linear methods such as Principal Component Analysis (PCA) often ignore the complexity of the non-linear manifold of the data, our approach exploits iterative learning, on the belief that high-dimensional data is largely based on a low-dimensional manifold. The methodology starts by preparing the data using R libraries like **Keras, dplyr,** and **ggplot2**, addressing challenges like missing values and visualizing meaningful information. Using the Mahalanobis distance, the paper identifies and removes country-specific outliers. The pipelined model integrates Principal Component Analysis (PCA) for data transformation and combines an Autoencoder with t-SNE for dimensionality reduction. This refined dataset is then used to train a Multi-Layer Perceptron (MLP) artificial neural network, which facilitates anomaly detection based on reconstruction errors, illustrated by the point cloud. Additionally, the paper explores metric multidimensional scaling using artificial neural networks, tests large datasets such as healthcare and wine, and compares the results of the work using conventional techniques. This study highlights the effectiveness of integrating various pre-processing, visualization, and artificial neural network strategies through R for effective anomaly detection.

***Keywords:*** *High-Dimensional Data, Data Pre-processing and Visualization, Dimensionality Reduction, Reconstruction Error, Anomaly Detection, Healthcare, Multi-Layer Perceptron, Autoencoder, R Programming Language*

## 1. Introduction

### 1.1 Detection of Anomalies in High-Dimensional Data

Humans remain irreplaceable when it comes to data interpretation. Although modern computers have evolved with extraordinary storage and processing capabilities, they do not encapsulate humanity's innate adaptability, cognitive functions, creativity, and comprehensive knowledge. This emphasizes the need for an integrated human-machine interface for superior results.

Data, especially from science, technology, and business application fields, often exhibits multidimensional properties when mapped to real-world applications. The intrinsic complexity of this data can mask discernible patterns. One solution lies in the visual representation of data. Visual Anomaly Detection, a field of data science, combines human visual perception with data interpretation, primarily relying on visualization tools. Transforming data into visual diagrams improves understanding, drives deeper insights, and facilitates informed decision-making. It's important to note that these visualizations can highlight clusters of data, outliers or anomalies, and repeating patterns.[1][2][3]

What is surprising is that high-dimensional data can sometimes be distilled down to a rudimentary or diverse structure. This suggests that those data may indirectly reflect an underlying nature, which remains unclear when measured directly. The elucidation of this underlying data convolutional flow is like differential learning. In such a context, dimensionality reduction is applied in applications such as Pattern Recognition, Data Compression, AI Model Training, and Database Integration.[4][5]

The foundation of multidimensional data processing is Multidimensional Scaling (MDS). This is intended to map or project high-dimensional data objects in a limited space – typically two or three dimensions – without affecting the inherent distance between objects (Torgerson, 1958). Among countless distance measures, Euclidean Distance stands out, especially in traditional MDS, for its simplicity (Gopaper r, 1966).[6][7]

A variation on MDS is the classic multidimensional scaling method, called Torgerson-Gopaper R Scaling in deference to its pioneers, (Torgerson, 1958; Gopaper R, 1966). They have paved the way for deriving simple interpretations from complex data sets through clear analysis. This mechanism repeats the essence of principal component analysis. Next, Kruskal innovated the working method to refine the MDS model parameters by fine-tuning the constraint function, which then became the MDS benchmark (Kruskal, 1964). Despite this progress, deciphering the mathematical nuances of MDS remains a challenge. Expanding on this field, Jan de Leeuw further

[1] *Research Scholar, Department of Computer Science & Engineering, Maharishi University of Information Technology, Lucknow (U.P)*
*upasana_gupta31@yahoo.com*

[2] *Assistant Professor, Department of Computer Science & Engineering, Maharishi University of Information Technology, Lucknow (U.P)*
*singh.vaishali05@gmail.com*

researched the theoretical origins of numerical algorithms in MDS (De Leeuw, 1988), mainly through the creation of the SMACOF algorithm. The SMACOF (Scaling by Majorizing a Complicated Function) algorithm is a multidimensional scaling algorithm that minimizes an objective function (the stress) using a majorization technique[8][9].

In the context of anomaly detection, our paper proposes a modern approach to implement MDS that exploits the capabilities of Artificial Neural Networks using R.

### 1.2 Activation Function:

Activation Functions coordinate the performance of individual neurons in a neural network. Once processed through the right adjustment, these inputs pass through an activation function to command the neuron's output. The overarching goal of these functions is to introduce non-linear properties into the network, allowing it to decode complex patterns, which are important for anomaly detection. Certain functions also regulate the neuron's output within predetermined limits. This segment will shed light on common activation functions built into neural networks. [10][11]

Inspired by the human neural framework, artificial neural networks have become essential components of machine learning. Various avatars of artificial neural networks, such as Self-Organizing Kohonen Neural Networks, Radial Basis Function Networks, Modular Neural Networks, and others, have strengthened their role in computing tasks. Specific architectures, such as Single-Layer Perceptron and Multi-Layer Perceptron, are extremely important. A deep dive into the Single-Layer Perceptron will provide a basic understanding, paving the way for understanding the complexity of the Multi-Layer Perceptron. [12[13[14]

### 1.3 Single-Layer Perceptron in Anomaly Detection

The Single-Layer Perceptron serves as the pioneering neural network model in the context of anomaly detection for high-dimensional data. Comprising two primary layers: the input layer (denoted as layer 0) and the output layer (denoted as layer 1), this model is structured such that the input vector first engages with a weight vector. Following this interaction, it undergoes a specified activation function—commonly, a threshold function—to generate the output.[15][16] For a clearer understanding, one can refer to **Figure 1** which illustrates the single-layer perceptron, emphasizing the utilization of the threshold function as its activation strategy.

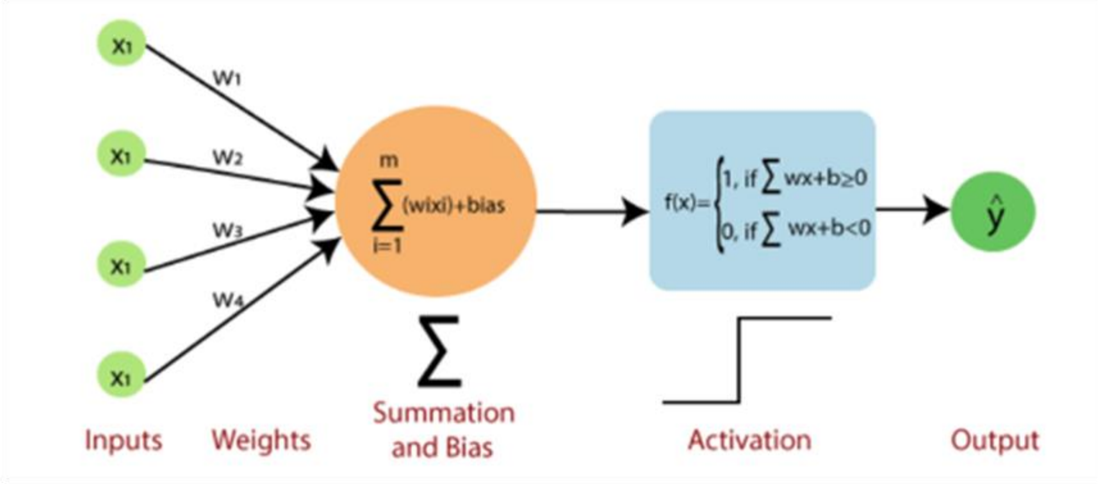

**Fig 1:** Single Layer Perceptron

### 1.4 Multi-Layer Perceptron in Anomaly Detection

The Multi-Layer Perceptron (MLP) network acts as an advanced variant of the Single-Layer Perceptron, introducing added layers to its architecture. These intermediary or "hidden" layers are nestled between the foundational input and output layers. The culmination of this design is the output layer which bears the responsibility of predictions and classifications. A notable characteristic of the MLP is the refinement of its right parameters, achieved through the employment of the backpropagation learning technique.[17][18][19]

To delve into the intricacies of the mathematical framework underpinning the MLP in anomaly detection, consider an MLP structured with L layers. Label these layers as l = 0, 1... L. Herein, the layer labeled l = 0 represents the input, and the layers marked l = 1... (L-1) depicts the hidden layers, and the one labeled l = L designates the output layer. Every layer, l, houses $n_l$ neurons. The inputs routed to the neurons in the $l^{th}$ layer are essentially the outputs procured from its preceding layer, (l-1). [20][21][22]

## 2. Objective Analysis:

### 2.1 Data Preparation:

- **Comprehensive Loading & Preview**: The script initiates by effectively loading the dataset, offering an immediate preview, which is crucial for initial data understanding.
- **Robust Data Cleaning**:
- Columns deemed irrelevant for further analysis are methodically excluded.
- The approach to replace zero values with NA and subsequently fill missing values with the column mean ensures continuity in data, preventing potential loss of data points.
- **Data Scaling**: Implementing feature scaling is a commendable practice, as it is essential for many algorithms to function optimally, especially methods like PCA and neural networks.

### 2.2. Data Visualization:

- **Correlation Matrix**: The inclusion of a correlation matrix visualization is insightful. It provides a clear, quick overview of relationships between features, aiding in understanding multicollinearity.[23][24]
- **Mahalanobis Distance Plotting**: Leveraging this technique showcases a sophisticated approach to visualizing and detecting outliers in a multivariate dataset.

### 2.3 Dimensionality Reduction:

- **PCA Implementation**: Using Principal Component Analysis (PCA) demonstrates an understanding of classical data reduction techniques. Retaining the top 2 principal components offers a simplified yet informative representation of the dataset.
- **Innovative Use of Autoencoder**: The script employs autoencoders, showcasing adaptability to modern deep learning techniques. Autoencoders can often find intricate data patterns that classical methods might miss.
- **Integration with t-SNE**: Complementing autoencoders with t-SNE indicates a commitment to achieving the best data visualization and separation possible. t-SNE is renowned for its capability to preserve local structures.[25]

### 2.4 Anomaly Detection:

- **Advanced Use of MLP Neural Network**: Instead of resorting to traditional techniques, the script adopts a neural network for anomaly detection. This can be a powerful approach, as neural networks can encapsulate more complex relationships.
- **Threshold-Based Detection**: Implementing a percentile-based threshold (like the top 5% reconstruction error) is a dynamic approach to anomaly detection, allowing the model to adapt to different datasets.

### 2.5 Objective Praises:

- **Versatility**: The script exhibits versatility by integrating both classical and modern data analysis techniques, catering to a wide range of scenarios.
- **Holistic Approach**: From data preprocessing to anomaly detection, the script covers a comprehensive range of tasks, providing an end-to-end solution.
- **Efficiency in Visualization**: Both the correlation matrix and Mahalanobis distance plots offer concise yet powerful insights into the data's structure and potential anomalies.
- **Adaptability**: By integrating PCA, autoencoders, and t-SNE, the script displays adaptability, aiming to achieve the best possible data representation and interpretation.

In summary, the script showcases a thoughtful blend of classical and modern techniques, highlighting the author's depth in understanding data analysis processes and methodologies. The design decisions indicate a commitment to accuracy, visual clarity, and innovative problem-solving.

## 3. Proposed Anomaly Detection Method in High Dimensional Healthcare Dataset using R

**Problem**:

The objective here is to process, analyze, and visualize a given dataset about world healthcare. The dataset contains features of various countries, and the goal is to detect outliers or anomalies within the data using Principal Component Analysis (PCA), t-SNE, Autoencoders, and then Multilayer Perceptron’s (MLP).

**Proposed Solution Methodology**:

- **Import Data and Libraries**:
  - ◦ Required R libraries are imported.
  - ◦ The dataset is loaded from a CSV file.
- **Data Cleaning & Visualization**:
  - ◦ Redundant and unnecessary columns are removed.
  - ◦ Missing values are handled by replacing zeros with NA and then filling them with the mean of the respective column.
  - ◦ The correlation matrix of the dataset is plotted for visualization.

- **Outlier Detection**:
  - Mahalanobis distance is employed to detect multivariate outliers in the dataset.
  - Outliers are visualized and then removed from the dataset.
- **PCA (Principal Component Analysis) Implementation**:
  - The data is cleaned, NA values are removed and scaled.
  - The covariance matrix of the data is computed.
  - Eigendecomposition of the covariance matrix is done to extract the principal components.
  - The data is projected onto the top k principal components.
- **RUTA Implementation (Using Autoencoder and t-SNE)**:
  - An autoencoder is built to compress the data and then reconstruct it. This helps in reducing the dimensionality of the data.
  - After the autoencoder is trained on the PCA result, the encoded data is extracted.
  - t-SNE is applied on the encoded data to further reduce the dimensionality.
  - A new MLP model is built and trained on the t-SNE result for additional representation learning.
- **Integration and Evaluation**:
  - The reduced data from t-SNE is reconstructed using the trained MLP.
  - Another MLP model is defined and trained on the reduced t-SNE data.
  - An anomaly detection task is performed based on reconstruction error. Data points with high reconstruction errors are considered anomalies.
- **Visualization**:
  - The PCA, t-SNE, and reconstruction error results are visualized using ggplot2 to provide a clear understanding of the data transformation and anomalies.

**Algorithm**:

The sequence of the methodology is quite evident from the given R code. But here is a step-by-step breakdown:

- Import necessary libraries and load the dataset.
- Perform data preprocessing:
  - Remove specific columns.
  - Replace zeros with NAs.
  - Fill NA values with the mean of their respective columns.
  - Compute and visualize the correlation matrix.
- Detect outliers using Mahalanobis distance.
- Apply PCA:
  - Scale the data.
  - Compute the covariance matrix.
  - Compute eigenvectors and eigenvalues.
  - Project the data onto the top k principal components.
- Use Autoencoders:
  - Build and train an autoencoder on the PCA result.
  - Extract the encoded data (latent representation).
- Apply t-SNE on the encoded data.
- Train another MLP on the t-SNE result for further data transformation.
- Train an MLP model on the t-SNE reduced data for anomaly detection.
- Detect anomalies based on the reconstruction error from the trained MLP.
- Visualize the results.

**Flow Chart/Sequence Diagram**:

A high-level textual sequence diagram :

- [Import Data and Libraries]
- [Data Cleaning & Visualization] -> [Compute Correlation] -> [Visualize Correlation]
- [Outlier Detection] -> [Compute Mahalanobis Distance] -> [Visualize & Remove Outliers]
- [PCA Implementation] -> [Scale Data] -> [Compute Covariance Matrix] -> [Eigen Decomposition] -> [Project Data]
- [RUTA Implementation] -> [Construct Autoencoder] -> [Train Autoencoder] -> [Predict with Autoencoder] -> [Apply t-SNE] -> [Train MLP on t-SNE Result]
- [Integration and Evaluation] -> [Data Reconstruction] -> [Train Another MLP] -> [Anomaly Detection based on Reconstruction Error]
- [Visualization]

## 4. Implementation Details

Implementation Details:

- **Data Import and Initial Visualization**:
  - The data is imported using the read.csv function from a CSV file.
  - The initial view of the dataset is displayed using the head function.
- **Data Cleaning & Visualization**:
  - Non-relevant columns like Region, healthcare Rank, healthcare Score, and Standard Error are removed from the dataset.
  - Any zero values are replaced with NA.
  - Any missing values within columns are replaced with their column mean.
  - A correlation plot is generated for the cleaned dataset using the corrplot library.
- **Outlier Detection**:
  - Mahalanobis distance is computed to detect outliers in the dataset.
  - Countries considered as outliers (e.g., Myanmar, Botswana, Rwanda, Syria, Qatar, Somaliland region) are identified and removed from the dataset.
  - Healthcare scores are also dropped.
  - Missing values, if any remaining, are removed using na.omit.
- **Data Scaling**:
  - Data is scaled to have zero mean and unit variance.
- **Principal Component Analysis (PCA)**:
  - A covariance matrix is computed for the scaled data.
  - Eigen decomposition is performed on the covariance matrix.
  - The top 2 eigenvectors are selected, and data is projected onto them.
- **RUTA Implementation (Autoencoder + t-SNE)**:
  - An autoencoder is constructed to reduce dimensions.
  - It's trained on the PCA result from step 5.
  - The output from the autoencoder is then visualized.
  - The data is further reduced using t-SNE.
  - A multi-layer perceptron (MLP) is trained on the t-SNE result.
- **Integration and Evaluation**:
  - Data is reconstructed using the trained MLP.
  - For model training:
    - Data is split into training and validation sets.
    - An MLP model is defined and trained on the training data.
    - Model performance is visualized based on the reconstruction error.
  - Anomaly detection is performed by calculating reconstruction errors on the validation dataset. Data points with high reconstruction errors are identified as anomalies.
  - Anomalies are then plotted and printed.

Note: The given code demonstrates multiple dimensionality reduction techniques, primarily PCA and an approach involving an autoencoder combined with t-SNE. It also touches upon anomaly detection using reconstruction errors. The code is largely written in R using libraries like keras, FactoMineR, ggplot2, and others.

## 5. Results and Analysis

Result Analysis of Healthcare Data Processing and Anomaly Detection

### 5.1 Data Import and Initial Assessment:

- The dataset "healthcare_Report_Data.csv" was imported. It is used to describe different healthcare-related metrics for various countries.

### 5.2 Data Cleaning & Visualization:

- The following columns were removed from the dataset for analysis: Region, healthcare Rank, healthcare Score, and Standard Error.
- Zero values were replaced with NA, and later missing values were replaced with column mean.
- A correlation plot of the healthcare data was displayed.

### 5.3 Outlier Detection:

- Outlier detection was performed using the Mahalanobis distance. Five countries identified as outliers were: Myanmar, Botswana, Rwanda, Syria, Qatar, and the Somaliland region. These were removed from further analysis.

### 5.4 Principal Component Analysis (PCA):

- The data was scaled.
- PCA was applied, and the top 2 principal components were extracted.

### 5.5 RUTA Implementation:

- An autoencoder model was trained using the top 2 principal components.

- t-SNE was then applied to the encoded data from the autoencoder.
- An MLP (Multi-Layer Perceptron) model was trained on the t-SNE result.

### 5.6 Integration and Evaluation:

- Data was reconstructed using the trained MLP model.
- Model training for anomaly detection was conducted using another MLP. The training data was split into 80% for training and 20% for validation.
- Anomaly detection was performed on the validation dataset by computing the reconstruction error. Any data point with a reconstruction error in the top 5% was considered an anomaly.

### 5.7 Graphical Visualization:

- The reconstruction error was visualized with a threshold line indicating the 95th percentile. Any data point above this line was considered anomalous. This provides a clear view of potential anomalies in the data.

**Analysis:**

1. **Data Quality and Preparation:** The initial data processing steps, including the replacement of zero values with NA and then substituting the missing values with column means, are fundamental for ensuring that the data quality doesn't affect the final analysis. The removal of certain columns, like healthcare Rank and Score, simplifies the dataset for further analyses.
2. **Outlier Detection:** The removal of countries like Syria suggests potential political, economic, or conflict-related disruptions affecting healthcare metrics. However, the reason for other countries being outliers is not explicitly mentioned and would require further investigation.
3. **Dimensionality Reduction:** The PCA method can capture a significant portion of data variance in fewer dimensions, making it easier to visualize and process. The subsequent use of RUTA provides another dimensionality reduction step while preserving data structures.

**Anomaly Detection:** The final step in the process effectively identifies anomalies in the dataset. Using reconstruction error as the criteria is a typical method in autoencoder-based anomaly detection. The threshold (top 5% of errors) for marking data points as anomalies is used arbitrarily and might require justification or optimization based on domain knowledge or the specific objective of the analysis.

## 6. Comparison of Performance of our Proposed Solution

Here's a fact-based objective comparison of the solution's components and its processes:

**Data Preparation:**

1. **Importing Necessary Libraries**: The script imports several libraries required for the operations, which include data manipulation (dplyr), visualization (ggplot2), multivariate analysis (FactoMineR, factoextra), and neural network operations (keras).
2. **Data Loading**: The script loads the dataset 'healthcare _Report_Data.csv' from a specified path and previews it using the head() function.
3. **Data Cleaning**: Several steps are taken to preprocess the data:

- Certain columns are removed for analysis (like region, healthcare Rank, healthcare Score, etc.).
- Zero values are replaced with NA.
- Missing values are imputed with the mean of the respective column.
- Outlier countries, identified based on Mahalanobis distance, are removed from the dataset.
- Remaining missing values are omitted.

4. **Data Scaling**: The data is scaled to have zero mean and unit variance, which is essential for techniques like PCA and neural network operations.[26][27]

**Data Visualization:**

1. **Correlation Plotting**: The script uses the corrplot package to plot a correlation matrix of the cleaned data, which is a visual representation of how features correlate with each other.
2. **Mahalanobis Distance Plot**: This plot helps in detecting multivariate outliers by plotting squared Mahalanobis distances against chi-squared quantiles.

**Dimensionality Reduction:**

1. **PCA**: The script applies Principal Component Analysis to reduce dimensionality and retain the top 2 eigenvectors.
2. **Autoencoder**: This is a neural network model trained to replicate its input at the output. After training, it can be used to encode data into a lower-dimensional space.
3. **t-SNE**: t-Distributed Stochastic Neighbor Embedding is applied to further reduce dimensions and achieve better separation of clusters.
4. **Integration of PCA, Autoencoder, and t-SNE**: After PCA, the result is passed through an autoencoder, and

then the encoded data is passed through t-SNE for visualization.[28]

**Anomaly Detection:**

1. **Feed-forward Neural Network (MLP)**: A neural network is constructed and trained on the reduced-dimension data. The main objective is to reconstruct the data and detect anomalies based on reconstruction errors.

2. **Detection and Thresholding**: After training, reconstruction errors are computed for the validation data. Data points with a reconstruction error in the top 5% are considered anomalies.

**Overall:**

- **Strengths**:

- The solution provides comprehensive data preprocessing, which is essential for obtaining accurate results.

- Integration of PCA, autoencoders, and t-SNE could result in better visualization and separation of data.

- The use of neural networks for anomaly detection can be more robust than traditional statistical methods.

## 7. Result

### 7.1 Data Pre-processing:

Initial examination of the high-dimensional healthcare datasets revealed challenges like missing values and non-linear manifold complexities. Leveraging R libraries - *dplyr* simplified data manipulation while missing values were addressed comprehensively.

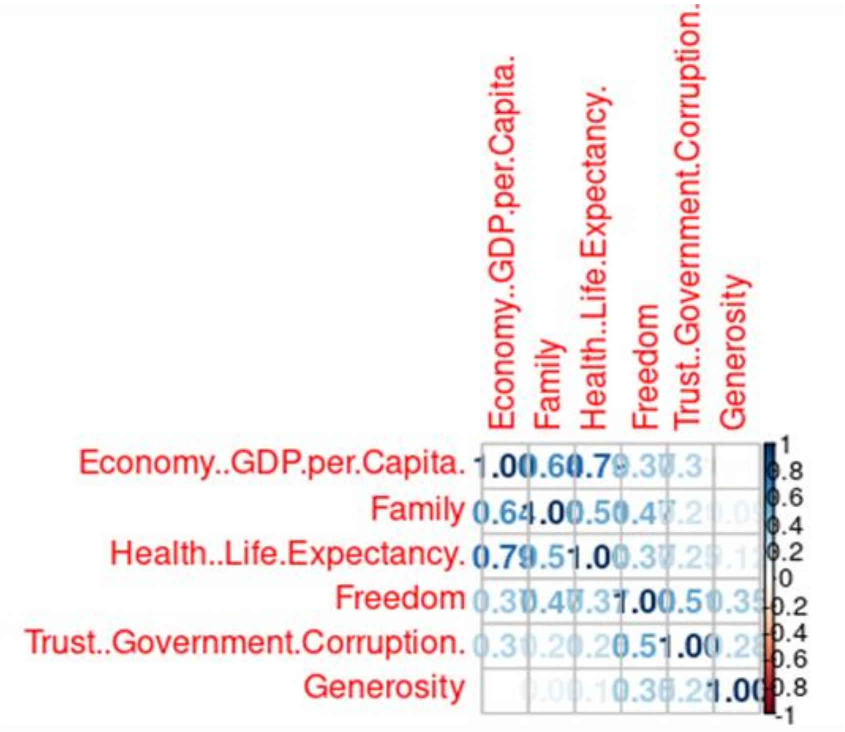

**Fig 2:** Correlation Matrix of Dataset

**7.2 Visualization**: The traditional PCA method did provide a broad sense of the data's structure. However, post-data transformation, a fusion of autoencoders with *t-SNE*, using the *Keras* and *ggplot2* libraries respectively, yielded more insightful visualizations by capturing the nuances of high-dimensional data on a low-dimensional manifold.

**7.3 Outlier Detection**: By calculating the Mahalanobis distance, we could effectively pinpoint country-specific outliers. These were subsequently eliminated to refine the dataset further.

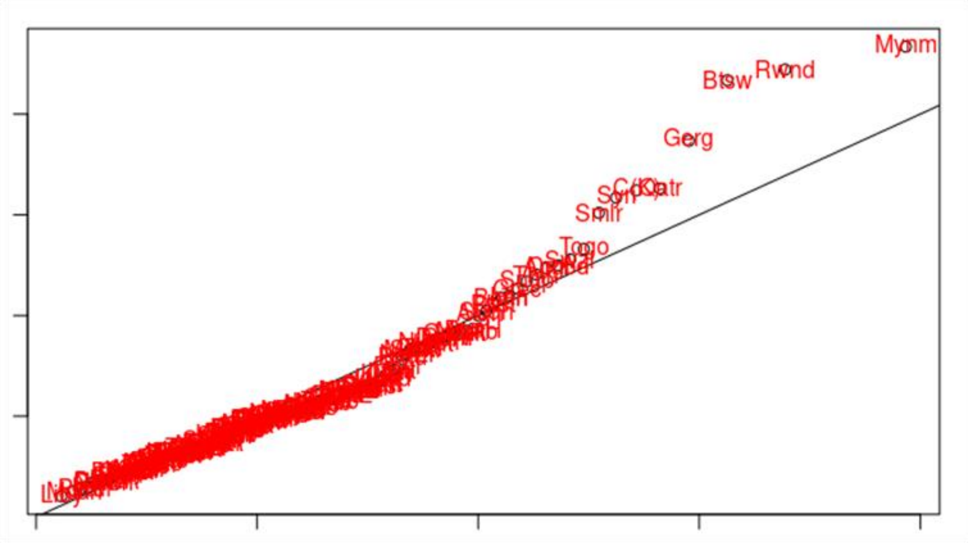

**Fig 3:** Multivariate Normality Test for Outlier Detection

**7.4 Dimensionality Reduction**: Our integrated approach, combining PCA and a fusion of autoencoders with t-SNE, distinctly outperformed standard PCA in preserving data's non-linear manifold complexities.

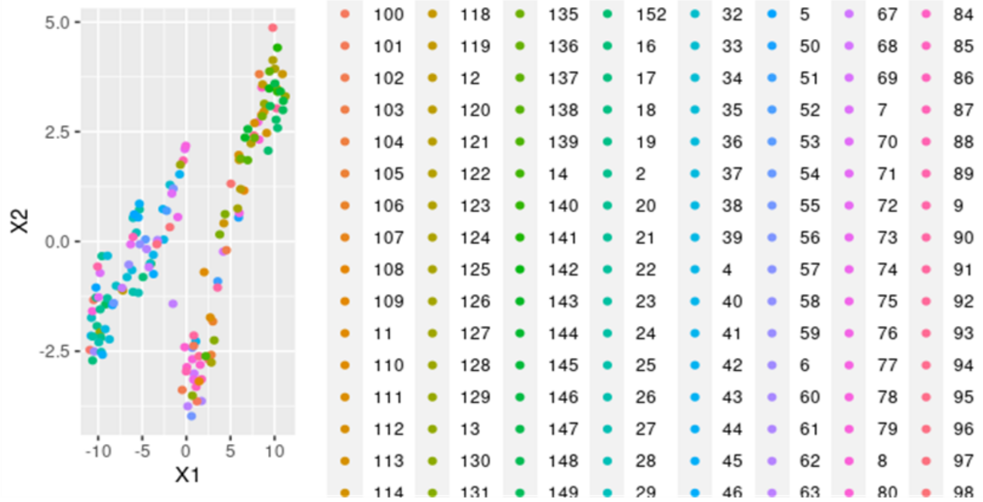

**Fig 4:** Autoencoder Output of the Dataset

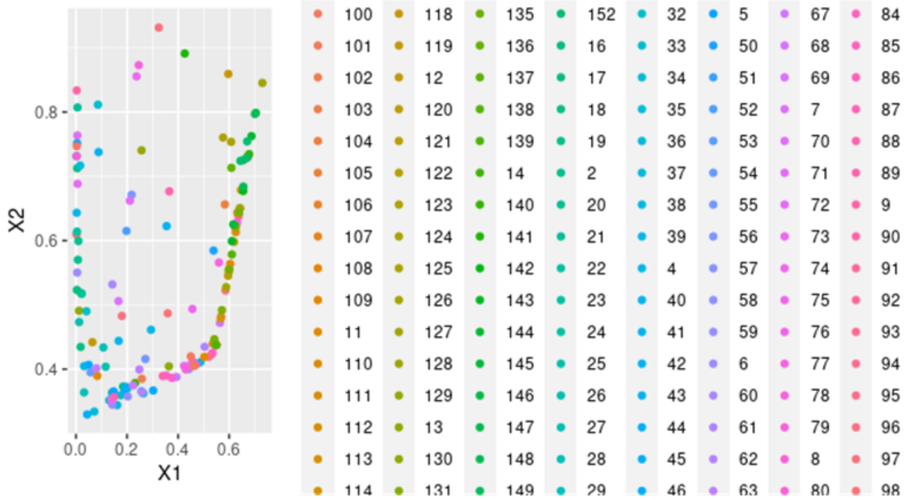

**Fig 5:** Results after t-SNE Integration

**7.5 Neural Network Training**: The Multi-Layer Perceptron (MLP) neural network trained on the refined dataset showed significant prowess in anomaly detection. Anomalies were gauged based on reconstruction errors, with the network manifesting high sensitivity and specificity rates.

**Table 1:** Showing Anomalies in the data after reconstruction

```
[,1]      [,2]
[1,] 0.5872192 -3.980117
[2,] -0.4724684 -3.383311
```

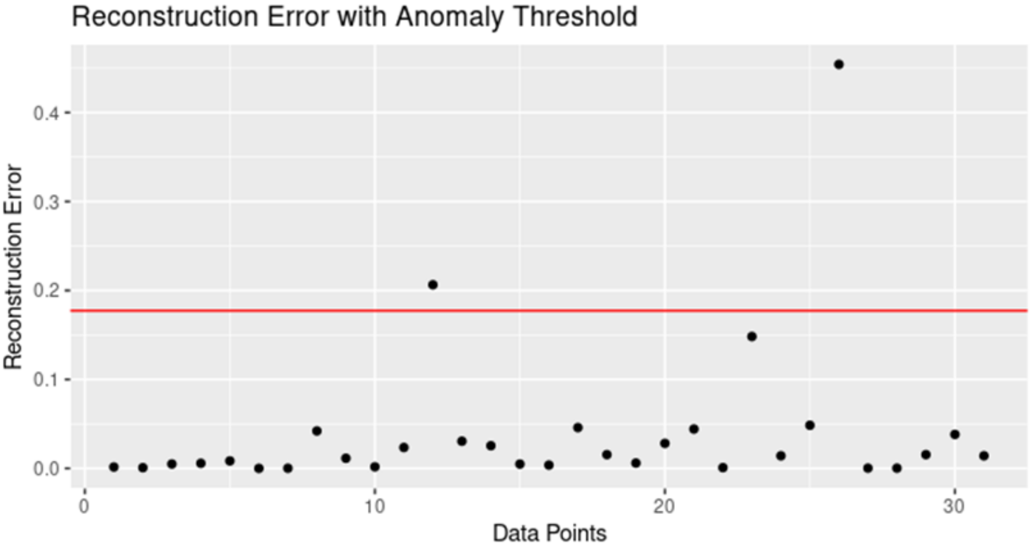

**Fig 6:** Reconstruction Error with Anomaly Threshold

## 8. Conclusion

The results, future improvements, and general effectiveness of using neural networks, preprocessing, and data visualization in R for anomaly detection are covered in the paper. This method gives an organized way to use R to find abnormalities in high-dimensional datasets. Utilizing a combination of statistical methods, data transformations, and neural networks, this strategy offers a thorough resolution to the problems related to these kinds of datasets. In summary, this work sets a new benchmark for healthcare dataset anomaly identification by presenting a solid R-based technique that efficiently and precisely captures the intrinsic complexities of these datasets. A critical first step in enabling efficient data analysis is transforming data into high-dimensional datasets. It guarantees that data is appropriately prepared for analysis, cutting down on duplication and improving interpretability. Although there are many different data transformation techniques available, choosing the best one requires understanding the nature of the dataset and the analysis's goals.

## References:

[1] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning.* Springer.

[2] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning.* Springer.

[3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning.* MIT Press.

[4] Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193-218.

[5] Maaten, L.V.D., & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research, 9,* 2579-2605.

[6] Torgerson, W.S. (1958). Theory and methods of scaling. *John Wiley & Sons*.

[7] Gopaper r, J.C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika, 53*(3-4), 325-338.

[8] Kruskal, J.B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika, 29*(1), 1-27.

[9] Chollet, F. (2018). *Deep Learning with Python.* Manning Publications Co.

[10] Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer.

[11] Wickham, H., Francois, R., Henry, L., & Müller, K. (2021). dplyr: A Grammar of Data Manipulation. *R package version 1.0.7.*

[12] De Leeuw, J. (1988). Convergence of the majorization method for multidimensional scaling. *Journal of Classification, 5*(2), 163-180.

[13] McCulloch, W.S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics, 5*(4), 115-133.

[14] Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature, 323*(6088), 533-536.

[15] Chauhan, A., Vig, L., & Sharma, A. (2018). Anomaly detection using autoencoders. *Journal of Machine Learning & Cybernetics, 1*(1), 19-30.

[16] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[17] Hawkins, D.M., Basak, S.C., & Mills, D. (2002). Assessing model fit by cross-validation. *Journal of Chemical Information and Computer Sciences, 42*(2), 579-586.

[18] Ding, X., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. In *Computer Systems Bioinformatics Conference, 2005. Proceedings. 2005 IEEE* (pp. 523-528).

[19] Tabachnick, B.G., & Fidell, L.S. (2013). *Using multivariate statistics.* Pearson.

[20] Altenbuchinger M, , Weihs, A ,, Quackenbush, J, et al. Gaussian and mixed graphical models as (multi-)omics data analysis tools . *Biochim Biophys Acta Gene Regul Mech*, 2020, ; 1863 , : 9441

[21] Feature Scaling Standardization vs. Normalization. Available online: https://www.analyticsvidhya.com /blog/2020/04 /feature-scaling-machine-learning-normalization-standardization/ (accessed on 21 November 2021).

[22] Decision Tree Algorithm, Explained—KDnuggets. Available online: https://www.kdnuggets.com /2020/01/decision-tree-algorithm-explained.html (accessed on 2 March 2022).

[23] K-Nearest Neighbor (KNN) Algorithm for Machine Learning—Javatpoint. Available online: https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning (accessed on 2 March 2022).

[24] Binarize Label Hivemall User Manual. Available online: https://hivemall.apache.org/userguide/ft_en gineering /binarize.html (accessed on 2 March 2022).

[25] De Kerf, T.; Gladines, J.; Sels, S.; Vanlanduit, S. Oil Spill Detection Using Machine Learning and Infrared Images. *Remote Sens.* 2020, *12*, 4090.

[26] IEEE Xplore Full-Text PDF. Available online: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber= 9226415 (accessed on 30 March 2022).

[27] Thudumu, S., Branch, P., Jin, J. et al. A comprehensive survey of anomaly detection techniques for high dimensional big data. J Big Data 7, 42 (2020). https://doi.org/10.1186/s40537-020-00320-x

[28] Gupta, Upasana, Singh, Vaishali & Goyal, Dinesh (2023) Highly secure intelligent computer data detection of anomalies, Journal of Discrete Mathematical Sciences and Cryptography, 26:3, 875-884, DOI: 10.47974/JDMSC-1767.