JOURNAL OF ENGINEERING RESEARCH & SCIENCES

JENRS

Editorial

The integration of advanced computational techniques, emerging communication paradigms, and innovative modeling frameworks is reshaping the research landscape across a wide spectrum of disciplines. From healthcare and education to industrial diagnostics and telecommunications, the presented studies demonstrate the transformative impact of machine learning, AI, and data analytics in resolving real-world challenges. These research contributions underscore not only the necessity of adopting sophisticated models but also the importance of aligning technology with human-centered design, ethical considerations, and systemic robustness.

Dealing with imbalanced datasets remains a significant obstacle in the deployment of predictive modeling, especially in regression problems involving real-time continuous data. This study undertakes a comparative evaluation of traditional machine learning algorithms and neural network architectures, with and without dimensionality reduction, using a satellite-based air pollution dataset. Principal Component Analysis (PCA) is applied for feature selection, and five regression techniques—Multilinear, Ridge, Lasso, Elastic Net, and SVM—are tested. The findings demonstrate that deep neural networks offer superior performance over conventional models, especially under skewed data distributions. By independently modeling and testing each approach, this research provides valuable insights into the role of neural networks in enhancing prediction accuracy amidst data imbalance [1].

The convergence of 5G networks, Internet of Things (IoT), and artificial intelligence is heralding a new era of intelligent connectivity, particularly in the context of wireless private networks. This work introduces a conceptual business model tailored for AI-enabled 5G-IoT private networks, aiming to guide operators in capitalizing on this disruptive shift. The framework addresses the economic and strategic considerations necessary for maintaining competitive edge while fostering sustainable innovation. With the mobile services sector undergoing rapid transformation, the study encourages a re-evaluation of existing business paradigms to support the growth of intelligent and adaptive network services [2].

Autonomous vehicles are becoming central to the conversation around the future of mobility, bolstered by AI, sensor integration, and V2X communications. This comprehensive analysis explores the infrastructure requirements, security vulnerabilities, regulatory landscapes, and social implications of autonomous vehicles, drawing on case studies of industry leaders like Tesla, Waymo, and General Motors. A comparative analysis with drones offers insights into shared cybersecurity risks, including GPS spoofing and unauthorized access. The study underscores the urgent need for regulatory clarity, robust cybersecurity protocols, and public trust to enable the safe deployment of AVs. It presents a forward-looking assessment of how AVs can revolutionize transportation while remaining secure and sustainable [3].

Kernel methods, a foundational concept in machine learning, are revisited in this extended exploration through the lens of high-dimensional geometry and clustering. The paper revisits the k-means clustering algorithm, tracing its classical origins and detailing its adaptation using Reproducing Kernel Hilbert Space (RKHS). The kernel trick facilitates complex operations without explicit embedding, enhancing computational efficiency. By refining initialization strategies and allowing quantification of target function improvements, the paper offers a more robust and interpretable version of kernelized k-means. This methodological advancement strengthens the applicability of clustering algorithms in various analytical domains [4].

Fleet maintenance and reliability analysis depend on accurately modeling system failures and evaluating heterogeneity across units. This research critiques existing methods for estimating mean times between failures (MTBFs) by highlighting the discrepancies that arise when comparisons are made at different time points. The proposed method introduces a unified reference process to estimate MTBFs across systems simultaneously, resulting in more consistent and robust evaluations. Through the analysis of three datasets, the superiority of the proposed

approach is validated, offering a practical solution for reliability engineers and fleet managers seeking improved fault diagnosis and prediction [5].

Visualizing biclusters derived from gene expression data presents a unique challenge due to their overlapping and bi-dimensional nature. This work surveys and evaluates various visualization strategies that enable researchers to analyze multiple biclusters simultaneously. The paper emphasizes how meaningful interpretation of biclustering results is crucial for identifying gene interactions across conditions. By categorizing and assessing different techniques, the study aids biologists and data scientists in selecting the most appropriate visualization tools for complex genomic datasets, ultimately enhancing their ability to derive actionable biological insights [6].

Mental health monitoring among university students, especially in demanding disciplines like engineering, is an area of growing importance. This study expands on prior work by incorporating physiological and eye-tracking data into a framework for assessing emotional well-being. By analyzing baseline data, the study uncovers correlations between emotional states and physiological signals—such as the link between fear and physical activity or happiness and electrodermal activity. Temporal trends indicate emotional spikes during evening hours, supporting the need for context-aware mental health interventions. The framework sets the stage for real-time, personalized support systems that promote student wellness and academic performance through non-invasive monitoring [7].

These diverse yet interconnected studies reflect the depth and breadth of contemporary research aimed at harnessing the power of data, algorithms, and interdisciplinary thinking. Whether enhancing system resilience, decoding biological patterns, or supporting mental health, each contribution underscores a collective vision for a smarter, more equitable, and technologically empowered future.

References:

[1] S.K. Mondal, A. Sen, "An Integrated Approach to Manage Imbalanced Datasets using PCA with Neural Networks," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 1–12, 2024, doi:10.55708/js0310001.

[2] L. Banda, "Conceptual Business Model Framework for AI-based Private 5G-IoT Networks," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 13–20, 2024, doi:10.55708/js0310002.

[3] V. Tiwari, "Navigating the Autonomous Era: A Detailed Survey of Driverless Cars," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 21–36, 2024, doi:10.55708/js0310003.

[4] B. Jürgen Falkowski, "On a Kernel k-Means Algorithm," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 37–43, 2024, doi:10.55708/js0310004.

[5] R. Jiang, K. Zhang, X. Xu, Y. Cao, "Evaluation of equivalent acceleration factors of repairable systems in a fleet: a process-average-based approach," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 44–54, 2024, doi:10.55708/js0310005.

[6] H. Aouabed, M. Elloumi, F. Algarni, "Biclustering Results Visualization of Gene Expression Data: A Review," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 54–68, 2024, doi:10.55708/js0310006.

[7] Y. Liu, A. Tofighi Zavareh, B. Zoghi, "Enhancing Mental Health Support in Engineering Education with Machine Learning and Eye-Tracking," *Journal of Engineering Research and Sciences*, vol. 3, no. 10, pp. 69–75, 2024, doi:10.55708/js0310007.

Editor-in-chief

Prof. Paul Andrew

JOURNAL OF ENGINEERING RESEARCH AND SCIENCES

CONTENTS

An Integrated Approach to Manage Imbalanced Datasets using PCA with Neural Networks

Swarup Kumar Mondal* , **Anindya Sen**

Department of Electronics and Communication Engineering, Heritage Institute of Technology, Kolkata, 700107, India
*Corresponding author: Swarup Kumar Mondal, Kolkata 700107, swarup.kumarmondal.ece24@heritageit.edu.in

ABSTRACT: Imbalanced dataset handling in real time is one of the most challenging tasks in predictive modelling. This work handles the critical issues arising in imbalanced dataset with implementation of artificial neural network and deep neural network architecture. The usual machine learning algorithms fails to achieve desired throughput with certain input circumstances due to mismatched class ratios in the sample dataset. Dealing with imbalanced dataset leads to performance degradation and interpretability issue in traditional ML architectures. For regression tasks, where the target variable is continuous, the skewed data distribution is major issue. In this study, we have investigated a detailed comparison of traditional ML algorithms and neural networks with dimensionality reduction method to overcome this problem. Principle component analysis has been used for feature selection and analysis on real time satellite-based air pollution dataset. Five regression algorithms Multilinear, Ridge, Lasso, Elastic Net and SVM regression is combined with PCA and non PCA to interpret the outcome. To address unbalanced datasets in real-time, deep neural networks and artificial neural network architectures have been developed. Each model's experiments and mathematical modelling is done independently. The Deep neural network is superior compared to other conventional models for performance measures of target variable in imbalanced datasets.

KEYWORDS: Imbalanced data, Regression, Deep Neural Network, Artificial Neural Network, Support Vector Machine

1. Introduction

In the domain of artificial intelligence (AI), the persistent challenge of imbalanced datasets remains a focal point. Accuracy, a prime indicator of model performance is primarily dependent on the balance of the data set. This issue is of paramount importance in practical applications like fraud detection [1], medical diagnosis [2], remote sensing [3], engineering [4] and anomaly detection [5], where the class of interest is typically a minority within the dataset. An uneven distribution of inputs and outputs (classes) in an imbalanced dataset is defined as one class, called the minority class, having substantially fewer instances than the other classes, called the majority class. Due to the uneven class ratios in the dataset, handling unbalanced datasets presents a challenge. In an out of balanced dataset, the majority class may make up a large portion of the training dataset, while the minority class is

underrepresented in the dataset. The problem with a model trained on such out of balanced data set is that the model progressively learns to achieve high accuracy by consistently predicting the majority class, even if recognizing the minority class is equal or more important when applying the model to a real-world scenario.

Many machine learning approaches have been widely applied in a variety of applications. Machine learning models such as Multilinear Regression (MLR), Ridge Regression (RR), Lasso Regression (LR) and Elastic Net Regression (ELR) are mainly used for regression task in diverse ways. However, the traditional regression algorithms are not always beneficial to meet the requirement of large-scale raw datasets.

One of the most helpful algorithms that aims to identify underlying links in a batch of data by simulating

how the human brain functions is the neural network. Neural networks are one of the most effective methods for fitting various types of stationary and non-stationary data. A degree of balance in the dataset is essential for neural networks to function well. However, the conventional assumption of a balanced data becomes untenable for certain scenarios, where the minority class is significantly underrepresented. This imbalance leads neural networks to exhibit negative performance, particularly in accurate prediction of minority classes. The issue of data imbalance can also be addressed via deep learning synaptic models, which have been shown to considerably raise performance benchmarks in both regression and classification tasks. Neural networks topologies keep changing and getting better in 21st century, that present an opportunity to enhance the computational modeling capabilities and enabling more accurate predictions in various data-driven use cases [6].

1.1. Hypothesis

In an imbalanced stationary dataset, the neural network-based architecture works more efficiently and produce better performance accuracy over multiple regression models (MLR, RR, LR, ELR).

1.2. Objective

This research implements a novel way to apply neural networks to imbalanced real-time satellite datasets. Multiple regression model is trained for analyzing the performance for real time satellite data based on Air Quality Index (AQI) prediction. The same dataset is used for the implementation of Multilinear Regression (MLR), Ridge Regression (RR), Lasso Regression (LR), Elastic Net Regression (ELR), and Support Vector Machine Regression (SVM). In addition, the dataset is processed using Principal Component Analysis (PCA). Essentially, two neural network architectures- the Artificial Neural Network (ANN) and the Deep Neural Network (DNN) are compared with four traditional regression models and SVM. After comparison, the proposed method produced promising results which shows how to analyze the imbalanced dataset in an effective manner. Experiments are presented to demonstrate how the neural network approach efficiently improves the model accuracy and R-squared score (R2).

2. Literature Review

Earlier works which specified different approaches and methods for theoretical and practical implementation as well as handling the imbalanced dataset are discussed below:

A training procedure for Multilayer Perceptron (MLPs) is implemented using an imbalanced dataset. This article [3] uses a remote-sensing dataset that includes agricultural classes including potatoes, carrots, wheat, sugar beets, and stubble. The suggested method seeks to increase the stability of classification findings and accelerate training by an average of 41.5 times.

In order to solve Haberman's surviving unbalanced data set challenges, a modified learning method [6] for ANN is presented. The artificial neural network's output layer uses Particle Swarm Optimization (PSO) as part of its technique to optimize the step function's decision boundary. The study demonstrates increase in the average Geometric-Mean Test of classifier performance (80.16 for training and 70.47 for testing).

Three image data sets and five document data sets with different degrees of imbalance are used in an experiment [7]. It proposed a loss function called mean false error (MFE) and its enhanced counterpart, mean squared false error (MSFE), for training DNN on unbalanced data sets. Theoretical analysis is empirically confirmed, and experiments and comparisons demonstrating the superiority of the proposed approach over traditional methods in classifying imbalanced data sets on deep neural networks demonstrate the effectiveness of the proposed methods in extremely imbalanced data sets.

The creation of a system to simulate and predict maize and soybean yields on a county-by-county basis in the American Midwest and Great Plains using Artificial Neural Networks (ANN) has been studied [8]. It utilized multi-temporal remote sensing images to derive NDVI values, which characterized the entire growing process. The methodology employed a feed-forward multi-layer perceptron (MLP) neural network for learning and the SCE-UA method for training the NN. The outcomes showed that multivariate linear regression (MLR) is 20% inferior than the ANN.

The purpose of gathering seven severely unbalanced data sets [9] is to assess how well various Support Vector Machine (SVM) modelling techniques work. Various "rebalance" procedures, including cost-sensitive learning and over and under sampling, were included into SVM modelling to tackle the problem of class imbalance. It introduced GSVM-RU algorithm, which comes up as a state of art approach with 85.2 G-Mean, 91.4 AUC- ROC, 66.5 F Measure and 181 Efficiency.

The work [10] focuses on the unique problem of Deep Imbalanced Regression (DIR) and uses large-scale datasets from computer vision, natural language processing, and healthcare. Feature distribution smoothing (FDS) and label distribution smoothing (LDS) are two efficient

techniques that the study suggests using to handle unbalanced data with continuous objectives. The MAE and G-Mean improved significantly ranged from 0.1 to 2.7.

A large-scale dataset is utilized including face attribute classification and edge detection tasks, as well as controlled class imbalance in the MNIST digit classification. A novel method [11] for learning deep feature embeddings that effectively handle imbalanced data classification. The proposed approach involves quintuplet sampling and a triple-header hinge loss to enforce relationships during feature learning. The LMLE-kNN has outperformed by a large margin with a mean per class accuracy of 84 as compared to other traditional method.

To confirm the notion that models trained using Dense Loss perform better in underrepresented areas of the dataset than models built with a conventional training approach, synthetic datasets [12] with varied features, such as heavy-tailed datasets, are utilized. The findings demonstrated that MLP with Dense Loss performs better than MLP without Dense Loss, with an average Root Mean Square Value (RMSE) ranging from 1.21 to 7.02.

According to a study [13] on small and medium-sized enterprises (SMEs) in Malaysia, such as those in the transportation, storage, catering, lodging, and hotel industries, the Synthetic Minority Oversampling Technique (SMOTE) is effective in improving the classification accuracy of logistic regression models when the data are highly unbalanced. The study used a dataset of 601 failed and 26,284 failing SMEs between 1999 and 2013. The findings demonstrated that, with 57.23% sensitivity and 58.83% specificity, the SMOTE logistic regression technique produced superior metrics when compared to classical logistic regression.

SMOGN algorithm [14] combines two oversampling techniques knows as SMOTE and gaussian noise. This algorithm is used as a preprocessing solution for unbalanced regression problems with the aim of improving the performance of regression algorithms. SMOGN performs exceptionally well with multivariate adaptive regression spline (MARS) and random forest (RF) learners, demonstrating enhanced recall without appreciable loss of precision, according to tests on 20 distinct regression datasets.

3. Materials and Methods

This section introduces the imbalanced stationary data regression algorithms and the experimental data sets used in this study.

3.1. Dataset

The dataset consists of the satellite records of greenhouse gases (GHGs) emission in India. The initial dataset consists of two million five hundred eighty nine thousand and eighty three sets of reviews. It possesses a total of fifteen features as Stationid, PM2.5, PM10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, Toluene, Xylene, AQI and AQI_Bucket. The raw data belongs to a stationary imbalanced category. A preprocessed version of the data having five thousand and ninety two samples is utilized in this study. The dataset used in this study contains no missing features. The original dataset is open source and can be downloaded here: https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india?select=station_hour.csv

3.2. Operating system and software

The analytical code implementation and execution have been conducted on a 64-bit operating system with an x64-based processor system type, complemented by 8GB of RAM. Version 6.3.0 of Jupyter Notebook is utilized as an interactive programming environment. In addition, the Python script has been run on the Google Colab cloud platform, which has an Intel(R) Xeon(R) CPU operating at 2.30 GHz, 12.7 GB of RAM, and 108 GB of disc space.

3.3. Data Preprocessing

The quality of the underlying data is the primary determinant of the success of visualizations and the construction of efficient machine learning (ML) models. Preprocessing is essential for increasing data quality since it reduces noise, which boosts ML systems' processing speed and capacity for generalization. Two prevalent issues encountered in data extraction and monitoring applications are outliers and missing data. Different techniques have been employed to identify outlier values and handle not-a-number (NAN) values. Table 1 shows the visual representation of distribution of missing values across different features of the dataset. Notably, Xylene emerges as the feature with the highest number of missing values, while CO exhibits the least. To guarantee that the significance of the variables is not impacted by changes in their ranges or units, all missing values have been removed from the dataset and the standard scaler normalization process has been executed.

Table 1: Missing Values of Features.

Features	Total Missing Values
PM$_{2.5}$	647,689
PM$_{10}$	1,119,252
NO	553,711
NO$_2$	528,973
NO$_x$	490,808

NH$_3$	1,236,618
CO	499,302
SO$_2$	742,737
O$_3$	725,973
Benzene	861,579
Toluene	1,042,366
Xylene	2,075,104
AQI	570,190
AQI Bucket	570,190

3.4. Feature Selection

To select significant features, the correlation of the AQI with other features has been analyzed. Table 2 displays the precise correlation values between the AQI and each contaminant in the dataset. It is evident that the majority of the features have very little correlation. The ideal number of input variables for the creation of machine learning models has been found using a feature selection method based on Principle Component Analysis (PCA). To improve computing efficiency and accuracy, the intended objective is to project a d-dimensional dataset onto a (k)-dimensional subspace in order to minimize its dimensions.

Table 2: Correlation between AQI and Pollutants.

Sl. No.	Features	Correlation value	Sl. No	Features	Correlation value
1	PM$_{2.5}$	0.786344	7	CO	0.432508
2	PM$_{10}$	0.757663	8	SO$_2$	0.135806
3	NO	0.288469	9	O$_3$	0.094589
4	NO$_2$	0.441733	10	Benzene	0.125644
5	NO$_x$	0.426584	11	Toluene	0.169872
6	NH$_3$	0.283593	12	Xylene	0.090680

3.5. Experiment Performed

3.5.1. Proposed Architecture

Figure 1 illustrates the workflow of our proposed architecture. It begins with collecting dataset and data preprocessing. After that, it checks if there are any missing values present or not in the dataset. It there are no missing values then the non-redundant dataset go through the exploratory data analysis process. The next steps involve applying the features and target value into various ML and neural network architecture. If the result is unstable, and poor then the input goes through PCA with 2, 3 and 5 components to enhance the model stability and handle the imbalanced real time dataset. After PCA, it selects the optimal features and target values to perform data splitting and perform regression to produce the improved results. This approach enhanced the model stability, interpretability and able to overcome the issue of handling imbalanced dataset using standard regression models.

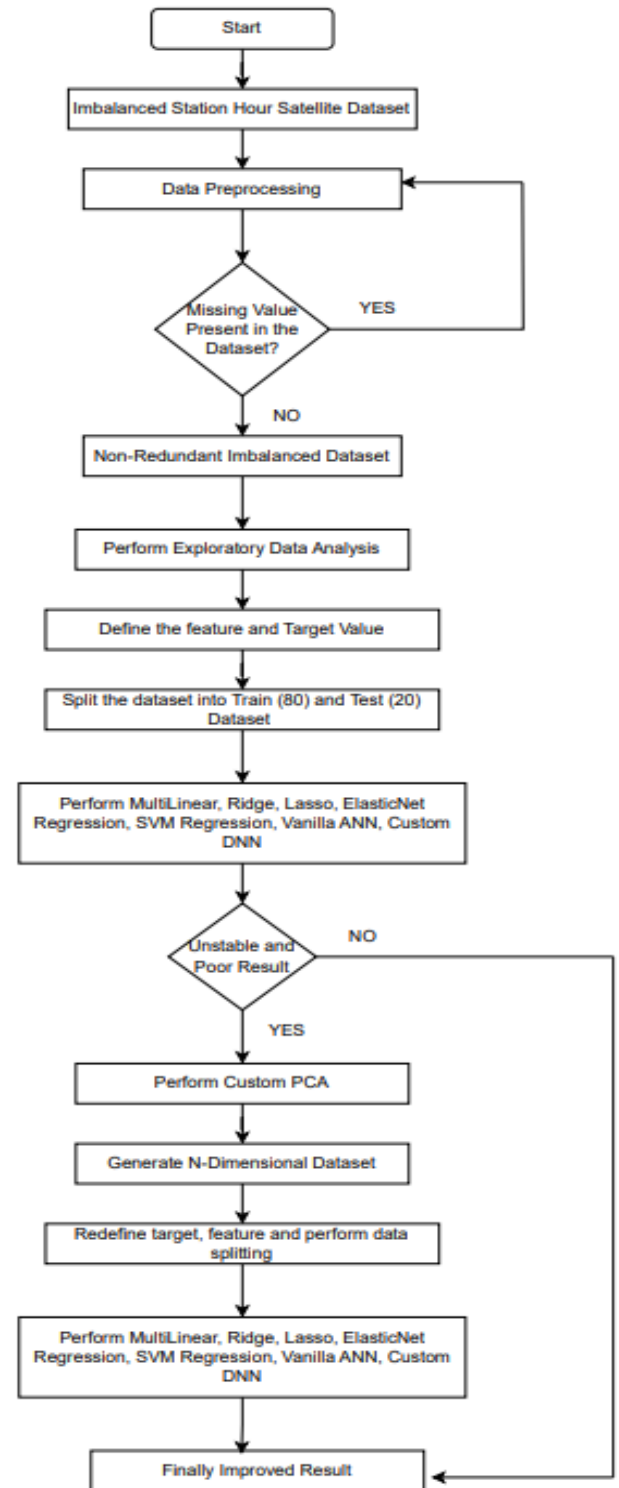Figure 1: Process Workflow of the Proposed Architecture

3.6. Algorithm

Algorithm: PCA Combined Neural Network

Input: D → Complete dataset (Station hour Satellite dataset)

Output: Predicted AQI Value

3.7. Steps

1) Load and Clean Data

Load Dataset D, handle missing values and make

the dataset non-redundant.

2) Perform Initial EDA and Model Evaluation

I. Define features and target variable.

II. Split dataset into training (80%) and testing (20%) sets.

III. Apply the following 7 models on the imbalanced dataset: MLR, RR, LR, ELR, SVM, Vanilla ANN and Custom DNN

IV. Evaluate performance metrics and check if the results are unstable or not. If the results are not good then go to step 3.

3) Apply Custom PCA

I. Standardize the dataset: $X_{std} = X - \mu/\sigma$, where X is the original data, μ is the mean and σ is the standard deviation statistical error that separates an observation from its predicted value.

II. Compute Covariance Matrix: $\Sigma = \frac{1}{n-1} X_{std}^T X_{std}$, where Σ is the covariance matrix, X_{std}^T is the transpose of the standardized data, and n is the number of observations.

III. Obtain Eigenvectors and Eigenvalues: $\Sigma v = \lambda v$, where v is the eigenvectors and λ is the eigenvalues.

IV. Select Top k Eigenvectors: Sort eigenvalues in descending order and select the top k eigenvectors corresponding to the largest eigenvalues. Construct the projection matrix W from these eigenvectors.

V. Transform Data: $Y = X_{std}W$, Where Y is the transformed data in the new k-dimensional space.

4) Perform Regression with PCA optimized data:

I. Split PCA-transformed data into training (80%) and testing (20%) sets.

II. Apply the following 7 models in the dataset Y with 2 component PCA, 3 component PCA and 5 component PCA: MLR, RR, LR, ELR, SVM, Vanilla ANN and Custom DNN

III. Evaluate performance for each configuration

5) Compare Results: Compare performance metrics from initial and PCA-transformed datasets to get the improved results.

3.7.1. Proposed Model

For this work, five regression model, and two neural network-based architecture models are introduced.

3.7.1.1. Multilinear Regression

Multilinear regression (MLR) is an analytical method of predicting the value of a specific metric by taking into account several independent variables. This algorithm is used to find the optimal straight line to predict the AQI value for various parameters. First, we define the independent features as x1, x2, ………., x10. Then, we have defined the dependent feature as y i.e., AQI. It can be mathematically represented via equation 1.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon \qquad (1)$$

For nth observation, β_0 is a constant term also known as y-intercept and β_n is the slope coefficient for each explanatory variable. Parameter ϵ stands for the statistical error that separates an observation from its predicted value.

3.7.1.2. Ridge, Lasso and Elastic Net Regression

It is common for real-time unbalanced datasets to have a significantly larger number of input variables than observations. With many predictors, fitting the full model without penalization will result in large prediction intervals. For that reason, we have used three model tuning regression method i.e., Ridge Regression (RR), Lasso Regression (LR) and Elastic Net Regression (ELR) to analyze the problem better. RR constitutes a model refinement technique employed for the analysis of datasets afflicted by multicollinearity [15]. Employing L2 regularization, this method addresses situations where the occurrence of multicollinearity imparts bias to least-squares estimates and induces elevated variances. The mathematical representation of RR is in equation 2.

$$RRL_2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{P} \beta_j^2 \qquad (2)$$

Here, the L2 term is equivalent to the square of the coefficient magnitudes (βj) and the regularisation penalty is denoted by λ. As we increase the value of λ this constraint causes the value of the coefficient to tend towards zero [16]. By applying a penalty proportional to the absolute value of the magnitude of the coefficients, Lasso regression carries out L1 regularisation (equation 3). Certain coefficients may become zero and be removed from the model as a result of this kind of regularisation, producing sparse models with few coefficients.

$$LRL_1 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{P} |\beta_j| \qquad (3)$$

To improve the model prediction rate, both regularization of lasso and ridge are combined to produce the elastic net model loss. The collinearity coefficient is difficult to eradicate, according to the fundamental rationale behind. In this study, the Grid Search approach (GS) for hyperparameter tweaking has been employed to enhance the model's performance. This research makes use of the Scikit-Learn class GridSearchCV. The GridSearchCV evaluates, all possible combinations of parameter values and finally, the best parameter combination is retained. After tuning the optimal λ value is 1, 0.0001, and 0.0001, for RR, LR, and ELR, respectively.

3.7.1.3. Support Vector Machine (SVM)

SVM divides data into multiple classes by locating a hyperplane in a high-dimensional space [17]. The primary objective is to optimize the margin, which is the space between the hyperplane and the closest support vectors, while simultaneously minimizing the error SVM uses a technique called the kernel trick to translate the feature space into a higher-dimensional space in order to improve data point separation when dealing with non-linearly separable data [18], [19]. The kernel functions used in the SVM model are linear and Radial Basis Function (RBF). The hyperparameter C, and epsilon, are taken as 5, and 1, respectively after tuning for the proposed model. C is the regularization parameter that controls the trade-off between achieving a low training and testing error while maximizing the margin. Epsilon is the width of the epsilon-insensitive tube in SVM Regression. It defines a margin of tolerance where no penalty is given to errors.

3.7.1.4. Artificial Neural Network (ANN)

The best model for automatically identifying and simulating intricate non-linear correlations between the "output" (i.e. AQI) and the "inputs" (i.e. NO, NO2, NH3 etc.) of the network is the artificial neural network (ANN), which can also take into account all potential interactions between the input variables [20]. As imbalanced dataset contains a large proportions of skewed classes so conventional regression method fails to recognise the non-linear relationship between inputs and outputs. The architectural depiction of the ANN model employed in this study is displayed in Figure. 2.
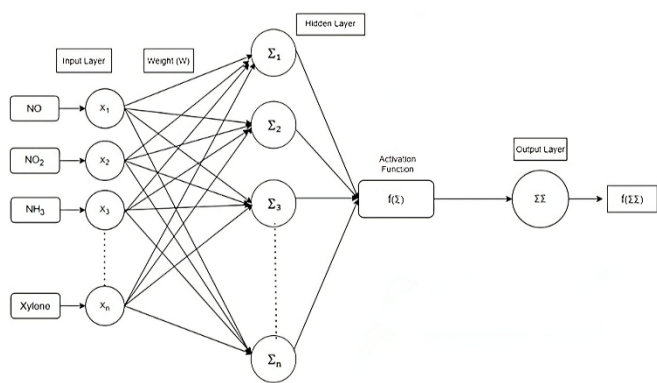

Figure 2: Visual Representation of ANN Architecture

The ANN consists of one input, one hidden and one output layer. The input features are represented as X1, X2, X3,...., Xn. The synaptic weights between the input and hidden layer are defined as W1, W2,...Wn. The biases are taken as b0 and b1 for hidden and output layer respectively. Σ denotes each node takes the weighted sum of its inputs, and passes it through a non-linear activation function. The mathematical equation of tanh is

represented via equation 4.

$$f(\Sigma) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}} \qquad (4)$$

The AQI value is then predicted by feeding the hidden layer's output into the output layer. To reduce the error rate during backpropagation, the gradient descent is calculated and used to modify the weights and biases. The ANN model has been analysed combined with PCA for 2, 3 and 5 components respectively to increase the model interpretability and computational efficiency.

3.7.1.5. Deep Neural Network (DNN)

DNN refers to a multilayer perceptron model that has more than one hidden layer. When it comes to assessing the AQI value, DNN is a more potent and reliable neural network model than ANN. DNN exhibits of layer wise feature extraction methodology and combine low level feature to generate high level features [21], [22]. Figure. 3 shows the visual representation of DNN model. It is composed of one input layer with 10 neurons, two hidden layer (i.e., 256 and 512 neurons respectively) and one output layer. Weights and biases are initialised to the proper values at each layer of the network. Rectified Linear Unit (ReLu) activation function (represented by equation 5) and linear function are utilised at the hidden layer and output layer, respectively. To achieve state of the art performance, the model has then been evaluated using a combination of PCA and DNN.
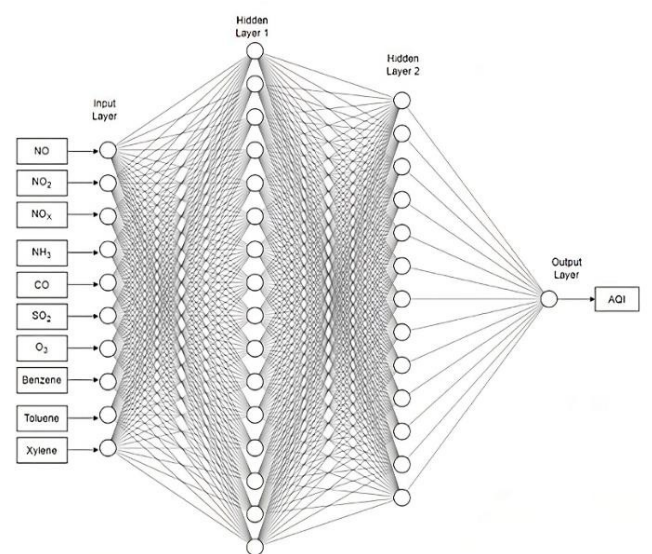
$$\text{ReLu}(z) = \max(0, z) \qquad (5)$$

Figure. 3: Deep Neural Network Model Structure.

4. Results

The models for forecasting the AQI level based on different atmospheric pollution characteristics are

empirically evaluated in this section. Metrics such as mean absolute error (MAE), mean squared error (MSE), root mean square error (RMSE), mean absolute percentage error (MAPE), and coefficient of determination (R2) have been used to evaluate the model. Every model has been simultaneously trained, examined, and integrated using PCA and non-PCA analysis. For RR, LR, ELR, and SVM regression, the Grid Search method has been used.

Table 3: Comparison of Regression model results without PCA.

Model	Non PCA			
	Normal		GS	
	MSE	R2	MSE	R2
MLR	440.823	0.752		
RR	439.022	0.753	439.022	0.753
LR	446.132	0.749	439.294	0.753
ELR	510.646	0.712	439.148	0.753

The performance characteristics of a standard regression model with and without Principal Component Analysis (PCA) integration are shown in Tables 3 and 4, respectively. We can clearly see that RR has outperformed as compared to other model in terms of MSE and R2 score.

MLR and LR has slightly degraded performance in both normal and grid search method (GS) as compared to RR. After using PCA, it is shown that using five principal components produces outcomes that are almost similar to those that come from non-PCA analysis.

The association between the independent variables and AQI is explained in great detail by the trained MLR model. Equation 6 provides a statistical illustration of the concept represented in equation 1. Because LR regression employs the L1 norm, some of the coefficients are absolutely zero. This shows that the relevant elements have been effectively excluded from the model, rendering them useless in predicting the target variable. This increases model interpretability while also possibly improving prediction performance by decreasing overfitting.

$$MLR(Y_{AQI}) = [4.6, 27.41, -18.07, 10.84, 4.02, -0.21, 18.98,$$
$$4.85, 1.61, -0.27]_{1x10} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{10} \end{bmatrix}_{10x1} + 92.57 \quad (6)$$

The equation 6 represents the coefficient matrix (β_1= 4.6, β_2 = 27.41, ...,β_{10} = −0.27), independent features from NO to Toluene (X1, X2,...,X10)and intercept value(β_0 = 92.57) for MLR. The values from β_0 to β_{10} are experimentally obtained corresponding to the MLR entry of table 3 without PCA. Similarly, equation 7, equation 8

and equation 9 defines the coefficient matrix, independent features and intercept values used for RR, LR and ELR model respectively. The statistical equations from 7 to 9 are generated with all the features without using PCA. The final combined PCA model equations along with coefficient matrix and intercept can also be generated in the similar way.

$$RR(Y_{AQI}) = [2.01, 21.07, -10.47, 10.87, 3.99, -0.19, 18.95, 4.81,$$
$$1.62, -0.27]_{1x10} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{10} \end{bmatrix}_{10x1} + 92.57 \quad (7)$$

$$LR(Y_{AQI}) = [-1.47e-02, 1.15e+01, 0, 1.11e+01, 3.28, 0,$$
$$1.8e+01, 4.46, 8.99e-01, 0]_{1x10} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{10} \end{bmatrix}_{10x1} + 92.57 \quad (8)$$

$$ELR(Y_{AQI}) = [-1.41, 5.07, 3.23, 9.27, 4.36, 2.07, 12.3, 3.81, 2,$$
$$0.97]_{1x10} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{10} \end{bmatrix}_{10x1} + 92.57 \quad (9)$$

As compared to traditional regression model SVM works better than in terms of R2 score. Table 5 describes that RBF with GS approach has the lowest RMSE and R2 score as compared to linear and normal RBF. With implementation of combined PCA and SVM, the model able to able to outperformed in each type of kernel by high margin. In regression, accuracy refers to the model's ability to forecast the percentage difference between the actual and estimated values. Table 5 compares all three types of SVM kernels to determine the overall best accuracy.

The derived equations for the linear and RBF kernels are provided in equations 10 and 11, respectively. The results of the AQI prediction are obtained using these equations. For linear SVM, positive coefficients indicate that a factor has a positive effect on predicted AQI, while negative coefficients indicate a negative effect. The intercept term represents the baseline AQI value. In contrast, with the RBF-kernelized SVM, the dual coefficients indicate the importance of the support vector in defining the decision region, with positive and negative coefficients indicating the direction of the effect on the predicted AQI.

Table 4: Comparison of Regression Model results with PCA

Model Name	PCA				Model				Type			
	2 PCA Component				3 PCA Component				5 PCA Component			
	Normal		GS		Normal		GS		Normal		GS	
	MSE	R2	MSE	R2	MSE	R2	MSE	R2	MSE	R2	MSE	R2
MLR	546.891	0.692			461.973	0.740			450.89	0.746		
RR	546.896	0.692	546.997	0.692	461.979	0.740	462.039	0.74	450.90	0.746	450.97	0.746
LR	549.518	0.691	546.909	0.692	467.196	0.737	462.006	0.74	457.81	0.742	450.89	0.746
ELR	596.489	0.664	547.008	0.692	527.029	0.703	461.988	0.74	520.04	0.707	451.07	0.746

Table 5: SVM Regression Results with PCA and NON PCA.

Kernel Type	NON PCA			PCA			Model			Type		
				2 PCA Component			3 PCA Component			5 PCA Component		
	RMSE	R2	Accuracy	RMSE	R2	Accuracy	RMSE	R2	Accuracy	RMSE	R2	Accuracy
Linear	21.476	0.725		23.619	0.686		21.667	0.736		21.383	0.742	
RBF	20.867	0.741	80.86	20.540	0.762	80.16	20.351	0.767	80.523	20.217	0.770	81.00
RBF_GS	20.045	0.761		20.186	0.770		19.541	0.785		18.984	0.797	

Table 6: Comparison of ANN Model Results.

Metrics	NON PCA	PCA Model Type		
		2 PCA Component	3 PCA Component	5 PCA Component
MAE	14.616	15.025	14.979	15.235
MSE	390.485	408.578	398.932	427.794
RMSE	19.760	20.213	19.973	20.683
MAPE	0.198	0.206	0.205	0.207
R2	0.757	0.746	0.752	0.734
Accuracy	80.152	79.341	79.442	79.206

After training using linear SVM equation 10 shows the final weight matrix (W1= -0.3, W2= 0.52,……, W10= 0.09), input feature matrix (X1, X2,...,X10) and bias as 9.58 for computing the AQI output. The RBF kernel based SVM represented via equation 11 analyze for the weights consisting between -1 and 1, bias term (113.14), input feature vectors from X1 to X10 and each support vectors $(K(X,X_1), K(X,X_2), \ldots\ldots\ldots, K(X,X_{10}))$ passing through X1 to X10.

F(AQI) = [-0.30, 0.52, 0.03, 1.99, 4.41, 0.21, 0.59, 1.28, 0.18,

$$0.09]_{1 \times 10} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{10} \end{bmatrix}_{10 \times 1} + 9.58 \qquad (10)$$

Table 6 shows the result after training the imbalanced dataset using ANN architecture. ANN outperformed all of the earlier models in terms of strength and performance. It achieved lowest RMSE and MSE as compared to SVM, RR, LR, ELR and MLR. ANN Model

Table 7: Comparison of DNN Network Results

Metrics	NON PCA	PCA Model Type		
		2 PCA Component	3 PCA Component	5 PCA Component
MAE	13.08907	15.67781	17.27715	15.70377
MSE	334.054	438.4483	583.0541	469.349
RMSE	18.27715	20.93916	24.14651	21.66446
MAPE	0.172188	0.213611	0.229836	0.208632
R2	0.792717	0.72794	0.638211	0.708766
Accuracy	82.78118	78.63886	77.01642	79.13676

obtained with ten inputs, one hidden layer with one twenty-eight neurons and final output layer with predicted AQI value.

$$f(AQI) = [-1\ -1\ -1\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots1\ -1\ 1]_{1 \times 10} \begin{bmatrix} K(X, X_1) \\ K(X, X_2) \\ \cdot \\ \cdot \\ \cdot \\ K(X, X_{10}) \end{bmatrix}_{10 \times 1} + 113.14 \qquad (11)$$

The DNN Model is a very sophisticated and potent neural network designed to manage unbalanced datasets. Table 7 shows that the DNN model works better than the ANN by significantly increasing the R2 score and simultaneously decreasing metrics like MSE, MAE, and RMSE. Notably, the DNN model gets an accuracy rate of 82.78%, outperforming all other models in the comparison

A lower MSE and a higher R² score indicate that the DNN model also performs even better than conventional regression models. The DNN model is more successful without requiring feature extraction, as seen by its lower RMSE and greater accuracy when compared to the SVM model using five component PCA. During training, ANN and DNN possess the ability to autonomously extract meaningful features from raw data. They can adapt to high-dimensional input spaces without explicit feature extraction. ANN and DNN model with combined PCA result has been shown in table 6 and table 7 respectively. However, complicated connections among features can be handled by the proposed DNN model without the need for PCA. This not only increases efficiency and interpretability, but also reduces training time and computational cost, making the DNN model a state-of-the-art approach for various performance metrics

DNN model takes ten neuronal inputs, and two hidden layers consisting of two fifty-six and five hundred twelve neurons respectively. The output layer involves a 512×1- dimensional weight matrix along with a linear function to derive the AQI value.

Figure 4 and 5 illustrate the rate of MSE change for the

ANN model employing 10 input features and utilizing dimensionality reduction with 2 features, respectively. It is clear that the ANN model with 10 input features converges much more quickly than the one using 2 features plus PCA because of inadequate feature representation. Similarly, Figure 6 and 7 depict the MSE convergence dynamics during the training of neural network architectures for final output layer prediction. The DNN model exhibits standard MSE performance in both training and testing datasets, while the DNN utilizing 2 principal components tends to converge prematurely, resulting in less accurate predictions for the final output layer.

Consequently, our original DNN synaptic model architecture demonstrates superior performance with robust convergence compared to alternative networks.

DNN models can converge faster and produce more accurate results without the usage of PCA. The suggested DNN model is advantageous for achieving optimal outcomes, especially in scenarios involving imbalanced datasets.
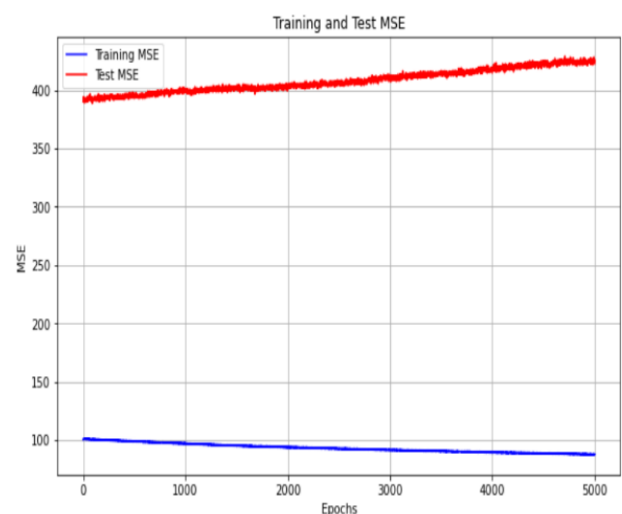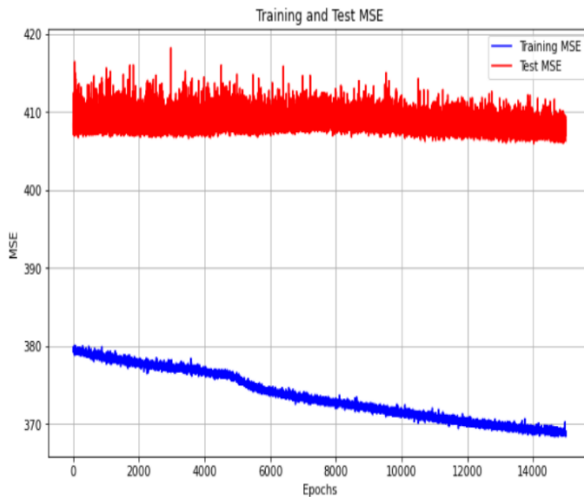

Figure 4: ANN MSE Convergence.
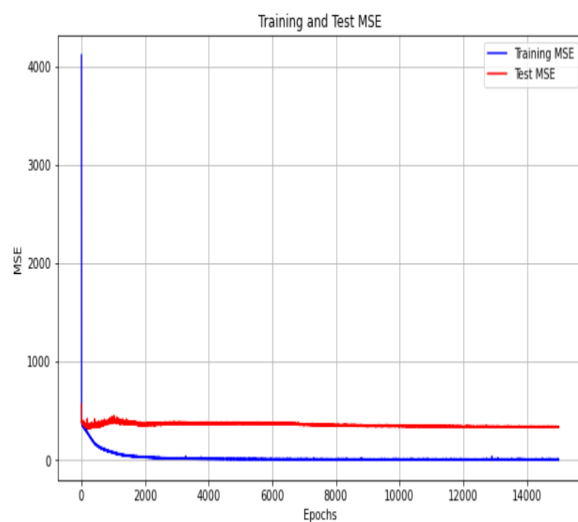
Figure 5: ANN 2PCA MSE Curve
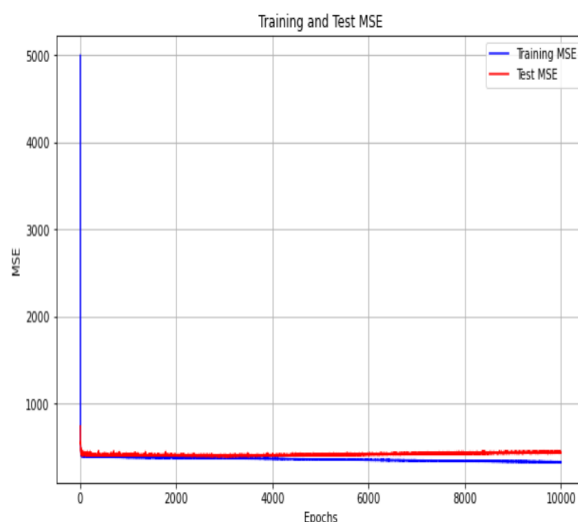

Figure 6: DNN MSE Curve

Figure 7: DNN 2PCA MSE Plot

PCA effectively reduces the dimensionality of the dataset into 2, 3, and 5 components, respectively. It identifies the most significant features by projecting the data into k-dimensional spaces that capture the most critical variance. This process streamlines the data, minimizing noise and redundancy, which makes the data more regular and structured. By introducing the most relevant components, PCA allows the DNN to operate more efficiently, avoiding overfitting and improving its ability to learn complex patterns in the data. This enhancement in data regularity enables the DNN to outperform other regression models in terms of performance and interpretability.

5. Conclusion

Though "Accuracy" is a prime indicator of model performance however its reliability depends upon the dataset. In an out of balanced dataset, the majority class makes up a large portion of the training dataset, while the minority class is underrepresented in the dataset. The problem with a model trained on this out of balanced data set is that the model learns that it can achieve high accuracy by consistently predicting the majority class, even if recognizing the minority class is equal or more important when applying the model to a real-world scenario. Dealing with real-time out of balance datasets presents substantial difficulties, especially in jobs such as anticipating the air top quality index as well as creating appropriate regression network designs. The complexity stems from the dynamic character of the environment, the variability of pollutant levels, and their geographical and temporal irregularity. In the present work, we utilize an air pollution dataset comprising various pollutants to address this challenging problem. The dataset is extensively looked into making use of exploratory information evaluation strategies to efficiently pre-process it for research study objectives. As a result of large number of parameters in the dataset, dimensionality reduction methods such as PCA are utilized. MLR, RR, LR, ELR, and SVM are the machine learning algorithms used for training and testing with the goal of predicting the air quality index (AQI). Additionally, neural network designs particularly ANN along with DNN are checked out for more evaluation.

Each model is trained with and without PCA to compare error rate and overall performance. ANN design shows durable convergence and also exceptional efficiency as contrasted to different ML design. By harnessing the power of high-level feature extraction, the DNN model is able to surpasses traditional regression techniques and even outperforms other neural network designs like ANN. This study showcases the efficacy of neural networks, particularly the DNN model that exhibits superior accuracy, lower MSE, and higher R2 scores, indicating its ability to automatically identify relevant characteristics from unprocessed data, eliminating the requirement for explicit feature extraction methods.

6. Future Work

Future research may enhance the management of imbalanced datasets by utilizing sophisticated neural architectures such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). Additionally, real time imbalance may also be improved by using ensemble methods and transfer learning. Furthermore, statistical time series models like Seasonal Autoregressive Integrated Moving Average (SARIMA), Autoregressive Integrated Moving Average (ARIMA), and Long Short-Term Memory (LSTM) can forecast trends for dangerous pollutants including CO, NO, and NO2 etc.

Acknowledgement

Conflict of Interest

The authors declare no conflict of interest.

References

[1] A. D. Pozzolo, O. Caelen, Y.A.L Borgne, S. Waterschoot, G. Bontempi, "Learned lessons in credit card fraud detection from a practitioner perspective," *Expert Syst. Appl.* 2014, 41, 4915–4928, doi: https://doi.org/10.1016/j.eswa.2014.02.026.

[2] B. Anuradha, V. C. Veera Reddy, "ANN for classification of cardiac arrhythmias," *Asian Research Publishing Network Journal of Engineering and Applied Sciences*, vol.3, no.3, 1-6, 2008.

[3] L. Bruzzone, S. B. Serpico, "A classification of imbalanced remote-sensing data by neural networks," *Pattern Recognition Letters*, vol.18, pp.1323-1328, 1997, doi: https://doi.org/10.1016/S0167-8655(97)00109-8.

[4] G. H. Nguyen, A. Bouzerdou, S. L. Phung, "A supervised learning approach for imbalanced data sets," *Proc. of the 19th International Conference on Pattern Recognition*, 1-4, 2008, doi: 10.1109/ICPR.2008.4761278.

[5] G. Pang, C. Shen, L. Cao, A. Van Den Hengel, , "Deep learning for anomaly detection: A review," *ACM Comput. Surv. (CSUR)*, 54, 38, 2021, doi: https://doi.org/10.1145/3439950.

[6] A. Adam, M. Shapiai, Z. Ibrahim, M. Khalid, "A Modified Artificial Neural Network Learning Algorithm for Imbalanced Data Set Problem," *International Conference on Computational Intelligence, Communication Systems and Networks*, CICSyN 2010, doi: 10.1109/CICSyN.2010.9.

[7] S. Wang, W. Liu, J. Wu, "Training Deep Neural Networks on Imbalanced Data Sets," *International Joint Conference on Neural Networks (IJCNN)*, 4368-4374, 2016, doi: 10.1109/IJCNN.2016.7727770.

[8] A. Li, S. Liang, A. Wang, J. Qin, "Estimating Crop Yield from Multi-temporal Satellite Data Using Multivariate Regression and Neural Network Techniques," *American Society for Photogrammetry and Remote Sensing*, Vol. 73, No. 10, 1149–1157, 2007, doi: 10.14358/PERS.73.10.1149.

[9] Y. Tang, V. N. Chawla, "SVMs Modeling for Highly Imbalanced Classification," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 39, 281 – 288, 2008, doi: 10.1109/TSMCB.2008.2002909.

[10] Y. Yang, K. Zha, "Delving into Deep Imbalanced Regression," *ICML 2021*, https://arxiv.org/abs/2102.09554.

[11] C. Huang, Y. Li, C. L. Change, X. Tang, "Learning deep representation for imbalanced classification," *IEEE conference on computer vision and pattern recognition*, pages 5375–5384, 2016, doi: 10.1109/CVPR.2016.580.

[12] M. Steininger, K. Kobs, P. Davidson, "Density-based weighting for imbalanced regression," *Mach Learn*, 110, 2187–2211, 2021, doi: https://doi.org/10.1007/s10994-021-06023-5.

[13] A. Rahim, N.A. Rashid, A. Nayan, A. Ahmad, "SMOTE Approach to Imbalanced Dataset in Logistic Regression Analysis," *ICMS 2017*, 429-433, 2019, doi: https://doi.org/10.1007/978-981-13-7279-7_53.

[14] P. Branco, L. Torgo, P. R. Ribeiro, "SMOGN: a Pre-processing Approach for Imbalanced Regression," *In First international workshop on learning with imbalanced domains: Theory and applications*, pages 36–50. PMLR, 2017.

[15] C. Peng, Q. Cheng, "Discriminative Ridge Machine: A Classifier for High-Dimensional Data or Imbalanced Data," *IEEE Trans. on Neural Networks and Learning Systems*, 2595 – 2609, 2020, doi: 10.1109/TNNLS.2020.3006877.

[16] A. SzeTo, K. C. Wong, "A Weight-Selection Strategy on Training Deep Neural Networks for Imbalanced Classification," *International Conference Image Analysis and Recognition*, 3-10, 2017, doi: https://doi.org/10.1007/978-3-319-59876-5_1.

[17] R. Akbani, S. Kwek, N. Japkowicz, "Applying Support Vector Machines to Imbalanced Datasets," *European Conference on Machine Learning (ECML)*, 39–50, 2004, doi: https://doi.org/10.1007/978-3-540-30115-8_7

[18] Y. H. Liu, Y. T. Chen, S. S. Lu, "Face Detection Using Kernel PCA and Imbalanced SVM," *International Conference on Natural Computation*, 351–360, 2006, doi: https://doi.org/10.1007/11881070_50.

[19] J. Mathew, M. Luo, C. K. Pang, H. L. Chan, "Kernel-based smote for SVM classification of imbalanced datasets," *IECON*,.1127-1132, 2015, doi: 10.1109/IECON.2015.7392251.

[20] R. Anand, K. G. Mehrotra, C.K. Mohan, S. Ranka, "An improved algorithm for neural network classification of imbalanced training sets;" *IEEE Trans. Neural Networks* 4, 962–969, 1993, doi: 10.1109/72.286891.

[21] H. Larochelle, Y. Bengio, J. Louradour, J. Lamblin, "Exploring strategies for training deep neural networks," *Journal of machine learning research*, vol 10, 1-40, 2009, doi: 10.1145/1577069.1577070.

[22] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, R. Togneri, "Cost-Sensitive Learning of Deep Feature Representations from Imbalanced Data," *IEEE Trans. Neural Network Learn System*, pp 3573 - 3587, 2017, doi: 10.1109/TNNLS.2017.2732482.

SWARUP KUMAR MONDAL is a Software Engineer at CoreLogic India. He is an Electronics and Communication Engineering Undergrad from Heritage Institute of Technology, Kolkata, India. He has one research paper publication.

His research areas are Machine Learning, Deep Learning, Optimization and Medical Image Processing.

ANINDYA SEN is a Professor at the department of Electronics and Communication Engineering, Heritage Institute of Technology, a private autonomous engineering college in Anandapur, Kolkata, India. He received his B.E. from Jadavpur University, India in the year 1980, PhD from University of Minnesota, Twin Cities in 1996, and got his Post-Doctoral training from University of Chicago from 1996 to 2000. He currently holds one US patent and 65, research paper publications. His research interests include, Medical Image processing, Internet of things, Artificial Intelligence and VLSI design.

Conceptual Business Model Framework for AI-based Private 5G-IoT Networks

Laurence Banda[*]

Wits Business School (WBS), University of the Witwatersrand, Johannesburg, 2050, South Africa
*Corresponding author: Laurence Banda, +27 834282216, laurencebandad@gmail.com

ABSTRACT: The fusion of fifth generation (5G) networks, Internet of Things (IoT) and artificial intelligence (AI), referred to as intelligent connectivity by most industry experts, can be seen as a crucial success factor for sustainable digitalization. Until recently, research into these key triad technologies has been conducted in isolation. One of the promising applications of intelligent connectivity is wireless private networks. This article presents a conceptual business model framework that can be adopted by AI-driven private 5G-IoT networks. These emerging and disruptive private networks will certainly change the mobile business landscape. Private network operators should therefore rethink and adapt their business models in order to remain economically competitive, create innovative mobile services and support sustainable practices.

KEYWORDS: 5G, Artificial Intelligence (AI), Business Model, Conceptual Framework, Internet of Things (IoT), Private Networks

1. Introduction

Fifth generation (5G) networks are being widely deployed around the world with the hope that these cutting-edge mobile systems will contribute to the sustainable digital transformation and economic prosperity of society [1]. With the introduction of more affordable devices and the expansion of commercial 5G mobile networks, customer and business interests in the market are increasing rapidly. Furthermore, the Internet of Things (IoT) and artificial intelligence (AI), which are revolutionizing our modern world through digitalization, have been enabled in large part by 5G networks [2].

IoT technologies pervasively connect natural and artificial objects and people across all network infrastructures [3]. Furthermore, IoT is considered a multidisciplinary field that encompasses numerous areas, including natural ecosystems, infrastructure, public services, social activities, technology, and the business and economic sectors [4]. The aim is to create innovative network architectures, applications and services based on smart IoT networks and smart sensors based on the 5G-IoT technology standard. 5G-IoT is therefore expected to generate massive amounts of sensor data, which requires machine learning (ML) to analyze and make intelligent decisions.

AI is a relatively new technical science that studies theories, techniques, tools and application systems that mimic and enhance human intelligence [5]. In particular, through massive data learning, machines mimic human thinking and behavior. AI is a remarkable technology with applications that cannot be fully covered by traditional methods as current solutions require advanced automation and optimization. Mobile networks in particular generate large amounts of data and make them amenable to machine learning (ML) optimization techniques. Therefore, the application of AI to networks actually refers to the use of AI/ML methods to develop supporting functions for network optimization, configuration and management operations.

The foundation of "intelligent connectivity," as some have termed it, is the combination of 5G, IoT and AI [2,6]. This is the beginning of an era characterized by ubiquitous hyper-connectivity and highly contextualized and personalized experiences. Almost every aspect of our everyday lives could be affected, including the way we work and interact with colleagues, as well as the way we
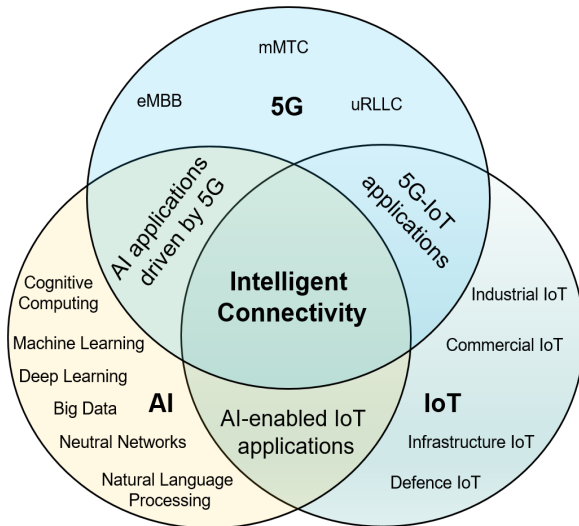
Figure 1: Fusion of 5G, IoT and AI for intelligent connectivity

consume digital entertainment [7]. Consequently, the convergence of 5G, IoT and AI will improve the quality of goods and services, lead to significant productivity gains and enable more effective use of global resources by both businesses and consumers. Key applications of intelligent connectivity include industrial IoT (iIoT), sustainable smart cities, Industry 4.0, and network security [2,6,7]. Figure 1 shows the fusion of 5G, IoT and AI for intelligent connectivity.

The 5th Generation Partnership Project (5GPP) has standardized private networks to provide specialized wireless connectivity with improved communication qualities, optimized services and tailored security features in specific locations [8]. These wireless local area networks are deployed in isolated and remote areas that are not adequately covered by traditional public cellular networks. 5G, IoT and AI are arguably the three most revolutionary technologies of the current decade. The three technologies were used independently until recently and each has exceeded our expectations. It should come as no surprise that adoption or integration of 5G, IoT, and AI is expected to increase as more businesses embrace the data trend. In addition, the combination of 5G, IoT and AI technologies has increased the number of network partners. To remain sustainable and profitable, AI-based private 5G-IoT network operators must therefore realign their business models.

This study contributes to the development of innovative business model solutions for novel triad technologies (5G, IoT and AI). In particular, the paper aims to contribute to research, development and innovation in AI-based private 5G-IoT networks by proposing a conceptual business model framework that can be adopted by operators of such networks. The article also presents some considerations for the cross-sector implementation of business models. Furthermore, the article is a continuation of our research publications on

business models for 5G mobile network operators (MNOs) and private 5G-IoT networks [9,10]. The proposed business models are viewed from both a strategic and technology theoretical perspective. The rest of the article is structured as follows. Section II highlights related works from the literature. Section III discusses private 5G-IoT networks with AI capabilities. Section IV presents the proposed conceptual business model framework, while Section V discusses industrial applications of the proposed conceptual framework. Section VI concludes the article.

2. Related Work

Recent developments in 5G wireless networks, IoT and AI have led to attempts to review relevant literature. Specifically, [11] essentially conducted a techno-economic analysis of private 5G network architectures. To accelerate the achievement of return on investment (RoI) by private network operators and other third parties, the paper focuses primarily on cost elements associated with the network deployment of private 5G networks. However, the paper ignores other crucial elements of the AI-based private 5G IoT network value chain, such as the value proposition, customer interface and infrastructure management. In [10] new business models for private 5G-IoT networks were presented. The study first develops a conceptual framework for the private 5G-IoT network value chain and then describes several business models relevant to different use cases and vertical markets. Nevertheless, the authors' study did not address the AI component.

A theoretical framework for Internet of Everything (IoE) applications for creativity in the context of 5G and 6G ecosystems was introduced in [12]. The study discusses issues with the IoE-based business platform. This is particularly about creative, cross-domain business solutions that cover different application paradigms and business platforms. However, the authors of [12] did not identify the industry's vertical target customers. Business model options for 5G and future mobile network operators were discussed in [10,13]. The paper first outlines the limitations of current business models for mobile network operators, before presenting new business models for 5G and future networks that mobile network operators should consider when introducing such networks. There was more focus on traditional or public 5G networks, while less attention was paid to private 5G networks. A business model agenda for 5G and 6G players is discussed in [14]. The authors' work focuses on the telecommunications ecosystem and emphasizes the significant changes in business models brought about by 5G. In addition, potential future business models beyond 5G and 6G are identified and attention is drawn to the respective business opportunities of the players. However, important new technologies such as IoT and AI enabled by 5G and 6G networks were not taken into account.

This article differs from previous related works in that it exclusively addresses business models for AI-powered private 5G- IoT networks using a conceptual framework; it highlights some of the key concerns of private network operators related to certain AI-driven 5G-IoT business models; and it highlights some of the economic benefits and concerns associated with integrating 5G, IoT and AI technologies.

3. AI-based Private 5G-IoT Networks

The introduction of 5G operational networks, which are almost mature, is followed by the concept and architectural design of AI-powered private 5G-IoT networks. The GSMA estimates that around 65% of the world's population will have access to 5G and high-speed mobile internet connectivity by the end of 2025 [9]. Proper deployment of new technologies such as sensor-based and AI-powered IoT, integrated with the 5G Machine Type Communication (MTC) ecosystem and associated smart industries, will enable the growth of new smart industries and drive digitalization.

3.1. 5G-IoT-AI Integration

By integrating AI into 5G IoT networks, millions of simultaneous intelligent connections are expected across a range of smart devices, automated machines, connected homes, smart grids, and smart transportation systems [9]. The idea behind intelligent connectivity is to use 5G, IoT and AI together to accelerate technological advancement and open up new, disruptive digital services [7]. In the context of intelligent connectivity, artificial intelligence (AI) technologies will analyze, contextualize and present the digital data collected by the machines, devices and sensors of the 5G-IoT networks to users in a more meaningful and useful way [6]. This would facilitate the delivery of personalized experiences to users and improve decision-making, leading to richer and more satisfying interactions between individuals and their environment.

Intelligent connectivity aims to achieve several key goals, including enhanced communication reliability, higher data rates, multiple sensor connectivity, and incredibly low power consumption [15]. However, the integration of 5G, IoT and AI faces implementation challenges due to different platforms, including: device platform (e.g. radio frequency security and device battery life), application platform (e.g. centralization constraints), and the central network platform (e.g. enabling always-online services) [15]. By dividing the physical network infrastructure into multiple logical networks (slices), each with unique service requirements and performance standards, network slicing technologies can help resolve these issues [9,16].

3.2. The Concept of AI-driven 5G-IoT Networks

A private 5G network, also known as a non-public 5G network, is a local area network that provides dedicated wireless connectivity within a specific area. Its owner can manage the network independently and has complete control over all aspects of the network, including resource scheduling, security, and priority access [17]. Therefore, an AI-based private 5G-IoT network can be described as a non-public 5G network enabled by AI technologies and tailored to IoT-based vertical markets.

Private 5G networks can be deployed in either standalone (independent) or public network integrated (dependent) mode, depending on the availability of spectrum resources and infrastructure as well as the level of network management and access control. A standalone private 5G network is a network that is deployed as an isolated and independent system without relying on a public 5G network. Conversely, a private 5G network integrated into a public network is anchored in the public network and therefore has a lower level of customization, self-control and security compared to standalone mode [9,17]. Figure 2 shows typical private 5G networks supporting various vertical markets as illustrated in [18].
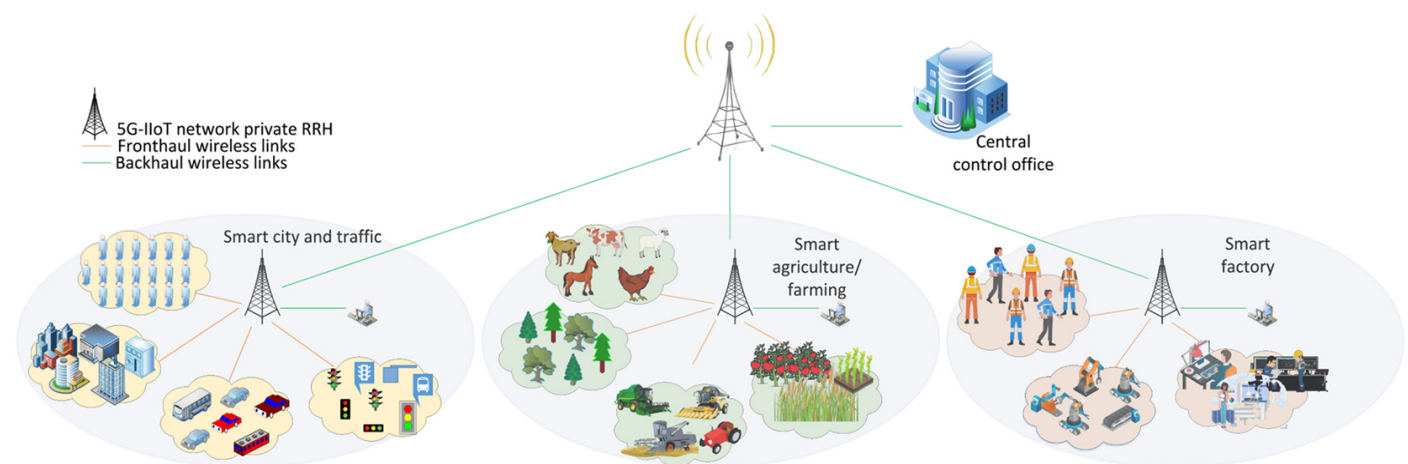

Figure 2: Private 5G networks [18]

4. Conceptual Business Model Framework

4.1. The Business Model Theoretical Concept

Since the mid-1990s, when the Internet first became widespread, the concept of business model (BM) has become increasingly popular and is now a key area of interest for researchers [19]. The term "business model" is widely used in academic literature and management practice, but there is no formal, accepted definition [19]. This is primarily because business models can be used in different research and practical fields [9]. A business model in the context of this study can be defined as follows:

The rationale for implementing the business plan of an AI-based private 5G-IoT network by leveraging interconnected elements such as customer relationships, value proposition, financial aspects and infrastructure management to create, deliver and capture value within the network ecosystem.

The present study adopts two distinct research perspectives to examine the business model concept. The first is a strategic perspective that focuses on the value proposition, creation, delivery, and capture activities within the AI-driven 5G-IoT value chain. The second perspective is technological in nature, wherein AI-driven 5G-IoT networks are considered as technological enablers for innovative business models aimed at private network operators [9]. Figure 3 illustrates the business model for AI-driven private 5G-IoT networks from both a strategic and technological perspective.

4.2. Proposed AI-based Private 5G-IoT Value Chain

Based on the business model ontology presented in [20], the relationships between the value chain components within the global market should be precisely defined in order to develop viable business models for AI-based private 5G-IoT networks. The following value chain components are proposed and shown in Figure 4.
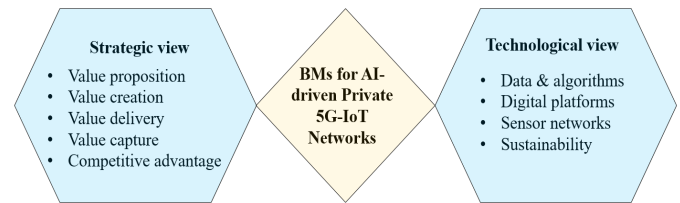
Figure 3: Strategic and technological views of AI-driven private 5G-IoT networks' business models

Who? Which target customer segments will benefit from the proposed value offered by the private network operator? This part of the value chain is about identifying the specific needs of these customers and offering tailored services to ensure their satisfaction. Customer segments include government and public institutions (such as public safety and smart cities), private companies (such as media and entertainment), industries (such as energy and healthcare), and individual consumers.

What? This value chain component explains the private network operator's value proposition for a specific customer group with the aim of generating added value for both the provider and the customer. These services include: enhanced mobile broadband (eMBB) applications such as ultra-high definition video; massive machine type communication (mMTC) applications such as smart factories; and ultra-reliable low latency communication (uRLLC) applications such as remote surgery.

How? This part of the value chain describes the technology design, network infrastructure and network resources available to the private network operator to provide the relevant services to intended customers. The following could be included in the "How?" component: AI-based key technologies for private 5G-IoT networks; sharing site infrastructure such as antennas, baseband resources, radio resources and tower space; spectrum sensing and acquisition such as mmWave band, sub-3GHz band and C-band; and collaboration with partner networks such as platform suppliers, device manufacturers and device providers.

Figure 4: Relationships between value chain components of AI-based private 5G-IoT network

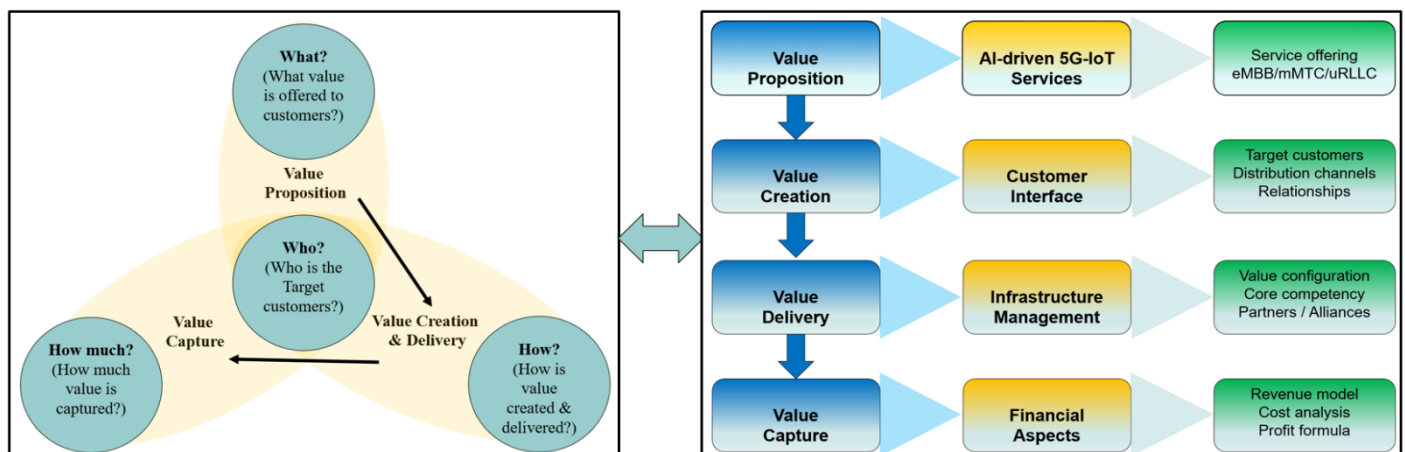**How much?** This part of the value chain provides a blueprint for how an operator wants to maximize profits while minimizing costs within the AI-based 5G private IoT network ecosystem: How much value will the private network operator derive from the target customer segment? The "How much?" components include: revenue structures such as revenue streams, profit margins and return on investment (RoI); pricing strategies such as unit cost pricing and on-demand pricing; and cost structures such as deployment costs, capital investments and operational costs.

4.3. Proposed Conceptual Business Model Framework

A conceptual framework serves as an analytical tool that provides a preliminary theory of a phenomenon under study [21]. It establishes the connection between concepts by illustrating the theories that impact the research investigation, thereby providing a theoretical basis for designing and understanding the results [21]. Table 1 shows the ten components of the proposed conceptual business model framework for AI-based private 5G-IoT networks based on the business model canvas developed by [20].

Key partners: demonstrates the need for collaborative arrangements with external organizations to effectively deliver and monetize the proposed value.

Key activities: describes the activities and competencies required to implement the private network operator's business model.

Value proposition: provides a comprehensive overview of a range of products and services offered by a private network operator.

Customer segment: describes the customer segments to which a private network operator wants to offer value. These are the target groups of the value proposition.

Customer relationships: explains the type of connections that a private network operator establishes between itself and its various customer segments.

Key resources: these are the resources available to a private network operator to ensure that value is created and delivered in line with the operator's business objectives.

Distribution channels: describes various means that a private network operator can use to engage with its customers and offer them the value proposition.

Infrastructure management: describes the technology design, network infrastructure, and network resources available to the private network operator to deliver the relevant proposed value to intended customers.

Cost structure: sums up the monetary consequences of the resources used in the business model of a private network operator.

Revenue streams: describes the way a private network operator makes money through a variety of revenue flows.

5. Discussion of Industrial Applications of the Proposed Conceptual Framework

5.1. Sustainable Industry using AI-based Private 5G-IoT Networks, Business Models and Use Cases

Different business models can be used when implementing 5G cellular technologies for AI-driven private 5G-IoT network applications. The choice of business model is influenced by a number of factors, including the industrial sector, a specific use case, capital and operating costs, and the scalability of the network [9]. In addition to providing reliable services to key stakeholders, private 5G networks are fast, easy to manage, secure and can operate in remote or isolated locations. Table 2 summarizes typical business models, vertical industry target markets, key drivers and use cases for AI-based private 5G-IoT networks.

Furthermore, one of the key killer use cases identified in the recent COP28 climate agenda is the use of emerging digital technologies such as private 5G-IoT and AI systems to monitor natural ecosystems [9,10].

5.2. Business Model Implementation Considerations

(i) Adaptive service-based approach

Some of the more exciting 5G technologies such as network slicing [16] and digital platforms [12] require a variety of key players and partners to deliver a range of intelligent services to customers. Therefore, a shift from a product-based to a flexible service-based business model is required. This is crucial because customer needs can change over time. Adaptive, service-based business models have the potential to improve user satisfaction and increase revenue for partners and operators alike.

(ii) Security-focused business model approach

Research on 5G business models has focused more on the economic value aspects of key network performance metrics such as data throughput, latency, reliability and device connectivity density, with relatively little attention paid to security objectives such as confidentiality and privacy data integrity. However, the involvement of many role actors and the complexity of the 5G ecosystem, each with different security levels, increases the vulnerability to cyberattacks [4]. Consequently, new business models should be security-focused by integrating AI and IoT-based forensics as well as various other means of combating cyber threats to ensure confidentiality, privacy and data integrity. AI and IoT forensics are also suitable tools to avert impending quantum security risks on 5G and future networks [22].

Table 1: Proposed Conceptual Business Model Framework for AI-based Private 5G-IoT Networks

Key Partners	Key Activities	Value Propositions	Customer Segment	Customer Relationships
- AI algorithm developers - IoT sensor manufacturers - Public MNOs - Platform owners & designers - Regulations & policy makers	- Managing partnerships - Network & platform maintenance - Mobile service delivery - Customer technical support - Research & development (R&D)	- Immersive mobile services - Automated manufacturing - High-speed broadband access - Smart city services - Automated factory production lines	- Mass market - Niche market - Multi-sided platforms - Individual consumers	- Personal assistance - Self-service - Automated service - Co-creation partner services
Key Resources	**Distribution Channels**	**Infrastructure Management**	**Cost Structure**	**Revenue Streams**
- Physical - Human - Intellectual - Financial	- Web-based sales - Operator's own store - Partner's store - Wholesale	- Spectrum acquisition - Backhaul infrastructure - Radio equipment - Tower space management	- Fixed costs - Variable costs - Capital expenditure - Operational expenditure	- Subscription fees - Usage fees - Leasing/renting assets - Brokerage fees - Advertising

Table 2: AI-based Private 5G-IoT Network Business Models, Vertical Industries, Key Drivers, and Use Cases

AI- based Private 5G-IoT Network Business Model	Vertical Industry	Key Drivers	Use Cases
Smart Factory	• Manufacturing	• Proliferation of mobile robotics in process automation.	• Connected workers • Production line flexibility • End-to-end logistics
Smart Mining	• Mining	• Demand for high safety standards in the mining industry. • Rapid transition from human-driven machine control to automated remote controlled applications	• Early warning disaster signaling • Safety signal dissemination • Automated drilling machine communication
e-Health (Healthcare 5.0)	• Health	• Advancement in medical research • Low density of health practitioners • Upsurge in remote surgery practice	• Wearable health monitoring devices. • Smart implants for medical diagnostics • Connected contact lenses
Smart Sea Port	• Maritime	• Surge in sea port activities mainly driven by cargo transportation which cannot be handled by the aviation industry.	• Remote-controlled & automated cranes • Automated guided vehicles (AGVs) • Unmanned aerial vehicles (UAVs) • Sea condition monitoring
Smart Airport (Airport 4.0)	• Aviation	• Rise in air freight activities. • Upsurge of passenger air traffic, post COVID-19 era.	• Automated AI-based boarding procedures for passengers. • Constant security monitoring • Automated fever detection • Optimized luggage handing
Smart Railway	• Transportation	• Increase in use of railway systems as a means of inland cargo transportation. • Increased number of train commuters due to road congestion.	• Smooth scheduling & operations of trains. • Secure and low latency communication between drivers and signaling controllers. • Automated AI-based boarding procedures for passengers.
Smart City	• Public	• Rapid urban/city population explosion • Increase in CO_2 emission levels in urban areas	• Remote monitoring of roads and city infrastructure. • Automated water sanitation and waste management systems • Air pollution level detection
Smart Grid	• Energy	• Multiple wireless sensors with uninterrupted 5G connectivity deployed in specific substations.	• Effective monitoring and forecast of energy demand and consumption • Automated adjustment of electric profile via load control.
Smart Farming	• Agriculture	• Multiple wireless sensors with uninterrupted 5G connectivity deployed in specific crop fields.	• Optimized soil moisture content • Efficient use of water and fertilizers via customized applications
e-Learning	• Education	• Immersive real-time interaction via virtual presence applications with minimal visual or audio delays	• Immersive real-time virtual interaction between learners and educators. • Tele-education use cases for distance-learning courses.

(iii) Support for green and sustainable energy-efficient communication

Supporting green and energy-efficient communication technologies through business models for AI-based private 5G-IoT networks is critical to the environmental sustainability of these networks. To reduce the reliance on conventional fossil fuels to power network devices, green energy sources must be utilized. One of the goals of 5G-based network design is to minimize carbon emissions and reduce energy consumption of network devices without compromising network quality. In [23], research was conducted on energy harvesting techniques that enable communication devices to harvest energy from a variety of renewable sources. However, so far neither industry nor academia have shown much interest in exploring business models aimed at enabling environmental friendliness in the 5G ecosystem. The lack of end-2-end (E2E) testbeds or digital twins for AI-driven private 5G networks is detrimental to the development of innovative and sustainable 5G-IoT services and business models.

6. Conclusion

This article presented a conceptual business model framework for AI-based private 5G-IoT networks. The conceptual framework, formulated from a strategic and technological perspective, consists of ten inter-connected components that can be adopted by private network operators. The technical aspects and associated applications of AI-based private 5G IoT networks were highlighted. In addition, a discussion of the industrial applications of the proposed conceptual framework was presented. In particular, typical business models, vertical industry target markets, key drivers and exemplary use cases were discussed. Finally, critical aspects of a sustainable business model that must be taken into account when implementing AI-based private 5G-IoT networks were highlighted.

Since the current work is based on a theoretically formulated conceptual framework, future research will be implemented on an E2E testbed with machine learning-based analysis applied to the collected primary data to support the development of sustainable business models and innovative services for current and future 6G AI/IoT-based smart and sustainable industries and the preservation of natural ecosystems.

References

[1] J. Whalley, P. Curwen, "Creating value from 5G: The challenge for mobile operators," *Telecommunications Policy*, vol. 48, no. 2, 2024, doi:10.1016/j.telpol.2023.102647.

[2] E. Esenogho, K. Djouani, A.M. Kurien, Integrating Artificial Intelligence Internet of Things and 5G for Next-Generation Smartgrid: A Survey of Trends Challenges and Prospect, *IEEE Access*, vol. 10, , 2022, doi:10.1109/ACCESS.2022.3140595.

[3] A. Ghasempour, Internet of things in smart grid: Architecture, applications, services, key technologies, and challenges, *Inventions*, vol. 4, no. 1, 2019, doi:10.3390/inventions4010022.

[4] D.C. Nguyen, M. Ding, P.N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. Dobre, H.V. Poor, "6G Internet of Things: A Comprehensive Survey," *IEEE Internet of Things Journal*, vol. 9, no. 1, 2022, doi:10.1109/JIOT.2021.3103320.

[5] J. Tan, X. Sha, T. Lu, B. Dai, "A Short Survey on Future Research of AI and IoT Technologies," in *2022 International Wireless Communications and Mobile Computing, IWCMC 2022*, 2022, doi:10.1109/IWCMC55113.2022.9825425.

[6] A. Yarali, Intelligent Connectivity: AI, IoT, and 5G, 2021, doi:10.1002/9781119685265.

[7] M. Attaran, "The impact of 5G on the evolution of intelligent automation and industry digitization," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 5, 2023, doi:10.1007/s12652-020-02521-x.

[8] P. Curwen, J. Whalley, "Private Networks: A regular column on the information industries," *Digital Policy, Regulation and Governance* , vol. 23, no. 4, 2021, doi:10.1108/DPRG-06-2021-181.

[9] L. Banda, M. Mzyece, F. Mekuria, "5G Business Models for Mobile Network Operators - A Survey," *IEEE Access*, vol. 10, , 2022, doi:10.1109/ACCESS.2022.3205011.

[10] L. Banda, M. Mzyece, F. Mekuria, "Business Models for 5G-IoT Private Networks," in *IEEE AFRICON Conference*, 2023, doi:10.1109/AFRICON55910.2023.10293454.

[11] H. Frank, C. Colman-Meixner, K.D.R. Assis, S. Yan, D. Simeonidou, "Techno-Economic Analysis of 5G Non-Public Network Architectures," *IEEE Access*, vol. 10, , 2022, doi:10.1109/ACCESS.2022.3187727.

[12] S. Mishra, A.R. Tripathi, "IoT Platform Business Model for Innovative Management Systems," *International Journal of Financial Engineering*, vol. 07, no. 03, 2020, doi:10.1142/s2424786320500309.

[13] L. Banda, M. Mzyece, F. Mekuria, "Business Models for 5G and Future Mobile Network Operators," in *Proceedings - 2022 IEEE Future Networks World Forum, FNWF 2022*, 2022, doi:10.1109/FNWF55208.2022.00045.

[14] M. Moussaoui, E. Bertin, N. Crespi, "Divide and Conquer: A Business Model Agenda for Beyond-5G and 6G," *IEEE Communications Magazine*, vol. 61, no. 7, 2023, doi:10.1109/MCOM.001.2200748.

[15] K. Shafique, B.A. Khawaja, F. Sabir, S. Qazi, M. Mustaqim, Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT Scenarios, *IEEE Access*, vol. 8, , 2020, doi:10.1109/ACCESS.2020.2970118.

[16] X. Foukas, G. Patounas, A. Elmokashfi, M.K. Marina, Network Slicing in 5G: Survey and Challenges, *IEEE Communications Magazine*, vol. 55, no. 5, 2017, doi:10.1109/MCOM.2017.1600951.

[17] M. Wen, Q. Li, K.J. Kim, D. Lopez-Perez, O. Dobre, H.V. Poor, P. Popovski, T. Tsiftsis, "Private 5G Networks: Concepts, Architectures, and Research Landscape," *IEEE Journal on Selected Topics in Signal Processing*, vol. 16, no. 1, 2022, doi:10.1109/JSTSP.2021.3137669.

[18] B.S. Khan, S. Jangsher, A. Ahmed, A. Al-Dweik, "URLLC and eMBB in 5G Industrial IoT: A Survey," *IEEE Open Journal of the Communications Society*, vol. 3, , 2022, doi:10.1109/OJCOMS.2022.3189013.

[19] C. Zott, R. Amit, L. Massa, The business model: Recent developments and future research, *Journal of Management*, vol. 37, no. 4, 2011, doi:10.1177/0149206311406265.

[20] A. Osterwalder, Y. Pigneur, "Investigating the use of the business model concept through interviews," *Shaping Business Strategy in a Networked World, Vols 1 and 2, Proceedings*, 2004.

[21] S. Leshem, V. Trafford, Overlooking the conceptual framework, *Innovations in Education and Teaching International*, vol. 44, no. 1, 2007, doi:10.1080/14703290601081407.

[22] H. Han, J. Yao, Y. Wu, Y. Dou, J. Fu, "Quantum communication based cyber security analysis using artificial intelligence with IoMT," *Optical and Quantum Electronics*, vol. 56, no. 4, 2024, doi:10.1007/s11082-023-06185-7.

[23] M. Eshaghi, R. Rashidzadeh, "An Energy Harvesting Solution for IoT Devices in 5G Networks," in *Canadian Conference on Electrical and Computer Engineering*, 2020, doi:10.1109/CCECE47787.2020.9255802.

LAURENCE BANDA received the BEng degree in Electrical and Electronics Engineering from the University of Zambia, Lusaka, Zambia in 2006, the MSc in Electronics and Communications Engineering from EISEE, Paris, France in 2011, the MEng in Electrical Engineering from the Tshwane University of Technology, Pretoria, South Africa in 2012 and the PhD in Technology and Operations Management from the University of the Witwatersrand, Johannesburg, South Africa in 2024.

He worked for Huawei Technologies in South Africa for 10 years where he held various positions including Project Manager for Wireless Energy Management, Wireless Advisory Engineer and Technical Trainer. He has published widely in the areas of business models for wireless networks. His current research focus is on technology and innovation management for next-generation wireless networks incorporating AI and IoT.

Navigating the Autonomous Era: A Detailed Survey of Driverless Cars

Vaibhavi Tiwari[*]

Vaibhavi Tiwari, School of Computing, Montclair State University, New Jersey, USA

[*]Corresponding author: Vaibhavi Tiwari, Montclair State University, Email: tiwariv1@montclair.edu

ABSTRACT: The incorporation of cutting-edge technologies like sensor networks, artificial intelligence (AI), and vehicle-to-everything (V2X) communication has hastened the rollout of autonomous vehicles (AVs), offering significant possibilities for the future of transportation. This document offers an extensive overview of AV technology, covering essential elements such as technological infrastructure, degrees of automation, cybersecurity threats, societal impacts, regulatory structures, and emerging trends. This analysis emphasizes the existing obstacles and progress within the industry by examining the activities of key entities like Tesla, Waymo, and General Motors. Additionally, a comparative examination of autonomous vehicles and drones is performed, providing distinct perspectives on possible cybersecurity vulnerabilities shared by both technologies, including GPS spoofing, jamming, and unauthorized data interception. This multifaceted approach highlights not only the existing vulnerabilities but also proposes proactive measures that can be implemented to reduce comparable risks across various AV platforms. The results highlight the necessity of establishing strong cybersecurity measures, overcoming regulatory challenges, and building public confidence to realize the complete promise of autonomous vehicles as secure, effective, and eco-friendly transportation options. This analysis provides an essential resource for comprehending the complex aspects of AV technology and its consequences, offering readers a comprehensive perspective on the challenges and opportunities within the autonomous vehicle sector.

KEYWORDS: Driverless cars, autonomous vehicles, sensors, AI, V2X communication, SAE levels of automation, cybersecurity

1. Introduction

The automotive sector is swiftly progressing towards a future characterized by autonomous vehicles (AVs), with driverless cars at the forefront of this evolution. The vehicles utilize state-of-the-art technologies, such as intricate sensor networks, advanced artificial intelligence (AI), and real-time data processing, which together hold the potential to transform transportation by improving safety, efficiency, and convenience. Nonetheless, the incorporation of these complex systems brings forth significant cybersecurity issues that need to be tackled to safeguard AV performance and guarantee user safety.

Historically, autonomous vehicle research has spanned several decades, marking the transition of AVs from speculative fiction to a reality on our roads today. Experiments in this field began as early as the 1920s with Ralph Teetor's invention of cruise control, and later expanded into semi-autonomous systems developed by Japan's Tsukuba Mechanical Engineering Laboratory in the 1970s. These early projects laid the foundation for subsequent advancements, including Carnegie Mellon University's Navlab and ALV projects, which achieved milestones in autonomous cross-country travel by the mid-1990s [1, 2, 3]. Government support, such as the United States' $650 million allocation for the National Automated Highway System in the 1990s [4], further bolstered AV innovation, eventually enabling private industry players like Waymo, Tesla, and Nuro to lead commercial deployments from the late 2010s onward [5, 6].

In recent years, the concept of AVs has rapidly advanced due to technological progress, transitioning from controlled test environments to limited public road usage. Notably, McKinsey's 2023 global executive survey highlights significant advancements, with insights from 86 decision-makers forecasting the commercial availability of Level 4 (L4) autonomous vehicles and robo-taxis by 2030. Despite these advancements, substantial financial investments remain necessary. For instance, developing fully autonomous trucks and L4/L5 robo-taxis requires over $4 billion and $5 billion, respectively [7]. This ongoing progress underscores the industry's commitment but also highlights the resource-intensive nature of AV development.

This study uniquely contributes to the AV landscape by conducting a comparative analysis between AVs and drones. This comparison is particularly valuable because drones have encountered a wide array of cybersecurity threats. By examining these threats, the research explores how similar vulnerabilities could manifest within AV systems, thereby offering a basis for understanding potential misuse of AV technology. This insight allows for proactive mitigation strategies that can be applied to both AVs and drones, enhancing our overall preparedness against cyber threats.

However, the adoption of AVs at scale faces numerous hurdles, including regulatory, safety, and cybersecurity concerns. The cybersecurity landscape is especially complex, with threats such as remote exploits, unauthorized access, and adversarial machine learning attacks posing significant risks to AV systems [8]. The increasing reliance on wireless communication in AVs exposes them to attacks that could compromise vehicle control, sensitive data, and communication networks. The rapid evolution of cybersecurity risks between 2023 and 2024, such as adversarial attacks against AI systems and supply chain vulnerabilities, further emphasizes the need for robust cybersecurity protocols.

Beyond cybersecurity, AVs must also navigate regulatory and societal challenges. As the industry grows—projected to reach $556.67 billion by 2026—collaboration among automotive sectors, regulatory bodies, and researchers becomes essential to address the socioeconomic impacts, such as job displacement within driving-related industries. Trust-building initiatives, such as Waymo's extensive road testing and Tesla's Full Self-Driving (FSD) program, continue to play a critical role in driving public acceptance [6, 9, 10].

Through this study, readers will gain insights into AV technologies, the associated cybersecurity risks, and the broader societal and regulatory challenges. By presenting a detailed comparative analysis with drones, this research not only identifies shared vulnerabilities but also provides a framework for preemptive mitigation strategies. This exploration is intended to equip readers with a holistic understanding of AV technology, guiding them through the steps needed for safe, efficient, and socially responsible autonomous transportation. The projected growth trends [11] for the autonomous vehicle market are illustrated in Figure 1.

Figure 1: Statistics of the Autonomous Vehicle Market (2023 - 2033)

2. Motivation

There are a number of powerful incentives that are driving the development of autonomous vehicles. Each of these incentives addresses significant societal, economic,

and environmental challenges respectively:

2.1. Enhancing Road Safety

An important driving force behind the development of autonomous vehicles is the ability to improve road safety by greatly reducing human error, which is a major contributor to traffic accidents. As per the World Health Organization, some 1.3 million individuals perish annually in road traffic accidents, with the bulk of these incidents being caused by human error [12, 13]. Driverless cars utilize sophisticated sensors, machine learning algorithms, and real-time data processing to enhance decision-making capabilities, surpassing those of human drivers. This results in a decreased probability of accidents and a significant preservation of human life [14].

2.2. Increasing Transportation Efficiency

Through the optimization of vehicle movement and the reduction of the amount of time spent driving, autonomous vehicles have the potential to change the efficiency of transportation. It is possible for autonomous vehicles to establish contact with one another as well as with traffic management systems. This allows them to coordinate their movements, limit the number of occasions in which they are forced to stop and go, and determine the routes that are ultimately the most efficient. It is possible that these outcomes will result in greater traffic efficiency, decreased traffic congestion, and shorter travel durations [15, 16]. These outcomes are beneficial to individual commuters as well as the economy as a whole because they will increase productivity and decrease fuel consumption simultaneously [17, 18].

2.3. Reducing Traffic Congestion

Traffic congestion has a substantial impact on urban areas, leading to inefficiency in terms of time management, increased levels of pollution, and economic setbacks. Through the synchronization of their movements, the optimization of the timing of traffic lights, and the reduction of the need for parking spots in densely populated metropolitan areas, autonomous cars have the potential to alleviate traffic congestion. The integration of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication gives autonomous cars the capacity to dynamically adjust their speed and routes in order to avoid traffic congestion and maintain a continuous movement of vehicles. Such a feature allows autonomous vehicles to maintain a consistent movement of vehicles [19, 20, 21].

2.4. Improving Mobility for All

Autonomous vehicles have the potential to significantly improve the transportation alternatives available to individuals who are unable to operate a vehicle, such as the elderly, the disabled, or those who do not have their own means of transportation [22]. Individuals who may have difficulties with mobility can benefit from the on-demand transportation services provided by autonomous vehicles, which offer a reliable and convenient mode of transportation. This has the potential to improve self-sufficiency, facilitate access

to essential services, and eventually lead to an improvement in the overall quality of life for the aforementioned demographics [23].

2.5. Contributing to Environmental Sustainability

The optimization of driving patterns, the decrease of idle time, and the encouragement of the use of electric vehicles (EVs) are all ways in which autonomous vehicles have the potential to improve environmental sustainability. Because they are able to operate more efficiently and eliminate excessive acceleration and braking, driverless cars have the potential to reduce the amount of fuel that is consumed as well as the emissions of greenhouse gases [24]. A large number of projects involving autonomous vehicles are currently being developed concurrently with the development of electric vehicle technology. This has the potential to significantly reduce the negative impact that transportation has on the environment [25]. The incorporation of autonomous technology into electric vehicles (EVs) has the potential to hasten the adoption of clean energy within the transportation sector, thereby making a contribution to broader environmental goals [24, 26].

2.6. Economic Benefits

There is a high probability that the introduction of autonomous vehicles will result in considerable economic benefits. It is possible to realize significant cost savings in the areas of healthcare, emergency services, and vehicle repairs by reducing the number of accidents that occur on the roads. A large reduction in the number of accidents that occur in the United Kingdom, for instance, might result in savings of almost two billion pounds by the year 2030 at the very least [27]. Furthermore, it is predicted that the autonomous automobile industry will provide a multiplicity of employment opportunities in the areas of technological innovation, infrastructure advancements, and mobility services. This sector would also deliver more than £51 billion in economic benefits [28].

2.7. Accessibility and Inclusion

Autonomous vehicles have the potential to provide individuals with disabilities, the elderly, and other individuals who do not drive with enhanced mobility options, thereby boosting their freedom and general well-being [22]. Autonomous cars have the potential to be designed in such a way that they can accommodate a wide range of accessibility requirements, thereby offering a reliable and safe mode of transportation to all users.

3. Technologies Behind Driverless Cars

The technology used in driverless automobiles is varied and intricate, encompassing the integration of advanced sensors, artificial intelligence, vehicle-to-vehicle and vehicle-to-infrastructure communication, and sophisticated decision-making algorithms. These technologies collectively allow autonomous vehicles to accurately detect their surroundings, precisely determine their location, strategize safe routes, and efficiently navigate while interacting with other vehicles

and infrastructure. Table 1 provides a brief overview of these technologies, highlighting the essential components used in each.

Table 1: An Overview of Technologies

Technology	Description
Sensors and Perception Systems	Uses LiDAR, Radar, Cameras, and Ultrasonic sensors for environmental mapping. Provides real-time detection of surroundings.
Localization and Mapping	GPS, IMUs, and SLAM for positioning. Enables navigation in complex environments.
Decision-Making Algorithms	Path planning, obstacle avoidance, predictive modeling. Adapts to real-time traffic conditions.
Vehicle-to-Vehicle (V2V) Communication	Shares data on speed, position, and intentions. Improves coordination and safety.
Vehicle-to-Infrastructure (V2I) Communication	Interacts with infrastructure like traffic lights. Optimizes traffic flow.
Artificial Intelligence and Machine Learning	Deep learning for perception and decision-making. Reinforcement learning for pattern recognition.

3.1. Sensors and Perception Systems

The powerful sensors and vision systems used in autonomous vehicles are central to their operational capabilities. These systems combine various types of sensors to collect and interpret data from the surrounding environment, generating a comprehensive view. LiDAR (Light Detection and Ranging) systems emit laser pulses, measuring the time it takes for them to reflect off objects, which helps create high-resolution 3D maps [29]. This is crucial for identifying the shape, distance, and size of obstacles, enabling precise environmental mapping and navigation. Radar sensors, which use radio waves to detect the distance and speed of objects, are essential for adaptive cruise control and collision avoidance, and are particularly effective in adverse weather where optical sensors may fail [30]. Visual cameras capture images of the environment that are processed using computer vision algorithms for tasks such as object recognition, lane detection, and traffic sign identification. Modern autonomous vehicles typically deploy multiple cameras to cover various angles, creating a complete visual representation of the surroundings. Ultrasonic sensors [31], primarily used for short-range detection, assist in parking maneuvers by detecting objects close to the vehicle, ensuring safety during low-speed operations. Collectively, these perception systems synthesize data from multiple sources to produce an accurate model of the vehicle's surroundings, which is fundamental for decision-making and control activities.

3.2. Localization and Mapping

Accurate localization and mapping are essential for autonomous vehicle navigation, as they enable vehicles to determine their precise position and navigate complex environments. The Global Positioning System (GPS) provides basic positioning information, but in urban settings, where signal blockages by tall buildings can occur, GPS is often supplemented by other localization technologies. Inertial Measurement Units (IMUs) track the vehicle's orientation and motion, supplying critical data for dead-reckoning

and enhancing GPS accuracy. Simultaneous Localization and Mapping (SLAM) is another key technology, involving algorithms that build a map of the environment while simultaneously tracking the vehicle's position within it. SLAM integrates data from LiDAR, cameras, and IMUs to produce detailed and accurate maps, which are vital for navigating dynamic and unfamiliar surroundings. Advanced navigation systems often employ a hybrid approach, combining fundamental maps with real-time perception data to adapt to immediate environmental changes [32]. Innovations from institutions like MIT's CSAIL have even developed systems that rely on sparse topological maps and real-time sensor data, allowing autonomous vehicles to navigate without detailed maps.

3.3. Decision-Making Algorithms

Autonomous vehicles rely on sophisticated decision-making algorithms to interpret sensor data and execute driving tasks safely and effectively. Path planning algorithms determine the optimal route from the vehicle's current position to its destination, using techniques such as graph-based search (e.g., A* algorithm) and optimization methods to ensure collision-free paths. These algorithms also dynamically adjust to real-time traffic conditions and obstacles [33]. Obstacle avoidance algorithms detect and navigate around obstacles with inputs from LiDAR, radar, and cameras, utilizing methods like Detection and Tracking of Moving Objects (DATMO) to forecast obstacle movements and adjust the vehicle's path accordingly [33]. Additionally, predictive modeling anticipates the actions of other road users, such as pedestrians or other vehicles, by applying machine learning models, including deep neural networks, to enhance the accuracy of these predictions. Together, these decision-making processes enable the vehicle's control system to analyze the environment, adhere to traffic rules, and ensure secure and efficient movement.

3.4. Vehicle-to-Vehicle (V2V) Communication

Vehicle-to-Vehicle (V2V) communication enables autonomous cars to share real-time information on their speed, position, and driving intentions. This exchange of data improves situational awareness and facilitates coordinated maneuvers, such as platooning, where vehicles travel in close formations to reduce aerodynamic drag and enhance fuel efficiency. V2V communication ensures precise coordination of speed and braking among platooned vehicles, which bolsters both safety and efficiency. Furthermore, V2V is crucial for accident prevention, as it allows vehicles to relay alerts about potential hazards, sudden stops, or other critical incidents [34].

3.5. Vehicle-to-Infrastructure (V2I) Communication

Vehicle-to-Infrastructure (V2I) communication involves interactions between vehicles and road infrastructure, such as traffic lights, road signs, and other smart systems. This technology plays a key role in optimizing traffic flow, as vehicles can receive real-time updates on traffic conditions, signal timings, and available detours, thereby reducing congestion and improving travel times. V2I communication also enhances road safety by providing vehicles with warnings about hazards, construction zones, or changes in road conditions. Integration with smart city infrastructure further extends V2I's capabilities, enabling comprehensive traffic control and significant improvements in urban transportation [34].

3.6. Artificial Intelligence and Machine Learning

Artificial Intelligence (AI) and Machine Learning (ML) are critical components of the architectures that enable autonomous driving. These technologies empower vehicles to learn from data, recognize patterns, and make decisions in complex situations. Deep Learning, for instance, is used extensively for visual perception, object recognition, and decision-making. Through deep neural networks, autonomous vehicles can process inputs from various sensors to detect objects, recognize traffic signs, and predict the actions of other road users. Reinforcement Learning, on the other hand, allows vehicles to learn from real-world driving experiences and continually enhance their performance. This adaptive approach enables autonomous systems to refine their decision-making processes over time and respond effectively to new environments. Overall, AI systems in autonomous vehicles facilitate the interpretation of sensory data, enable situational awareness, and support safe navigation without human intervention, improving their capabilities as they encounter diverse driving conditions.

4. Levels of Autonomy

4.1. SAE Levels of Automation

The Society of Automotive Engineers (SAE) has defined six levels of vehicle automation, from Level 0 (no automation) to Level 5 (full automation), which describe the degree of driver intervention required and the vehicle's capabilities at each stage [35]. These levels provide a framework to understand the evolution and technical capabilities of AV systems:

1. **Level 0 (No Automation)**: At this level, the human driver is fully responsible for controlling the vehicle. While some technologies, such as warnings or momentary assistance (e.g., collision alerts), may be present, they do not control the vehicle.

2. **Level 1 (Driver Assistance)**: Basic driver-assist technologies, such as adaptive cruise control or lane-keeping assistance, enable limited control over either steering or acceleration/deceleration, but not simultaneously. Current technologies at this level often employ sensors and basic AI to interpret lane markings or maintain a safe following distance.

3. **Level 2 (Partial Automation)**: The vehicle can control both steering and speed under certain conditions, using a combination of radar, LiDAR, and camera systems to monitor the environment. Although the driver remains responsible for monitoring the road and staying alert, this level marks the transition to shared control.

4. **Level 3 (Conditional Automation)**: Vehicles at this level can manage all driving tasks under specific conditions, such as highway driving. However, the driver must be ready to take over when the system requests. The transition between automated and manual control relies on advanced sensor fusion and environmental mapping, allowing the system to make decisions based on real-time data processing.

5. **Level 4 (High Automation)**: At this stage, the vehicle can perform all driving functions and monitor the environment in designated operational design domains (ODDs), such as urban areas or specific weather conditions, without driver intervention. While human oversight is not needed within the ODD, technological limitations prevent full autonomy under all conditions, with industry leaders like Waymo and Cruise currently testing Level 4 systems in pilot.

6. **Level 5 (Full Automation)**: This level represents the goal of fully autonomous vehicles, capable of performing all driving tasks in all conditions without any human intervention. The technical challenges include creating a robust infrastructure of AI, machine learning, and V2X communication, but full industry adoption remains unrealized as of 2024.

Industry Adoption and Comparison: While significant strides have been made in Levels 2 and 3, full Level 5 automation is not yet achieved. Currently, automakers and tech firms, such as Tesla and Waymo, are focusing on improving Levels 3 and 4 with varied strategies. Tesla's approach, for example, relies heavily on computer vision and advanced neural networks for higher levels of autonomy, while Waymo incorporates high-definition mapping and extensive LiDAR systems. The comparative analysis within this paper explores these strategies, emphasizing how different companies prioritize elements like sensor fusion and machine learning to overcome the unique challenges at each automation level. By highlighting these approaches, the paper outlines key industry practices that contribute to the broader landscape of AV technology development [35].

4.2. Gradual Progression Towards Full Autonomy

The transition towards complete autonomy is slow, with cars now functioning at either Level 2 or Level 3. These levels enable vehicles to do specific driving tasks while still necessitating human supervision. The transition to higher levels of automation will be driven by notable progress in technology and legal frameworks. For instance, the testing and implementation of Level 3 systems such as Honda's "Traffic Jam Pilot" and Mercedes-Benz's Drive Pilot demonstrate encouraging advancements in minimizing human intervention in particular situations.

Autonomous vehicles must address various challenges before achieving full autonomy, including:

- **Advanced Software and Mapping**: Ensuring the software can handle diverse driving conditions and environments.

- **Human Factors**: Developing systems that can safely transfer control between human drivers and autonomous systems.

- **Regulatory and Legal Frameworks**: Establishing standards for liability, safety, and operation of autonomous vehicles.

- **Ethical and Security Concerns**: Addressing ethical dilemmas such as the "trolley problem" and ensuring robust cybersecurity measures.

The continued research and development efforts that are being made by industry leaders, government agencies, and academic institutions are absolutely necessary in order to overcome these hurdles and make progress toward completely autonomous vehicles.

5. Current State-of-the-Art

5.1. Industry Leaders and Major Players

Several important players who are at the forefront of developing and deploying autonomous vehicle (AV) technology are present in the industry of driverless cars, which is characterized by the presence of these key players. The following are notable businesses:

Tesla: Tesla has made great progress with its Full Self-Driving (FSD) software, which promises to provide full autonomy through continual over-the-air upgrades. This software has undergone significant development. Rather than relying on LiDAR, Tesla's strategy makes use of a vision-based system that is equipped with cameras. Additionally, the company extensively depends on artificial intelligence and neural networks to interpret visual data.

Waymo: Waymo, which happens to be a part of Alphabet Inc., is widely regarded as a pioneer in the field of autonomous vehicles. Waymo's autonomous driving systems are able to reach a high level of precision and reliability thanks to the utilization of a combination of LiDAR, radar, and cameras. Waymo's self-driving taxis, which are currently operating in Phoenix, Arizona, have delivered thousands of rides, in addition to providing critical data from the real world.

Uber: The Advanced Technologies Group (ATG) of Uber has been contributing to the development of autonomous vehicles (AVs) for its ride-hailing services. This group has been working on self-driving technologies. In order to enhance the safety and effectiveness of its autonomous systems, Uber has carried out extensive testing in a variety of metropolitan situations.

General Motors (GM) and Ford: GM and Ford, established car manufacturers, are making significant financial commitments to autonomous technology through their respective companies, Cruise and Argo AI. These firms are investigating several uses of autonomous vehicles (AVs), such as ride-sharing and delivery services, by utilizing their substantial knowledge and infrastructure in manufacturing.

5.2. Commercial Deployments and Pilot Programs

When it comes to the collection of data and the improvement of autonomous vehicle technology, commercial deployments and pilot programs are absolutely necessary. There are many noteworthy programs, including:

Waymo One: Waymo's autonomous taxi service has been operational in Phoenix, Arizona, since December 2018,

and recently expanded to San Francisco and Los Angeles. Within the confines of a geofenced area, this business offers trips that are completely autonomous to the general public, collecting vital data that may be used to improve their technology.

Tesla Full Self-Driving (FSD) Beta: Thousands of Tesla owners are participating in the FSD beta program, which is evaluating the software under real-world settings while it is being developed. Continuous feedback is provided to Tesla by this program, which enables incremental changes to be made to the capabilities of the FSD capability.

Cruise by GM: Cruise has been doing extensive testing of its driverless vehicles in San Francisco, with a particular emphasis on testing them in urban situations. In order to capitalize on General Motors' manufacturing skills, Cruise intends to establish a commercial robotaxi service shortly.

Argo AI and Ford: A number of cities, including Miami and Austin, are participating in pilot programs that are being carried out by Argo AI in conjunction with Ford. Within the context of ride-sharing and delivery services, these activities are intended to improve the autonomous driving systems in preparation for their eventual deployment in commercial settings.

5.3. Technological Limitations and Challenges

In spite of the progress that has been made, autonomous cars continue to confront a number of technological restrictions and opportunities:

Complex Urban Environments: This is a challenge for autonomous vehicles because of the unpredictability and complexity of urban environments. The use of autonomous vehicles (AV) presents considerable issues in situations where there is a high volume of traffic, pedestrians, cyclists, and emergency vehicles.

Adverse Weather Conditions: The reliability of autonomous vehicles (AV) systems can be negatively impacted by weather conditions such as rain, snow, and fog, which can hinder the functioning of sensors like as cameras and LiDAR. The creation of reliable systems that are capable of functioning successfully in any and all weather circumstances continues to be a significant problem.

Cybersecurity: When it comes to protecting autonomous vehicles (AVs) against hacking and data breaches, it is necessary to provide comprehensive cybersecurity safeguards. Because of the high level of connectivity that exists amongst autonomous vehicles, any vulnerabilities that may exist could be exploited, which could result in the theft of important data or the malicious control of the vehicle.

Regulatory and Ethical Issues: A thorough regulatory framework must be developed in order to facilitate the broad use of autonomous vehicles (AVs). The establishment of standards for safety, liability, and insurance is included in this requirement. In addition, it is necessary to handle ethical conundrums, such as the process of making decisions in situations where accidents are unavoidable.

To summarize, although there have been notable progressions, it is imperative to do further research and development in order to surmount these obstacles and fully realize the capabilities of autonomous cars.

6. Existing Cybersecurity Threats

Driverless cars are equipped with a multitude of interconnected systems that manage everything from navigation to communication with external devices. These systems are susceptible to various types of cyberattacks, including:

6.1. Remote Exploits and Unauthorized Access

The dependence on wireless connectivity renders autonomous vehicles susceptible to remote exploitation. Unauthorised gain of access to the vehicle's control systems by attackers has the potential to result in catastrophic failures. As an illustration, in 2015, security researchers Charlie Miller and Chris Valasek successfully conducted a remote hack on a Jeep Cherokee by taking advantage of weaknesses in its Uconnect smart entertainment system. Unauthorized access to the vehicle's internal network was successfully obtained, enabling the delivery of commands to vital systems such as the engine, gearbox, and brakes. The cyber assault demonstrated that vehicles with unsecured connectivity might be influenced from any location, therefore giving rise to significant apprehensions regarding the security of autonomous vehicles [36].

6.2. Data Breaches and Privacy Concerns

Autonomous vehicles amass huge quantities of data, encompassing personal information as well as driving trends. Data breaches have the potential to result in the unauthorised acquisition and improper use of this confidential data. The 2016 Uber data breach, which compromised the personal data of 57 million passengers and drivers, highlights the possible hazards linked to the extensive data retention inherent in autonomous vehicles. Uber's former Chief Security Officer, Joe Sullivan, endeavored to conceal the breach by offering the hackers $100,000 as part of a "bug bounty" scheme, resulting in severe legal repercussions [37].

6.3. Vehicle-to-Everything (V2X) Communication Attacks

Driverless cars communicate with infrastructure, other vehicles, and even pedestrians through Vehicle-to-Everything (V2X) technology. These communications are critical for safe operation but are vulnerable to various types of attacks.

In 2019, researchers demonstrated the feasibility of spoofing GPS signals to mislead autonomous vehicles, potentially causing them to veer off course or crash. This attack involves transmitting fake GPS signals that override the vehicle's legitimate signals, leading to dangerous miscalculations [38].

V2X communication also relies on wireless signals that can be jammed, disrupting the flow of crucial information. In 2021, a jamming attack was demonstrated where multiple vehicles were rendered unable to receive or send data, causing significant traffic disruptions and even roadblocks. These incidents highlight the vulnerabilities in V2X communication protocols that could be exploited to create gridlock or force vehicles into unsafe conditions [39].

Additionally, relay attacks have been used to intercept and delay V2X messages, causing vehicles to react to out-

dated or incorrect information. This can lead to dangerous situations where vehicles may fail to stop at red lights or collide due to delayed braking signals [40].

6.4. Sensor Spoofing Attacks

Vehicle perception systems, including LiDAR, radar, and cameras, are vulnerable to sensor spoofing. Attackers have the ability to create deceptive signals that can be mistaken for genuine objects by these sensors. As an illustration, studies have shown that by projecting certain patterns onto road signs, the vehicle's image recognition system can be tricked into misinterpreting the sign, which could result in risky decisions. In a similar vein, through the use of laser spoofing, malicious individuals have the ability to manipulate LiDAR readings. This can result in the creation of deceptive obstacles or the concealment of genuine ones [41].

6.5. Malware and Software Exploits

Vehicle software can be targeted by malware attacks that take advantage of weaknesses in the operating system or third-party applications. As an example, in 2018, a group of experts made an important finding regarding a vulnerability in Tesla's Model S. This finding enabled them to remotely control the vehicle. The attack was carried out by taking advantage of a vulnerability in the vehicle's browser, which enabled the researchers to insert harmful code. Once inside, they had the ability to control various aspects of the car, including unlocking doors and even steering [42].

6.6. Man-in-the-Middle (MitM) Attacks

Intercepting and potentially altering communications between the vehicle and external systems is a common occurrence in MitM attacks. In 2020, a groundbreaking discovery was made when experts successfully showcased a method of intercepting V2X communication, resulting in a potential vulnerability for Tesla vehicles. The manipulation of sensor data led to the vehicle misinterpreting its surroundings. Such an attack has the potential to result in the vehicle receiving inaccurate information, such as incorrect speed limits or traffic signals, which can lead to unsafe driving behavior [43].

6.7. Denial-of-Service (DoS) Attacks

Denial of Service (DoS) attacks aim to incapacitate a vehicle's systems or communication channels by inundating them with excessive traffic. In a recent experiment conducted in 2019, a group of researchers successfully demonstrated a disruptive attack on a fleet of autonomous vehicles. The attack involved overwhelming the vehicles' communication channels with an excessive amount of traffic, causing them to experience a denial-of-service (DoS) situation. This incident resulted in a total disruption of vehicle-to-vehicle (V2V) communication, resulting in a breakdown of synchronized driving and the possibility of accidents [44].

7. Emerging Threats in 2023-2024

Recent advancements in AI and machine learning have introduced new dimensions to cybersecurity threats in driverless cars. The following are some of the most concerning threats identified in the past year:

7.1. Adversarial Machine Learning (AML) Attacks

As driverless cars increasingly rely on AI to make real-time decisions, they become targets for adversarial machine learning (AML) attacks. These attacks exploit vulnerabilities in AI models by introducing carefully crafted perturbations to the input data. These perturbations, often imperceptible to human observers, can cause the AI to make incorrect decisions, such as misclassifying road signs or obstacles. For instance, in 2023, researchers demonstrated that by placing inconspicuous stickers on stop signs, they could cause AI systems in driverless cars to misinterpret them as yield signs, leading to potentially dangerous outcomes. The technical complexity of AML attacks lies in the ability to generate perturbations that evade detection while consistently fooling the model across various scenarios, making them a significant threat to the safety and reliability of autonomous vehicles [45].

7.2. Supply Chain Attacks

The global supply chain for automotive components is vast and interconnected, making it a prime target for sophisticated cyberattacks. In 2024, a major automotive supplier was compromised through a multi-stage attack that involved the insertion of malware into firmware updates for critical components used in driverless cars. The malware was designed to remain dormant, undetected by traditional security measures, until specific conditions were met, at which point it would activate and allow remote attackers to take control of the affected vehicles. This attack exemplifies the growing complexity and reach of supply chain threats, where the integrity of software and hardware components can be compromised at any point along the supply chain, from manufacturing to distribution. The challenge in defending against these attacks lies in the need for comprehensive verification processes and the integration of secure development practices across all tiers of the supply chain [46].

7.3. Ransomware Targeting Vehicle Systems

Ransomware targeting AI-driven and autonomous vehicles represents an emerging cybersecurity challenge due to the complexity and connectivity of these systems. Such attacks could exploit vulnerabilities in the vehicle's software stack, potentially locking down critical systems like navigation, braking, or powertrain control. By encrypting the vehicle's control systems, ransomware [47] can render the car inoperable until a ransom is paid, usually in cryptocurrency. The decentralized nature of these attacks makes them difficult to trace, and the integration of AI increases the risk, as machine learning models can be manipulated or disrupted to exacerbate the impact. Additionally, the connected ecosystem of vehicles, with constant communication to cloud services and other devices, introduces multiple attack vectors, such as over-the-air updates, that can be compromised. This highlights the urgent need for multi-layered

security strategies, including robust encryption, anomaly detection systems, and secure software development practices to mitigate these risks [48].

7.4. Keyless Car Theft and Relay Attacks

The advent of key less entry and ignition systems in modern vehicles has introduced new security vulnerabilities in the automotive industry. These conveniences have become targets for sophisticated theft techniques, particularly man-in-the-middle attacks. Such attacks exploit the communication between cars and key fobs, allowing thieves to unlock and start vehicles without triggering alarms [49]. In 2023, the discovery of Bluetooth relay attacks on Tesla vehicles highlighted another critical weakness. This exploit took advantage of vulnerabilities in Bluetooth Low Energy (BLE) protocols used in passive entry systems. By relaying Bluetooth signals, attackers could access vehicles without the physical presence of the key fob. Although Tesla addressed this specific issue with a software update, the incident underscored a broader cybersecurity challenge [50].

7.5. Risks in Autonomous Vehicle Charging Infrastructure

As electric vehicles (EVs) have evolved, cybersecurity vulnerabilities in their charging infrastructure have become a significant concern. These issues are even more critical for autonomous vehicles (AVs), which also rely on secure data exchange with charging stations. The increased complexity and autonomy of AVs heighten the risks, including the potential for malware, fraud, remote manipulation, and disabling of charging stations. These vulnerabilities pose serious threats to the safety and integrity of the autonomous vehicle ecosystem [51].

7.6. Causing Traffic Jams

The interconnected nature of autonomous vehicles makes them vulnerable to coordinated cyberattacks that could lead to significant traffic disruptions. By compromising the control systems of multiple driverless cars simultaneously, an attacker could create artificial traffic jams. For instance, cars could be programmed to slow down or stop in strategic locations, blocking key intersections or highways [52]. Such disruptions could be used as a form of protest, to create chaos in urban environments, or as a precursor to other criminal activities, such as facilitating the escape of criminals from law enforcement by creating diversions.

7.7. Noise Pollution

In recent events, San Francisco residents experienced disturbances due to Waymo's autonomous vehicles persistently honking in residential areas during nighttime hours. This unexpected behavior, initially captured on video by local residents, drew significant public attention and prompted Waymo to respond with an apology and immediate corrective measures. According to Waymo, the honking feature was originally designed to minimize collision risks on public roads by signaling to other vehicles and pedestrians. However, when applied in confined parking areas, the feature led to excessive noise pollution, particularly during early morning hours, causing distress to the community [53]. This incident not only highlights the complexities and

unintended consequences of deploying autonomous vehicles in urban environments but also raises concerns about potential misuse. Autonomous vehicles' honking functions could be exploited by cybercriminals to create targeted noise pollution in densely populated areas. If an AV system were to be compromised, attackers could manipulate honking or other sound features, intensifying urban noise pollution and causing widespread disruption. This example underscores the need for robust cybersecurity measures that address not only operational functionality but also safeguard against potential abuses that could impact public well-being. As AV technology continues to advance, these considerations will be crucial for ensuring harmonious integration within urban settings.

8. Countermeasures for AV Cybersecurity Threats

Autonomous vehicles (AVs) encounter various cybersecurity threats that can jeopardize their safety, operational capabilities, and the confidence of the public. This section examines essential strategies aimed at reducing these risks through the improvement of AV system security against a range of potential attacks.

8.1. Adversarial Machine Learning (AML) Attacks

Adversarial Machine Learning (AML) attacks exploit weaknesses in AI models, causing them to make incorrect decisions. To counter AML attacks, AV developers can utilize several advanced defense techniques:

- **Adversarial Training:** This technique involves training AI models on adversarial examples—data intentionally perturbed to mislead the model—so that the model learns to recognize and resist such attacks. By exposing the model to these adversarial inputs during training, its robustness against real-world attacks is improved [54].

- **Defensive Distillation:** This approach reduces the model's sensitivity to adversarial perturbations by smoothing the decision boundaries, making it harder for subtle modifications to alter the AI's output [55].

- **Feature Squeezing:** By reducing the complexity of input data, feature squeezing eliminates extraneous features that could be exploited. Techniques such as image bit depth reduction or smoothing can mitigate adversarial input by stripping away noise and focusing on essential information [56].

- **Robustness Verification:** Formal methods and automated testing frameworks can be applied to verify the resilience of AI models against adversarial perturbations, providing assurances that the models perform reliably under a range of potential attack scenarios [57].

8.2. Supply Chain Attacks

The complexity of AV supply chains introduces vulnerabilities that can be targeted through various means. Addressing these requires both organizational and technical strategies:

- **Blockchain for Component Tracking:** Implementing blockchain can provide transparent and immutable records for each component's origin and journey through the supply chain, helping to ensure that only verified components are integrated into AV systems [58, 59].

- **Secure Boot and Hardware Roots of Trust:** Secure boot mechanisms ensure that AV systems only load authenticated software at startup. Hardware roots of trust further protect the integrity of the system by establishing a secure hardware foundation, preventing malware insertion at the hardware level [58].

- **Code Signing and Verification:** All software updates and components should be cryptographically signed, and systems must verify these signatures before applying any updates. This process ensures that only authorized and untampered code is executed within AV systems.

- **Comprehensive Vulnerability Assessments:** Regular security audits and vulnerability assessments are essential for identifying and mitigating potential supply chain weaknesses. By partnering with trusted third-party cybersecurity firms, manufacturers can gain additional insights into their supply chain's security posture [59].

8.3. Ransomware Attacks Targeting Vehicle Systems

Ransomware poses a growing threat to AV systems by potentially locking critical functionalities. Effective countermeasures include:

- **Strong Encryption and Backup Protocols:** Encrypting key data within the vehicle's system and maintaining secure, regularly updated backups can prevent data loss and facilitate recovery in case of a ransomware attack [47].

- **Endpoint Detection and Response (EDR):** Advanced EDR systems provide real-time monitoring of AV endpoints, detecting unusual patterns that may indicate ransomware activities. Rapid response capabilities enable the containment and neutralization of threats before they escalate.

- **Anomaly Detection Systems:** Employing machine learning models that analyze normal system behaviors can help detect ransomware activity. When abnormal behavior patterns are identified, such as unauthorized encryption attempts, the system can isolate affected modules to prevent further damage [47].

8.4. Keyless Car Theft and Relay Attacks

AVs with keyless entry systems are susceptible to relay attacks that intercept and amplify signals between the key fob and the vehicle. Countermeasures include:

- **Ultra-Wideband (UWB) Technology:** UWB-based systems offer enhanced accuracy in determining the proximity of the key fob to the vehicle, reducing the likelihood of successful relay attacks [60].

- **Two-Factor Authentication (2FA) for Access:** By requiring a secondary authentication step (such as a smartphone verification) [61], AVs can add an additional layer of security that limits unauthorized access.

- **Interference Detection Systems:** These systems monitor and detect signal anomalies associated with relay attacks, enabling AVs to trigger alerts or disable keyless entry temporarily when an attack is suspected.

8.5. Risks in Autonomous Vehicle Charging Infrastructure

AV charging stations are potential targets for cyberattacks that could disrupt vehicle operations. Mitigating these risks involves:

- **Transport Layer Security (TLS) Protocols:** Secure communication between AVs and charging stations through TLS [62] ensures data integrity and confidentiality, making it more challenging for attackers to intercept or manipulate information.

- **Network Segmentation and Firewalls:** Isolating charging stations from broader vehicle and grid networks prevents unauthorized access and contains attacks to specific segments, limiting the scope of potential damage.

- **Regular Cybersecurity Audits:** Continuous monitoring and assessment of charging infrastructure cybersecurity are necessary to identify and mitigate emerging threats, ensuring robust defenses against attacks.

8.6. Countermeasures for Traffic Jams and Noise Pollution

Cybercriminals could exploit AV functionalities to create traffic disruptions and noise pollution. To counter these risks:

- **Decentralized Control Protocols:** Implementing decentralized control systems helps distribute vehicle decision-making, reducing susceptibility to coordinated attacks that could cause traffic jams.

- **Honk Limiting Features and Scheduling:** AVs can incorporate software that restricts honking based on the time of day and location, reducing potential disturbances in residential areas and mitigating risks if systems are compromised.

By implementing these strategies, the AV industry can establish a strong cybersecurity framework that safeguards vehicles against a variety of changing threats, thereby ensuring the safety and security of passengers and urban settings.

9. Statistical Analysis

9.1. The Status of Self-Driving Cars

In 2023, the development of autonomous vehicle (AV) technology is still ongoing, and there are currently only a limited number of self-driving cars being used on U.S. roads. Many of these vehicles are currently undergoing testing to assess and improve their self-driving capabilities. Last year, around 1,400 vehicles, including cars, trucks, and

other types, were undergoing testing by 80 different companies in 36 states throughout the United States. In January 2023, Mercedes-Benz made history by becoming the first automaker in the U.S. to receive government approval for a Level 3 driving feature in Nevada [63]. This achievement marked a significant milestone in the automotive industry. Level 3 automation is of utmost importance as it signifies a remarkable leap forward from Level 2, enabling the vehicle to take care of all driving responsibilities while the driver only needs to step in when required. Despite the relatively small number of autonomous vehicles currently in use, the industry is witnessing significant expansion.

According to market analysis, the global autonomous vehicle (AV) market is expected to experience significant growth in the coming years. The market value is projected to increase from USD 1,921.1 billion in 2023 to USD 13,632.4 billion by 2030, with a compound annual growth rate (CAGR) of 32.3% during this period [64]. In 2022, the Asia-Pacific region emerged as the dominant player in the autonomous vehicle industry, capturing an impressive market share of 50.44%. According to these projections, the autonomous vehicle market is set to experience significant growth, fueled by advancements in technology and the growing acceptance of these vehicles in different regions.

9.2. Consumer Perception and Safety Concerns

Public trust and acceptance are pivotal to the widespread adoption of autonomous vehicles (AVs). While approximately 57% of Americans familiar with self-driving cars are willing to ride in them, and 55% believe that most vehicles will be autonomous by 2029, there remains a significant proportion of the population—43%—who express apprehension about using driverless vehicles [65]. Surveys indicate that safety concerns are at the core of this reluctance. For example, in 2024, AAA reported that about 66% of Americans harbor fears about fully autonomous vehicles due to incidents and perceived risks associated with these systems [66].

Additionally, a World Economic Forum report underscores the role of clear communication and regulatory transparency in enhancing public trust in AV technology. The report emphasizes that as people become more familiar with AV safety measures and technological advancements, their confidence is likely to grow, thereby facilitating a smoother transition to autonomous transportation systems [67]. Understanding and addressing these consumer concerns is crucial for real-world implementation, as public acceptance directly impacts adoption rates and can guide regulatory approaches to foster trust in AV technologies.

9.3. Automated Vehicle Accident Statistics

The safety of self-driving cars is a subject that sparks heated discussions, especially as the technology progresses. In 2022, automakers disclosed around 400 crashes to the National Highway Traffic Safety Administration (NHTSA) involving vehicles equipped with partially automated driver-assist systems. Out of all the incidents, 273 of them involved Tesla vehicles, and interestingly, 70% of these Teslas were using the Autopilot beta feature at the time. It is worth mentioning that 11 of the crashes reported led to severe injuries, while five of them tragically resulted in fatalities. Meanwhile, in 130 reported accidents involving fully autonomous vehicles, there were no injuries in 108 cases, with the majority of incidents being rear-end collisions [65].

9.4. Liability in Autonomous Vehicle Accidents

Assessing responsibility in accidents involving autonomous vehicles introduces an additional level of intricacy to conventional traffic incidents. When a self-driving car is involved in an accident, it becomes essential to determine if the automated driving system was active during the collision and the level of human driver intervention that was anticipated. If there is a suspicion of a flaw in the AV's system, the incident might be subject to product liability law [68]. In such cases, it is necessary to provide evidence of the defect and establish a clear link between the defect and the resulting injury.

9.5. Market and Technological Advancements

The autonomous vehicle industry is making significant strides, with continuous advancements in levels of vehicle automation. Various systems currently exist, with different levels of assistance provided to drivers. At the lowest level, drivers receive only momentary assistance, while at the highest level, the vehicle is capable of handling all driving tasks in any condition. Driver-assist technologies have become a common feature in the majority of new cars. These technologies encompass a range of helpful features such as automatic emergency braking, lane-keeping assistance, and adaptive cruise control [69]. As companies like Mercedes-Benz receive approval to deploy these technologies, more advanced systems, such as Level 3 automation, are becoming increasingly prevalent. These systems allow vehicles to manage most driving tasks independently.

10. Comparative Analysis: Drones vs. Driverless Cars

Technological breakthroughs in autonomous systems have propelled drones and autonomous vehicles to the forefront of contemporary innovation, providing unparalleled advantages in transportation, logistics, and surveillance. Nevertheless, the autonomy that fuels these advantages also brings about substantial cybersecurity vulnerabilities, which give rise to concerns over their possible abuse.

Drones [70] and autonomous vehicles have common technological underpinnings, such as the utilization of GPS for navigation, sensors for environmental perception, and artificial intelligence (AI) for decision-making operations. Both technologies are highly dependent on real-time data processing and external connectivity, rendering them vulnerable to cyber attacks such GPS spoofing, jamming, and unauthorized data interception.

Nevertheless, their operational environments vary greatly. Drones function within airspace and are commonly employed for the purposes of surveillance, delivery, and reconnaissance, rendering them susceptible to abuse in illicit activities such as smuggling, covert surveillance, or even as weaponry [70]. Conversely, autonomous vehicles skillfully maneuver through intricate metropolitan landscapes, prioritizing transit and logistics, where the vulnerabilities to hacks

encompass hijacking, remote control, and the potential for accidents. Understanding these potential threats is crucial for developing appropriate countermeasures.

Given these distinct yet overlapping vulnerabilities, it is essential to explore the specific ways in which both drones and AVs can be exploited by cybercriminals. The following subsections delve into these potential threats, comparing how they manifest in drones and AVs.

10.1. Tracking and Surveillance

Driverless cars can be misused as tools for tracking and surveillance. Since these vehicles are equipped with GPS, cameras, and other sensors, a compromised car can be used to monitor an individual's movements without their knowledge. Cybercriminals or other malicious entities could hijack the car's systems to gather real-time data about the location and activities of the occupants, effectively turning the vehicle into a surveillance tool. While drones are also susceptible to being used for surveillance, their typical use in airspace makes them less likely to be involved in long-term tracking within dense urban environments, where AVs are more commonly found. This kind of misuse raises significant privacy concerns, especially if sensitive personal information is intercepted or recorded.

10.2. Transporting Illegal Items

Driverless cars could be exploited for the unauthorized transportation of illegal goods, such as drugs, weapons, or contraband. Since these vehicles operate autonomously, they could be programmed to follow predetermined routes to specific drop-off locations, reducing the need for human involvement and lowering the risk of detection.

Drones, too, are vulnerable to such misuse, particularly for smuggling goods across borders or into restricted areas. However, the ground-based nature of AVs means they are more likely to be used for covert operations within urban areas. This capability could be particularly appealing to criminal organizations looking to minimize the risks associated with traditional methods of smuggling and transportation.

10.3. Weaponization and Terrorism

In the worst-case scenarios, driverless cars could be weaponized by cybercriminals or terrorists. By hijacking a vehicle's control systems, an attacker could direct the car to drive into a crowded area or a critical piece of infrastructure, such as a bridge or a building. The vehicle could also be loaded with explosives or hazardous materials, turning it into a remote-controlled bomb. While drones have similarly been considered for such malicious purposes due to their aerial capabilities, the ground-based nature of autonomous vehicles allows them to carry larger payloads and target densely populated urban environments more effectively. The potential for such attacks highlights the importance of securing the software and communication systems of autonomous vehicles to prevent their use in acts of terrorism.

10.4. Coordinated Cyberattacks

Driverless cars, like drones, are part of a broader network of interconnected systems, including other vehicles, traffic management systems, and cloud-based services. This interconnectivity increases the risk of coordinated cyberattacks [71], where multiple vehicles are compromised and used in concert to achieve a malicious goal. Such attacks could disrupt urban infrastructure, create widespread panic, or even be used as a tool for cyber warfare. The ability to remotely control large numbers of autonomous vehicles presents a unique challenge to cybersecurity professionals, who must develop strategies to detect and neutralize such threats before they can cause harm. For instance, the Ukrainian government has developed advanced drone systems capable of coordinated attacks [72]. These drones can intercommunicate, autonomously decide on targets, and collect intelligence at speeds surpassing human capabilities. This development illustrates the potential for autonomous systems to be used in coordinated actions, highlighting the need for robust cybersecurity measures in both military and civilian autonomous vehicle networks.

11. Societal Impacts

11.1. Transportation Accessibility

Driverless cars have the potential to significantly improve transportation accessibility for individuals with disabilities, the elderly, and those who cannot drive. Autonomous vehicles (AVs) can provide on-demand mobility services, enhancing independence and quality of life for these populations. For example, AVs could offer new mobility options to the approximately 49 million Americans over the age of 65 and the 53 million people with some type of disability. This increase in accessibility could result in a 14% rise in vehicle miles traveled, potentially worsening congestion but significantly improving individual mobility and independence [73]. Furthermore, automated vehicles could create new employment opportunities for about 2 million people with disabilities in the United States [74].

11.2. Urban Planning and Traffic Management

Autonomous vehicles are expected to have a profound impact on urban planning and traffic management. By reducing the need for extensive parking facilities and enabling more efficient use of road space, AVs can contribute to better urban environments. Improved traffic management through vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication can reduce congestion and optimize traffic flow. Platooning, where vehicles travel closely together, can increase road capacity and reduce traffic delays [73]. Additionally, shared mobility services facilitated by AVs can support car-sharing concepts, reducing the number of privately owned vehicles and promoting more sustainable urban transport models [74].

11.3. Environmental Considerations

The adoption of electric autonomous vehicles can significantly contribute to environmental sustainability by reducing greenhouse gas emissions and improving air quality. Efficient driving patterns and reduced congestion lead to lower fuel consumption and environmental impact. According to the U.S. Chamber of Commerce, autonomous vehicles

have the potential to prevent 1.4 million accidents and 12,000 fatalities annually, resulting in substantial economic savings of $94 billion [75]. Moreover, the shift towards shared autonomous vehicles can further reduce the overall number of vehicles on the road, decreasing the carbon footprint associated with manufacturing and operating personal vehicles.

11.4. Job Displacement and Economic Implications

The transition to driverless cars may lead to job displacement in industries such as trucking, taxi services, and public transportation. In the United States, approximately 2.9% of workers are employed in driving occupations, with more than 4 million potentially affected by the adoption of AVs. This could result in an estimated annual income loss of around $180 billion for these workers [73]. However, the rise of autonomous vehicles also presents opportunities for economic growth, including new jobs in technology development, infrastructure upgrades, and mobility services. The motor insurance industry, worth over $300 billion annually in the U.S., may also experience significant changes as AVs reduce the frequency and severity of accidents, potentially lowering insurance premiums for consumers [73].

12. Regulatory Landscape

12.1. Government Initiatives and Policies

Governments worldwide are actively developing initiatives and policies to support the deployment of autonomous vehicles (AVs). These efforts include funding for research and development, establishing testing and certification standards, and creating frameworks for AV integration into existing transportation systems. For instance, in the United States, the National Highway Traffic Safety Administration (NHTSA) continues to lead federal efforts, focusing on creating guidelines that ensure safety while promoting innovation. Various states are also pioneering regulatory efforts by passing laws and executive orders to facilitate AV testing and deployment, highlighting the importance of harmonized federal regulations to avoid a fragmented regulatory landscape.

12.2. Legal Framework for Autonomous Vehicles

Establishing a robust legal framework is essential for the safe and effective deployment of AVs. Key components of this framework include:

1. **Defining Liability**: Clarifying liability in the event of accidents involving AVs is crucial. This involves determining the responsibility between manufacturers, software developers, and vehicle operators. Some states, like California, Nevada, and Florida, have implemented requirements for manufacturers to hold significant insurance policies or bonds to cover potential damages.

2. **Setting Standards for Performance and Safety**: Governments are establishing standards for vehicle performance and safety to ensure that AVs can operate safely on public roads. These standards cover aspects such as sensor accuracy, fail-safe mechanisms, and emergency response capabilities.

3. **Ensuring Data Privacy Compliance**: As AVs collect and process vast amounts of data, ensuring compliance with data privacy regulations is essential to protect user information and maintain public trust.

12.3. International Standards and Collaboration

International collaboration and the establishment of global standards are crucial for the interoperability and scalability of autonomous vehicle technology. Organizations such as the International Organization for Standardization (ISO) and SAE International play key roles in developing these standards. For example, the European Union has finalized a legal framework for fully automated vehicles, creating binding regulations that include comprehensive safety and performance criteria. Collaborative efforts are also evident in cross-border initiatives, where countries work together to create harmonized regulatory approaches, enabling seamless operation of AVs across different jurisdictions.

12.4. Comparative Regulatory Frameworks in Europe and Asia

European and Asian countries have developed distinctive regulatory frameworks that address the deployment and operation of AVs, reflecting diverse regional priorities and challenges. In the **European Union (EU)**, regulations such as the General Safety Regulation mandate that all new vehicles include advanced driver assistance systems (ADAS) by 2022, encompassing features like lane-keeping and automated emergency braking. Furthermore, *UN Regulation 157* on Automated Lane Keeping Systems (ALKS) permits Level 3 AVs on public roads, provided they meet strict safety, cybersecurity, and software update requirements. Germany, a leader within the EU, enacted a law in 2021 allowing Level 4 AVs to operate in specified public areas, setting a precedent for comprehensive AV legislation [76].

In the **United Kingdom**, efforts to position itself as an AV innovation hub are underscored by significant investments exceeding £200 million in AV testing and research infrastructure. The UK's *Code of Practice for Testing Automated Vehicles* provides guidelines on liability, insurance, and data protection, facilitating AV trials on public roads. By 2025, the UK aims to establish a legal framework to support widespread AV deployment, focusing on safety standards and clarifying manufacturer and user responsibilities [77].

In **Asia**, Japan has prioritized AV integration into public transport, especially in rural areas, and revised the *Road Transport Vehicle Law* to permit Level 3 AVs on public roads. Japan is actively preparing for the *2025 World Expo* in Osaka, where Level 4 AVs are expected to be deployed [78]. China, another major player, has developed city-specific policies, allowing for Level 4 testing in cities like Beijing and Shanghai. The *Beijing Autonomous Driving Policy* includes data security requirements, emphasizing compliance with national laws like the *Cybersecurity Law* and the *Personal Information Protection Law (PIPL)* [79]. South Korea, aspiring to be an AV industry leader, has implemented the *Framework Act on Intelligent Robots* to regulate AVs, along with establishing *K-City*, a large-scale AV testing facility simulating urban and highway environments [80].

13. Results

In this section, the primary findings of the study are presented, with a particular emphasis on the comparative analysis of cybersecurity threats facing autonomous vehicles (AVs) and drones. By examining the common vulnerabilities between these two technologies, this research sheds light on potential threats that AVs may encounter, drawing parallels from the established cybersecurity issues in drone systems. This cross-technology perspective provides valuable insights into emerging risks and underscores the importance of proactive threat mitigation.

Additionally, a comparison with recent related works is included to underscore the unique contributions of this study. Table 2 summarizes how this research diverges from existing literature by specifically focusing on emerging and future cybersecurity threats in AVs, inferred from drone vulnerabilities. This approach offers a forward-looking view on cybersecurity challenges, providing a foundation for understanding and anticipating potential misuse of AV technology.

Table 2: Threats in Autonomous Vehicles with Status

Threat	Potential Impact on AVs	Status
Remote Exploits	Unauthorized control over AV functions, posing safety risks	Existing
GPS Spoofing	Misleading AV navigation, causing route deviations	Emerging
Adversarial ML Attacks	Misinterpretation of road signs or obstacles, leading to unsafe decisions	Emerging
Supply Chain Attacks	Risk of tampered AV components before deployment	Emerging
Ransomware	Potential to incapacitate AV systems until payment is made	Possible Future
Denial of Service (DoS)	System overload, disrupting AV operations and real-time data processing	Possible Future
Noise Pollution	Excessive AV honking and increased vehicle numbers could heighten urban noise levels	Possible Future
Traffic Congestion	More AVs on roads could worsen traffic if not managed effectively	Possible Future
Risks in AV Charging Infrastructure	Cyberattacks on charging stations, grid strain, and bottlenecks at charging sites	Possible Future
Tracking and Surveillance	Unauthorized tracking using AVs for long-term monitoring within urban areas	Emerging
Transporting Illegal Items	Use of AVs for transporting contraband autonomously within urban settings	Possible Future
Weaponization and Terrorism	Hijacking AVs for attacks or as remote-controlled bombs targeting urban areas	Possible Future
Coordinated Cyberattack	Large-scale attacks involving multiple AVs to disrupt traffic or emergency response systems	Possible Future

14. Limitations

While this paper offers a broad survey of autonomous vehicle (AV) technology, it is limited by its high-level perspective, which does not delve into specific implementation details. The generalized nature of this study may not fully reflect the unique technological approaches of different manufacturers, thereby limiting the applicability of its findings across various AV systems. Additionally, due to the rapid pace of advancements in AV technology, some insights presented here may quickly become outdated as new innovations in sensors, AI algorithms, and cybersecurity measures continue to evolve.

Furthermore, the study focuses mainly on technological and cybersecurity challenges, with less emphasis on the ethical and social implications of AV adoption. Complex issues, such as the potential impact on employment, ethical

considerations in AI-driven decision-making, and regional differences in regulatory readiness, are acknowledged but not explored in depth. Future research could address these areas by conducting case studies, longitudinal analyses, and interdisciplinary investigations that incorporate ethical, social, and regional dimensions, providing a more holistic view of the challenges and opportunities surrounding autonomous vehicles.

15. Conclusion

Autonomous vehicles (AVs) signify a profound paradigm shift within the transportation sector, bearing both theoretical and practical ramifications. From a theoretical standpoint, AVs provide insights into the interplay of artificial intelligence, sensor integration, and real-time decision-making within complex systems. On a practical level, they present promising strategies to enhance road safety, alleviate traffic congestion, and mitigate the environmental consequences associated with conventional transportation. This study illustrates how the convergence of advanced sensors, AI algorithms, and vehicle-to-everything (V2X) communication technologies is advancing the AV landscape.

A key contribution of this research is the delineation of essential technologies and the examination of cybersecurity challenges that AVs encounter. By analyzing these elements, this study highlights the imperative need for robust cybersecurity frameworks designed to safeguard AV systems against potential threats. Furthermore, the study underscores the necessity for comprehensive regulatory frameworks that can foster public trust and support the widespread adoption of AV technology.

From an applied perspective, AVs are poised to deliver numerous benefits. They hold the potential to drastically reduce road fatalities attributable to human error, which remains a leading cause of vehicular accidents worldwide. In addition, AVs can enhance traffic management by optimizing routing and facilitating communication between vehicles and infrastructure, thereby promoting more efficient road usage. Moreover, the shift towards electric AVs offers considerable potential for reducing greenhouse gas emissions and supporting broader environmental sustainability goals.

Nevertheless, this research acknowledges certain limitations. Although it provides a comprehensive overview of AV technology and its associated challenges, the study does not delve into the specific implementation variations across different AV manufacturers. Additionally, given the rapid pace of technological advancements in this domain, the findings presented are susceptible to obsolescence. Future research could address these limitations by conducting longitudinal studies to track the ongoing evolution of AV technologies.

Looking forward, several avenues for future research merit attention. First, there is a pressing need for the development of standardized cybersecurity protocols that address the specific challenges posed by real-time data exchange and autonomous decision-making in AVs. Second, further research should explore the socio-economic impacts of AV proliferation, particularly concerning potential job displacement within the driving sector. Finally, interdisciplinary studies that integrate perspectives from urban planning, ethics, and AI safety are essential to comprehensively under-

stand how AVs will reshape transportation infrastructure and societal frameworks.

In conclusion, autonomous vehicles offer a transformative opportunity to develop transportation systems that are safer, more efficient, and more sustainable. Addressing the current challenges will necessitate ongoing collaboration among policymakers, industry stakeholders, and the research community. By surmounting these obstacles, AVs have the potential to become a fully integrated aspect of modern society, fundamentally altering the way we navigate our world for generations to come.

References

[1] T. Kanade, C. Thorpe, W. Whittaker, "Autonomous land vehicle project at cmu", "Proceedings of the 1986 ACM Fourteenth Annual Conference on Computer Science", CSC '86, pp. 71–80, ACM, New York, NY, USA, 1986, doi:10.1145/319838.319850.

[2] Mobileeye, "History of autonomous vehicles: From renaissance to reality", https://shorturl.at/asq4I, 2023, accessed: 2024-07-05.

[3] Arrow, "History of self-driving cars", https://www.arrow.com/en/research-and-events/articles/the-history-of-self-driving-cars, accessed: 2024-07-05.

[4] S. Magazine, "The national automated highway system that almost was", https://shorturl.at/UbL7x, 2013, accessed: 2024-07-05.

[5] Google, "Waymo: On the road", https://waymo.com/ontheroad/, 2017, accessed: 2024-07-05.

[6] G. V. Research, "Autonomous vehicle market to reach $214.32bn by 2030", https://www.grandviewresearch.com/industry-analysis/autonomous-vehicle-market, 2023, accessed: 2024-07-05.

[7] McKinsey and Company, "Autonomous vehicles moving forward: Perspectives from industry leaders", 2024, article.

[8] K. Bian, G. Zhang, L. Song, "Security in use cases of vehicle-to-everything communications", "2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)", pp. 1–5, 2017, doi:10.1109/VTCFall.2017.8288208.

[9] F. B. Insights, "Autonomous vehicle market size, share, trends | report [2030]", https://www.fortunebusinessinsights.com/autonomous-vehicle-market-102020, 2023, accessed: 2024-07-05.

[10] P. Research, "Autonomous vehicle market size to hit usd 2,752.80 bn by 2033", https://www.precedenceresearch.com/autonomous-vehicle-market, 2023, accessed: 2024-07-05.

[11] Precedence Research, "Autonomous vehicle market size to hit usd 2,752.80 bn by 2033", https://www.precedenceresearch.com/autonomous-vehicle-market#:~:text=The%20global%20autonomous%20vehicle%20market,USD%2059.92%20billion%20in%202023, 2024, last updated: June 2024.

[12] W. H. Organization, "Global status report on road safety 2018", *World Health Organization*, 2018, accessed: 2024-07-05.

[13] WHO, "Global status report on road safety 2023", *World Health Organization*, 2023, accessed: 2024-07-05.

[14] W. H. Organization, "Road traffic injuries", *World Health Organization*, 2023, accessed: 2024-07-05.

[15] Spectrum, "Autonomous vehicles can make all cars more efficient", https://shorturl.at/9ht7y, 2018, accessed: 2024-07-05.

[16] McKinsey, Company, "The future of autonomous vehicles (av)", https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/the-future-of-autonomous-vehicles-av, 2023, accessed: 2024-07-05.

[17] ScienceDirect, "The impacts of connected autonomous vehicles on mixed traffic flow: A review", https://www.sciencedirect.com/science/article/pii/S2352146520305170, 2020, accessed: 2024-07-05.

[18] N. Geographic, "Energy implications of autonomous vehicles: Imagining the possibilities", https://www.nationalgeographic.com/environment/article/energy-implications-of-autonomous-vehicles, 2019, accessed: 2024-07-05.

[19] B. C. Group, "Can self-driving cars stop the urban mobility meltdown?", https://shorturl.at/F0d5d, 2020, accessed: 2024-07-05.

[20] R. Du, *et al.*, "Effective urban traffic monitoring by vehicular sensor networks", *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 273–286, 2014, doi:10.1109/TVT.2014.2349320.

[21] Y. Li, *et al.*, "Vehicle detection based on the and–or graph for congested traffic conditions", *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 984–993, 2018, doi:10.1109/TITS.2018.2871040.

[22] R. F. Foundation, "Self-driving cars: The impact on people with disabilities", *Ruderman White Paper*, 2017.

[23] R. Corporation, "Self-driving vehicles offer potential benefits, policy challenges for lawmakers", *RAND Corp*, 2014.

[24] N. C. Onat, J. Mandouri, M. Kucukvar, *et al.*, "Rebound effects undermine carbon footprint reduction potential of autonomous electric vehicles", *Nature Communications*, vol. 14, p. 6258, 2023, doi:10.1038/s41467-023-41992-2, received 01 May 2023, Accepted 22 September 2023, Published 06 October 2023.

[25] A. J. G. Rodríguez, N. J. Barón, J. M. G. Martínez, "Validity of dynamic capabilities in the operation based on new sustainability narratives on nature tourism smes and clusters", *Sustainability*, vol. 12, no. 3, p. 1004, 2020, doi:10.3390/su12031004.

[26] Y. Zhang, Z. Wang, H. Wang, F. Blaabjerg, "Artificial intelligence-aided thermal model considering cross-coupling effects", *IEEE Transactions on Power Electronics*, vol. 35, no. 10, pp. 9998–10002, 2020, doi:10.1109/TPEL.2020.2980240.

[27] KPMG, "Connected and autonomous vehicles - readiness index", https://t.ly/_tAW_, 2020, accessed: 2024-07-05.

[28] SMMT, "Connected and autonomous vehicles: The global race to market", https://tinyurl.com/4kdakunk, 2021, accessed: 2024-07-05.

[29] R. Domínguez, E. Onieva, J. Alonso, J. Villagra, C. González, "Lidar based perception solution for autonomous vehicles", "Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on", pp. 790–795, 2011.

[30] E. Hasch, R. Topak, R. Schnabel, T. Zwick, R. Weigel, C. Waldschmidt, "Millimeter-wave technology for automotive radar sensors in the 77 ghz frequency band", *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, no. 3, pp. 845–860, 2012, doi:10.1109/TMTT.2011.2178427.

[31] M.-H. Lee, Y.-J. Chen, T. Li, "Sensor fusion design for navigation and control of an autonomous vehicle", "2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC)", pp. 2209–2214, 2011, doi:10.1109/ICSMC.2011.6084000.

[32] Q. Li, L. Chen, M. Li, S.-L. Shaw, A. Nuchter, "A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios", *IEEE Transactions on Vehicular Technology*, vol. 63, no. 2, pp. 540–555, 2014, doi:10.1109/TVT.2013.2289913.

[33] B. Padmaja, C. Moorthy, N. Venkateswarulu, *et al.*, "Exploration of issues, challenges and latest developments in autonomous cars", *Journal of Big Data*, vol. 10, no. 61, 2023, doi:10.1186/s40537-023-00701-y, published: 06 May 2023.

[34] L. Hobert, A. Festag, I. Llatser, L. Altomare, F. Visintainer, A. Kovacs, "Enhancements of v2x communication in support of cooperative autonomous driving", *IEEE Communications Magazine*, vol. 53, pp. 64–70, 2015, doi:10.1109/MCOM.2015.7105641.

[35] S. A. Centers, "The six levels of autonomous driving", https://www.schaeferautobody.com/the-six-levels-of-autonomous-driving/#:~:text=Conditional%20Automation%20(Level%203)%20Vehicles%20at%20this,to%20monitor%20for%20changes%20in%20those%20conditions, 2019, accessed: October 11, 2024.

[36] A. Greenberg, "Hackers remotely kill a jeep on the highway—with me in it", https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/, 2015, accessed: 2024-08-26.

[37] The Guardian, "Uber's joe sullivan faces trial over data breach cover-up", *The Guardian*, 2022, accessed: 2024-08-30.

[38] H. Shin, D. Won, S. Kim, "Gps spoofing attack on autonomous vehicles and its countermeasure", *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1738–1748, 2019, doi:10.1109/TITS.2019.2896358.

[39] J. Kenney, "Jamming attack vulnerabilities in v2x communication systems", "IEEE Vehicular Technology Conference (VTC2021-Fall)", 2021, doi:10.1109/VTCFall2021.2019.9198881.

[40] A. Hamid, P. Lin, R. Hussain, "Relay attacks in vehicular networks: Analysis, impact, and solutions", *IEEE Communications Magazine*, vol. 60, no. 4, pp. 79–85, 2022, doi:10.1109/MCOM.2022.3056782.

[41] Q. Yan, D. Zhang, "Sensor spoofing attacks on autonomous vehicles: A review of the vulnerabilities and mitigation strategies", *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2835–2856, 2020, doi:10.1109/COMST.2020.2976336.

[42] I. Foster, K. Koscher, "Tesla model s: A case study in software security and over-the-air updates", *Journal of Cybersecurity*, vol. 6, no. 1, pp. 1–15, 2018, doi:10.1093/cybsec/tyy010.

[43] J. Petit, S. Shladover, "Mitm attacks on connected vehicles: Techniques, impact, and countermeasures", *IEEE Vehicular Technology Magazine*, vol. 15, no. 2, pp. 70–77, 2020, doi:10.1109/MVT.2020.2983621.

[44] M. Groll, S. R. Weller, "Denial-of-service attacks in autonomous vehicle networks: A comprehensive analysis", *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 3, pp. 400–412, 2019, doi:10.1109/TIV.2019.2937765.

[45] T. B. Brown, D. Mane, A. Roy, M. Abadi, J. Gilmer, "Adversarial patch", https://arxiv.org/abs/1907.07736, 2019, accessed: 2024-08-26.

[46] U.S. News and World Report, "Car dealerships are being disrupted by a multi-day outage after cyberattacks on software supplier", https://shorturl.at/1bgnQ, 2024, accessed: 2024-08-30.

[47] S. C. Nayak, V. Tiwari, B. K. Samanthula, "Review of ransomware attacks and a data recovery framework using autopsy digital forensics platform", "2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)", pp. 0605–0611, 2023, doi:10.1109/CCWC57344.2023.10099169.

[48] L. Eliot, "Here's how ransomware is going to fiendishly impede ai self-driving cars", https://shorturl.at/ptDMH, 2021, accessed: 2024-08-30.

[49] Nahla Davies, "Keyless car theft", https://shorturl.at/d5PLL, 2023, accessed: 2024-08-30.

[50] Alan J, "Bluetooth relay attacks", https://thecyberexpress.com/tesla-ultra-wideband-vulnerable-relay-attacks/, 2024, accessed: 2024-08-30.

[51] David Strom, "Ev charging stations still riddled with cybersecurity vulnerabilities", https://shorturl.at/vpv6y, 2024, accessed: 2024-08-30.

[52] Chris Isidore, "Driverless cars flood san francisco: What happens when things go wrong?", https://www.cnn.com/2023/08/14/business/driverless-cars-san-francisco-cruise/index.html, 2023, accessed: 2024-08-30.

[53] J. Marcus, "San francisco residents fed up with self-driving car honking", https://shorturl.at/zQT5p, 2024, accessed: 2024-10-10.

[54] I. J. Goodfellow, J. Shlens, C. Szegedy, "Explaining and harnessing adversarial examples", *arXiv preprint arXiv:1412.6572*, 2015, adversarial training improves model robustness by training AI models on adversarial examples.

[55] N. Papernot, P. McDaniel, X. Wu, S. Jha, A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks", *IEEE Symposium on Security and Privacy*, 2016, defensive distillation smooths decision boundaries to mitigate the impact of adversarial inputs.

[56] W. Xu, D. Evans, Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks", *arXiv preprint arXiv:1704.01155*, 2018, feature squeezing reduces input complexity to counter adversarial attacks.

[57] X. Huang, M. Kwiatkowska, S. Wang, M. Wu, "Safety verification of deep neural networks", *arXiv preprint arXiv:1610.06940*, 2017, robustness verification through formal methods ensures AI model resilience against adversarial perturbations.

[58] IEEE Smart Cities, "Secure boot and hardware roots of trust in autonomous vehicles", https://smartcities.ieee.org/security/autonomous-vehicle-security, 2023, accessed: 2023-10-11.

[59] IBM - United States, "Ibm blockchain technology", https://www.ibm.com/blockchain/supply-chain, 2023, accessed: 2023-10-11.

[60] Y. Rahayu, T. A. Rahman, R. Ngah, P. Hall, "Ultra wideband technology and its applications", "2008 5th IFIP International Conference on Wireless and Optical Communications Networks (WOCN '08)", pp. 1–5, 2008, doi:10.1109/WOCN.2008.4542537.

[61] S. Ibrokhimov, K. L. Hui, A. Abdulhakim Al-Absi, h. j. lee, M. Sain, "Multi-factor authentication in cyber physical system: A state of art survey", "2019 21st International Conference on Advanced Communication Technology (ICACT)", pp. 279–284, 2019, doi:10.23919/ICACT.2019.8701960.

[62] D. Zelle, C. Krauß, H. Strauß, K. Schmidt, "On using tls to secure in-vehicle networks", "Proceedings of the 12th International Conference on Availability, Reliability and Security", ARES '17, Association for Computing Machinery, New York, NY, USA, 2017, doi:10.1145/3098954.3105824.

[63] Mercedes-Benz Group, "Drive pilot: Nevada becomes first us state to approve sae level 3 system for mercedes-benz", https://group.mercedes-benz.com/innovation/product-innovation/autonomous-driving/drive-pilot-nevada.html, 2024, accessed: 2024-08-30.

[64] Fortune Business Insights, "Autonomous vehicle market size, share and covid-19 impact analysis, by component (hardware and software), by type (passenger car and commercial vehicle), and regional forecast, 2023-2030", https://www.fortunebusinessinsights.com/autonomous-vehicle-market-109045, 2024, accessed: 2024-08-30.

[65] KNR Legal, "Self-driving car accident statistics", https://www.knrlegal.com/car-accident-lawyer/self-driving-car-accident-statistics/, 2024, accessed: 2024-08-30.

[66] AAA, "2024 aaa survey on autonomous vehicles", https://newsroom.aaa.com/2024/05/aaa-survey-autonomous-vehicles/, 2024, accessed: 2024-10-10.

[67] W. E. Forum, "Which trends are driving the autonomous vehicles industry?", https://www.weforum.org, accessed: 2024-07-05.

[68] Byrd Davis Alden and Henrichson, LLP, "Who is liable when a self-driving car causes a crash?", https://shorturl.at/yZ7nY, 2024, accessed: 2024-08-30.

[69] Alliance for Automotive Innovation, "Autonomous vehicles", https://www.autosinnovate.org/initiatives/innovation/autonomous-vehicles, 2024, accessed: 2024-08-30.

[70] S. C. Nayak, B. K. Samanthula, V. Tiwari, "Investigating drone data recovery beyond the obvious using digital forensics", "2023 IEEE 14th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)", pp. 0254–0260, 2023, doi:10.1109/UEMCON59035.2023.10315995.

[71] Cybersecurity and Infrastructure Security Agency (CISA), "Action guide for cybersecurity: Critical control actions", https://www.cisa.gov/sites/default/files/2022-11/Action%20Guide%20CCA%20508%20FINAL%2020190905.pdf, 2022, accessed: 2024-08-30.

[72] Kyiv Post, "Ukraine's tech hub develops ai-driven drone swarms to combat russian forces", https://www.kyivpost.com/post/34777, 2024, accessed:2024-08-30.

[73] F. Klaver, "The economic and social impacts of fully autonomous vehicles", https://www.compact.nl, 2021, accessed: 2024-07-05.

[74] Bosch, "Impact of self-driving cars on society", https://www.bosch.com, accessed: 2024-07-05.

[75] U. Chamber, "New u.s. chamber report on economic and social benefits of autonomous vehicles", https://www.uschamber.com, accessed: 2024-07-05.

[76] European Commission, "Commission adopts new rules to support safe deployment of automated and connected vehicles in the eu", https://ec.europa.eu/commission/presscorner/detail/en/ip_22_4312, 2022, accessed: 2023-10-11.

[77] United Nations Economic Commission for Europe (UNECE), "Un regulation on automated lane keeping systems (alks) extended to trucks, buses and coaches", https://unece.org/automated-lane-keeping-system-alks, 2021, https://unece.org/automated-lane-keeping-system-alks.

[78] Japan Times, "Kishida administration to boost self-driving car development in japan", https://www.japantimes.co.jp/news/2024/08/01/japan/kishida-self-driving-cars-boost/, 2024, accessed: 2023-10-11.

[79] T. Staff Reporter, W. Ke, "China accelerates autonomous driving via multiple pilot cities, with beijing, shanghai, wuhan leading the charge", https://t.ly/sMNut, 2024, accessed: 2023-10-11.

[80] Applied Intuition, "How the korean police science institute uses simian and basis to validate av stack performance", https://www.appliedintuition.com/news/korea-police-science-institute, 2023, accessed: 2023-10-11.

Vaibhavi Tiwari holds a MicroMasters credential in Data and Statistics from the Massachusetts Institute of Technology (MIT). She completed her bachelor's degree in Computer Applications from an esteemed institution in India and earned a double master's degree: one in Computer Applications from NITK Surathkal and another in Computer Science from Montclair State University. With over eight years of experience in healthcare technology, Vaibhavi specializes in data security, AI integration, and Cloud based technologies. Currently, she is the Head of Technology and Engineering at MyGreen Health, where she leads the development of innovative healthcare solutions.

Vaibhavi's research lies at the intersection of cybersecurity, healthcare, and big data. She has published extensively in IEEE conferences, with recent works including "Review of Ransomware Attacks and a Data Recovery Framework using Autopsy Digital Forensics Platform" and "Investigating Drone Data Recovery Beyond the Obvious Using Digital Forensics." These studies underscore her commitment to tackling security challenges within healthcare and other critical sectors. Her contributions emphasize proactive threat identification and mitigation strategies, showcasing her expertise in securing complex, data-driven environments. Her accolades include being a certified Globee Awards judge, which highlights her contributions to technology and business evaluation. Vaibhavi's work spans research, professional development, and community service, such as her mentorship in the Women in Big Data program and her active volunteer efforts for the ISEF.

On a Kernel k-Means Algorithm

Bernd-Jürgen Falkowski[*]

Fachhochschule für Ökonomie und Management, BWL, Wirtschaftsinformatik, Arnulfstrasse 30, D-80335 München, Germany
[*]Corresponding author: Bernd-Jürgen Falkowski & Email: bernd.falkowski@hochschule-stralsund.de

ABSTRACT: This is the extended version of a paper presented at CISP-BMEI 2023.
After a general introduction kernels are described by showing how they arise from considerations concerning elementary geometrical properties. They appear as generalizations of the scalarproduct that in turn is the algebraic version of length and angle. By introducing the Reproducing Kernel Hilbert Space it is shown how operations in a high dimensional feature space can be performed without explicitly using an embedding function (the "kernel trick"). The general section of the paper lists some kernels and sophisticated kernel clustering algorithms. Thus the continuing popularity of the k-means algorithm is probably due to its simplicity. This explains why an elegant version of a k-means iterative algorithm originally established by Duda is treated. This was extended to a kernel algorithm by the author. However, its performance still heavily depended on the initialization. In this paper previous results on the original k-means algorithm are transferred to the kernel version thus removing these setbacks. Moreover the algorithm is slightly modified to allow for an easy quantification of the improvements to the target function after initializaction.

KEYWORDS: Clustering, K-means Algorithm, Kernels

1. Introduction

Clustering algorithms are not new. They have played a prominent part in the area of information retrieval, where searching for a text that was similar to a given text was often a difficult task. This lead to the development of various similarity measures. With the rise of the Internet the importance of clustering became obvious. Potential customers of Internet shops had to be segmented into several classes in order to customize advertising.

Information about texts or customers was frequently stored in vectors. Hence similarity measures for vector spaces had to be constructed. Whilst at first the geometrical concepts of *length* and *angle* gave rise to primitive measures, it was soon realized that a purely algebraic description of these concepts was needed. This led to the *scalar product* and thus to the first primitive kernels. Kernels, however, had been known and employed mainly in the context of Probability Theory and Statistics. A systematic treatment within the realm of Artificial Intelligence does not seem to have appeared before the beginning of the century.

It was soon realized that in order to provide added flexibility it would be advantageous to embed the original vector space into a higher dimensional *feature space, .* However, it was also obvious that this would create complexity problems. Fortunately enough a solution could be provided by using an abstract construction, namely the Kernel Reproducing Hilbert Space (KRHS).

These considerations lead to the following outline of the article. In section 2 kernels are introduced starting from first principles. En passant two simple similarity measures are described and the section ends with a description of a kernel that does not even require a vector space structure.

In section 3 the Kernel Reproducing Hilbert Space (RKHS) is introduced. It involves a very abstract construction whose usefulness is not immediately obvious. Thus in section 4 a simple kernel explicitly shows that embedding the original vector space in a high dimensional feature space causes complexity problems. It may also lead to over fitting. In section 5 the value of the KRHS becomes evident: The operations in feature space concerning the generalizations of length and angle can be performed without explicit reference to the feature map. Moreover generalized similarity measures can be constructed. In section 6 a list of kernels is presented. This involves in particular a systematic construction of kernels. More historical references concerning Statistics/Probability Theory are included. The general part continues in subsection 7 with a listing of several clustering methods. It starts with a brief review of classical clustering and also mentions several sophisticated kernel clustering methods. As conclusion remains that the kernel k-means algorithm is still a popular method. Section 8 contains an overview of the main part of the paper. It starts with Duda's original algorithm. It also mentions some of the difficulties remaining typical of hill climbing methods. Unfortunately those are still present in the author's original kernel version. But a solution of these difficulties is mentionend. It contains a careful initialization. In section 9 the kernel version of the main algorithm is given without the more technical details. It is shown how the centres of the clusters change if an element is tentatively moved to another cluster. An easily evaluated criterion is provided for deciding whether this is advantageous as far as the target function is concerned. In section 10 the new initialization is described. It is easily transferred from the original space to the feature space since again the feature map is not needed explicitly. In section 11 the technical

details of the algorithm are presented. In particular the mean (centre) updates in terms of kernels are explained. In section 12 the previous results are collected together. This gives a pseudo code for the main algorithm. Section 13 contains reports on experimental results. The paper finishes with a conclusion in section 14 as is customary.

2. Kernels

Kernels arise within the realm of Statistics quite naturally, see [1]. However, within the aerea of Neural Networks the first systematic treatment seems to have appeared in [2]. In fact in this context the elementary geometrical concepts of length and angle played an important role. In the Cartesian plane or in three dimensions they were quite sufficient to construct a separating plane between (in the simplest case two) classes of vectors. Even similarity measures using a cosine between angles of vectors proved unproblematic.

Definition:
The squared length of a vector $\mathbf{x} = (x_1, x_2)$ denoted by $\|\mathbf{x}\|^2$ is given by

$$\|\mathbf{x}\|^2 = x_1^2 + x_2^2 \tag{1}$$

Suppose that $\mathbf{x}$ and $\mathbf{y}$ are unit vectors and that they make angles α_1 and α_2 respectively with the x-axis then the angle $\alpha = \alpha_1 - \alpha_2$ is given (using elementary trigonometry) by

$$\cos \alpha = \cos \alpha_1 \cos \alpha_2 + \sin \alpha_1 \sin \alpha_2 = x_1 y_1 + x_2 y_2 \tag{2}$$

However, it was soon realized that an algebraic version of these concepts was needed to cope with higher dimensions. Of course, the above definition in equation (2) immediately suggests an algebraic version of the geometric concepts by introducing the scalar product.

Definition:
Let two vectors $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$ be given. Then their *scalar product* denoted by $< ., . >$ is given by

$$< \mathbf{x}, \mathbf{y} > = x_1 y_1 + x_2 y_2 \tag{3}$$

This then generalizes to higher dimensions in the obvious way. Note also that due to the Schwartz inequality the definition of the cosine in higher dimensions is consistent with the definition in two and three dimensions since the Schwartz inequality guarantees that the cosine has modulus ≤ 1.

Utilizing these definitions one can easily construct two primitive similarity measures sim_1 and sim_2 between vectors $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ by setting

$$sim_1(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| \tag{4}$$

and

$$sim_2(\mathbf{x}, \mathbf{y}) = \cos(\alpha) \tag{5}$$

Here $\cos(\alpha) = \frac{1}{\|\mathbf{x}\|\|\mathbf{y}\|} < \mathbf{x}, \mathbf{y} >$.
More similarity measures can be found in [3, 4]. However, even the scalar product between vectors admits a further generalization. This generalization does not even require a vector space structure. The definition is given as follows.

Definition:
Given a topological space X and a continuous function $K : X \times X \rightarrow \mathfrak{R}$, where $\mathfrak{R}$ denotes the real numbers. Then K is called a *positive semi definite* (p.s.d.) kernel if it satisfies a symmetry condition, namely

$$K(x, y) = K(y, x) \quad \forall x, y \in X \times X \tag{6}$$

and a positivity condition

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j K(x_i, x_j) \geq 0 \quad \forall (\alpha_i, x_i) \in \mathfrak{R} \times X. \tag{7}$$

Example: The scalar product between vectors obviously satisfies conditions (6) and (7). This example shows that the given definition extends the scalar product. For more general kernels and in particular the construction of kernels see e.g.[2], p. 291-326.

Note that the elements in the above definition have not been described in bold face to emphasize that they are not necessarily vectors. However, in the sequel only vector spaces shall be considered.

3. The Reproducing Kernel Hilbert Space

This is an abstract construction that is most important for practical applications, see [5, 6, 7, 8]. It guarantees the existence of a map embedding the original sample space into a Hilbert space (sometimes also called feature space). Somewhat unusually the Hilbert space consists of functions where addition and scalar multiplication are defined pointwise. To be more precise:

Definition:
Given a p.s.d. kernel K on a vector space X. Let

$$\{ \mathcal{F} = \sum_{i=1}^{n} \alpha_i K(\mathbf{x_i}, .) : n \in \mathcal{N} \quad \forall (\alpha_i, \mathbf{x_i}) \in \mathfrak{R} \times X \} \tag{8}$$

In (8) define addition and scalar multiplication pointwise (the arguments of the functions have been indicated by "."). Let $f, g \in \mathcal{F}$ be given as $f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i K(\mathbf{x_i}, \mathbf{x})$ and $g(\mathbf{x}) = \sum_{j=1}^{m} \beta_j K(\mathbf{y_j}, \mathbf{x})$. Then define the scalar product by;

Definition:

$$< f, g > = \sum_{i=1}^{l} \sum_{j=1}^{m} \alpha_i \beta_j K(\mathbf{x_i}, \mathbf{y_j}) = \sum_{i=1}^{l} \alpha_i g(\mathbf{x_i}) = \sum_{j=1}^{m} \beta_j f(\mathbf{y_j}) \tag{9}$$

Clearly this scalar product has the required symmetry and bilinearity properties that follow from (9). The positivity condition follows from the p.s.d. kernel. Moreover the heading of the section is explained by observing on taking $g = K(\mathbf{x}, .)$ in (9) that

$$< f, K(\mathbf{x}, .) > = \sum_{i=1}^{l} \alpha_i K(\mathbf{x_i}, \mathbf{x}) = f(\mathbf{x}) \tag{10}$$

By separation and completion a Hilbert Space is obtained as usual. Note that by abuse of notation the scalar product has been denoted by the same symbol in both spaces. This

should not cause any problems since it will be obvious from the context which scalar product is intended.

From a practical point of view it is most important to realize that the existence of a map η into the feature space has been shown, namely

$$\eta(\mathbf{x}) = K(\mathbf{x}, .) \tag{11}$$

4. Embedding Map versus Kernel

It was realized at an early stage that by embedding the original sample space into a higher dimensional space greater flexibility could be achieved, see e.g. [8]. Unfortunately the corresponding embedding map η turned out to be somewhat difficult to handle in practice. This is going to be shown by considering a simple polynomial kernel given as

$$K(\mathbf{x}, \mathbf{y}) = (c+ < \mathbf{x}, \mathbf{y} >)^2 \text{ for a constant } c \geq 0. \tag{12}$$

If the original sample space is Euclidean of dimension n then the embedding function η will map into a space of monomials of degree ≤ 2. Knuth in [9] p. 488 gives the number of different monomials of degree 2 as $\binom{n+1}{2}$. Hence an easy induction proof over n shows that the number of monomials of degree ≤ 2 is given by $\binom{n+2}{2}$. The rather complicated embedding function is in [10] implicitly described via the kernel as;

$$K(\eta(\mathbf{x}), \eta(\mathbf{y})) = \sum_{i=1}^{n} x_i^2 y_i^2 + \\ \sum_{i=2}^{n} \sum_{j=1}^{i-1} \sqrt{2} x_i x_j \sqrt{2} y_i y_j + \sum_{i=1}^{n} \sqrt{2c} x_i \sqrt{2c} y_i + c^2 \tag{13}$$

Whilst the generation of this formula is not at all easy (it involves the multinomial theorem) it is quite simple to verify it by induction over n. Using (13) the explicit embedding function in [10] is given as;

$$\eta(\mathbf{x}) = (x_n^2, \dots, x_1^2, \sqrt{2} x_n x_{n-1} \\ \dots, \sqrt{2} x_n x_1 \dots, \sqrt{2c} x_n, \dots, \sqrt{2c} x_1, c) \tag{14}$$

Clearly it would be most cumbersome to deal with the explicit embedding function even in this simple example. Fortunately enough it is possible to avoid explicitly handling the feature map since all the information required is present in the kernel.

5. Length and Angle in Feature Space

From (11) it follows immediately that the length of an element in feature space is given by

$$\eta(\mathbf{x}) = K(\mathbf{x}, \mathbf{x})^{1/2} \tag{15}$$

Hence the (generalized) length of a vector in feature space can be computed without explicitly using the embedding map. This also holds for more general vectors in feature space as can be seen by using the bilinearity property

of the scalar product in feature space. The distance between vectors in feature space is similarly worked out:

$$\|\eta(\mathbf{x}) - \eta(\mathbf{y})\|^2 = K(\mathbf{x}, \mathbf{x}) - 2K(\mathbf{x}, \mathbf{x}) + K(\mathbf{y}, \mathbf{y}) \tag{16}$$

Again the feature map is not needed explicitly. It seems somewhat remarkable, that generalized similarity measures can be constructed by using the generalized notion of angle;

$$cos(\eta(\mathbf{x}), \eta(\mathbf{y})) = (\frac{1}{K(\mathbf{x}, \mathbf{x})K(\mathbf{y}, \mathbf{y})})^{(1/2)} K(\mathbf{x}, \mathbf{y}) \tag{17}$$

That the generalized cosine has modulus ≤ 1 is guaranteed by the Schwartz inequality. A further concept is needed for later use. Suppose now that the topological space X is a finite set $S := \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}\}$ of samples in a Euclidean space (this is the situation envisaged for the kernel k-means iterative algorithm below) and let $FS := \{\eta(\mathbf{x_1}), \eta(\mathbf{x_2}), ..., \eta(\mathbf{x_n})\}$ denote its image in feature space. Then its centre of mass or mean can be defined in feature space by;

Definition:

$$mean(FS) := 1/n \sum_{i=1}^{n} \eta(\mathbf{x_i}) \tag{18}$$

Note here that the mean my not have a preimage in X. Nevertheless the distance of a point $\mathbf{w} = \eta(\mathbf{x})$ in FS from the center of mass can be computed:

$$\|\mathbf{w} - mean(FS)\|^2 = < \eta(\mathbf{x}), \eta(\mathbf{x}) > -\frac{2}{n} < \eta(\mathbf{x}), \sum_{i=1}^{n} \eta(\mathbf{x_i}) > \\ +\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} < \eta(\mathbf{x_i}), \eta(\mathbf{x_j}) > = \\ K(\mathbf{x}, \mathbf{x}) - \frac{2}{n} \sum_{i=1}^{n} K(\mathbf{x_i}, \mathbf{x}) + \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} K(\mathbf{x_i}, \mathbf{x_j}) \tag{19}$$

6. List of Kernels

The history of kernels goes back a considerable time. As mentioned above they seem to have arisen mainly in the context of Probability Theory and Statistics. Two of the earliest papers seem to be due to Schoenberg [11, 12]. He considers kernels in the context of positive definite functions, where normalized positive definite functions can be seen as Fourier transforms of probability measures by Bochner's theorem. In the same context conditionally positive definite functions (they appear as logarithms of positiv definite functions) were treated in [5]. There was also exhibited a connection to the Levy-Khinchine formula.

A systematic construction of kernels is described in [2], as mentioned above. The particular choice of kernels suitable to solve a special problem can vary quite considerably.

Particularly popular are polynomial kernels of degree two or three, see [11]. Higher dimensional kernels are not frequently used since there is a danger of overfitting, see [13]. Further kernels may be found in [14] and [15] where additionally the connection to certain cohomology groups is treated.

7. Clustering Kernel Algorithms

Clustering has been a subject of study for a long time, see [13, 16, 17]. However, kernel clustering seems to be somewhat newer. One of the first systematic studies can probably be found in [2], p. 264-280. There among others measuring cluster quality, a k-means algorithm, spectral methods, clustering into two classes, multiclass clustering and the eigenvector approach are discussed.

More recently [8, 18] and [19] must be mentioned. In the latter as an example kernel PCA is treated. Of course, principal component analysis plays an important role where image recognition is concerned. Nevertheless the k-means algorithm in its various forms remains popular presumably because of its simplicity.

8. Overview of the Main Part of the paper

In the seminal book, the authors among other topics treat unsupervised learning and clustering in particular. They establish an elegant iterative version of the k-means algorithm [17], p.548. In order to increase its flexibility and efficiency a kernel version of this algorithm was given in [20], p.221; for related work see [1, 21], and for the well-known connection to maximum likelihood methods see also [11, 21]. Unfortunately one of the major shortcomings already pointed out by [17] that are typical of hill climbing algorithms, namely getting stuck in local extrema, remained. However, in [6] an initialization for general k-means algorithms was presented that proved applicable. Moreover it was shown in several tests that this improved the performance considerably, see [2]. Hence the slightly modified algorithm including this initialization is transferred to feature space here. It allows an easy quantification of the further improvements after the initialization. This was also employed to create a trial and error version using multiple repetitions and a ratchet, as in Gallant's Pocket Algorithm, see [22]. This was suggested in the conclusion of [2]. Also, en passant, several formal results present in [2] and [20] were modified allowing for better readability.

9. The Main Algorithm

The problem that is originally being considered may be formulated as follows:

Given a set of n samples $S := \{x_1, x_2, ..., x_n\}$, then these samples are to be partitioned into exactly k sets $S_1, S_2, ..., S_k$. Each cluster is to contain samples more similar to each other than they are to samples in other clusters. To this end one defines a target function that measures the clustering quality of any partition of the data. The problem then is to find a partition of the samples that optimizes the target function. Note here that the set S from now on will carry a vector space structure in the present paper. Note also that the number of sets for the partition will now be denoted by k since a k-means algorithm will be employed.

In contrast to Duda's original definition the target function will be defined as the sum of squared errors in feature space. More precisely, let n_i be the number of samples in S_i and let

$$\mathbf{c_i} = 1/n_i \sum_{\mathbf{x} \in S_i} \eta(\mathbf{x}) \tag{20}$$

be their mean in feature space, then the sum of squared errors is defined by

$$E_k := \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\eta(\mathbf{x}) - \mathbf{c_i}\|^2 \tag{21}$$

Of course, the above expressions, (20), (21)) can be expressed without explicitly using the feature map as shown in (19).Thus for a given cluster S_i the mean vector $\mathbf{c_i}$ is the best representative of the samples in S_i in the sense that it minimizes the squared lengths of the *error vectors* $\eta(\mathbf{x}) - \mathbf{c_i}$ in feature space. The target function can now be optimized by iterative improvement setting;

$$E_k := \sum_{i=1}^{k} E_i \tag{22}$$

where the squared error per cluster is defined by

$$E_i := \sum_{\mathbf{x} \in S_i} \|\eta(\mathbf{x}) - \mathbf{c_i}\|^2 \tag{23}$$

Suppose that sample $\mathbf{x_t}$ in S_i is tentatively moved to S_j then $\mathbf{c_j}$ changes to

$$\mathbf{c_j^\star} := \mathbf{c_j} + 1/(n_j + 1)(\eta(\mathbf{x_t}) - \mathbf{c_j}) \tag{24}$$

and E_j increases to

$$E_j^\star = E_j + n_j/(n_j + 1)\|\eta(\mathbf{x_t}) - \mathbf{c_j}\|^2 \tag{25}$$

For details see [20], p. 221, [23].

Similarly, under the assumption that $n_i \neq 1$, (singleton clusters should not be removed) $\mathbf{c_i}$ changes to

$$\mathbf{c_i^\star} := \mathbf{c_i} - 1/(n_i - 1)(\eta(\mathbf{x_t}) - \mathbf{c_i}) \tag{26}$$

and E_i decreases to

$$E_i^\star = E_i - n_i/(n_i - 1)\|\eta(\mathbf{x_t}) - \mathbf{c_i}\|^2 \tag{27}$$

These formulae simplify the computation of the change in the target function considerably. Thus it becomes obvious that a transfer of $\mathbf{x_t}$ from S_i to S_j is advantageous if the decrease in E_i is greater than the increase in E_j. This is the case if

$$n_i/(n_i - 1)\|\eta(\mathbf{x_t}) - \mathbf{c_i}\|^2 > n_j/(n_j + 1)\|\eta(\mathbf{x_t}) - \mathbf{c_j}\|^2 \tag{28}$$

Thus, if reassignment is advantageous then the greatest decrease in the target function is obtained by selecting the cluster for which

$$n_j/(n_j + 1)\|\eta(\mathbf{x_t}) - \mathbf{c_j}\|^2 \tag{29}$$

is minimal. It seems worth pointing out again that due to (19) the embedding map can be eliminated and thus no explicit reference to the feature space must be made.

10. Initialization

In [24], the authors prove an interesting method for initializing the classical k-means algorithm, see also [25]. They cite several tests to show the advantages of their careful seeding. In view of the distance function in feature space described without explicitly using the feature map in (15) this can easily be applied for the algorithm described here. For use below a lemma proves helpful that follows from lemma 2.1 in [26], see also [6], but has been transferred into feature space.

Leema: Let FS be as above, i.e.

$$FS := \{\eta(\mathbf{x_1}), \eta(\mathbf{x_2}), ..., \eta(\mathbf{x_n})\} \tag{30}$$

with center $mean(FS)$ and let $\mathbf{z}$ be an arbitrary point with $\mathbf{z} \in FS$. Then

$$\sum_{\mathbf{x} \in FS} \|\eta(\mathbf{x}) - \mathbf{z}\|^2 - \sum_{\mathbf{x} \in FS} \|\eta(\mathbf{x}) - \text{mean}(FS)\|^2 = \\ n * \|mean(FS) - \mathbf{z}\|^2 \tag{31}$$

Indeed the procedure in feature space may then be described as follows.
1. Choose an initial center $\mathbf{c_1}$ uniformly at random from FS.
2. Select the next center $\mathbf{c_i}$ from FS with probability

$$(D(\mathbf{c_i}))^2 / \sum_{\mathbf{c} \in FS} (D(\mathbf{c}))^2 \tag{32}$$

Here $D(\mathbf{c})$ denotes the shortest distance from the data point $\mathbf{c}$ to the closest center that has already been chosen.

In [2], the authors proves that the expected value of the target function is of order $8(ln k + 2)E_k(optimal)$ after the described initialization. However, seeing that only improvements can occur in the iterative algorithm this is somewhat satisfactory.

11. Technical Details of the Algorithm

Without explicitly using the kernel feature space function the increase in E_j can now be expressed as

$$n_j/(n_j + 1)[K(\mathbf{x_t}, \mathbf{x_t}) + 1/n^2 \sum_{\mathbf{x} \in S_j} \sum_{\mathbf{y} \in S_j} K(\mathbf{x}, \mathbf{y}) \\ -2/n \sum_{\mathbf{y} \in S_j} K(\mathbf{x_t}, \mathbf{y})] \tag{33}$$

The decrease in E_i can be obtained in a completely analogous fashion, as mentioned before. Thus, if reassignment is possible, then the cluster that minimizes the above expression should be selected.

Mean Updates in Terms of Kernels: It is useful to define an $n \times k$ indicator matrix $\mathbf{S}$ as follows, see [2]:

$$\mathbf{S} = \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1k} \\ s_{21} & s_{22} & \cdots & s_{2k} \\ \cdots\cdots\cdots\cdots\cdots \\ s_{n1} & s_{n2} & \cdots & s_{nk} \end{pmatrix} \tag{34}$$

Here $\begin{cases} s_{ij} = 1 \ if \ \mathbf{x_i} \in S_j \\ s_{ij} = 0 \ otherwise. \end{cases}$

Clearly the matrix $\mathbf{S}$ has precisely one 1 in every row whilst the column sums describe the number of samples in every cluster. Moreover a $k \times k$ diagonal matrix $\mathbf{D}$ is needed.

$$\mathbf{D} = \begin{pmatrix} 1/n_1 & 0 & \cdots & 0 \\ 0 & 1/n_2 & \cdots & 0 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ 0 & 0 & \cdots & 1/n_k \end{pmatrix} \tag{35}$$

The entries on the diagonal of $\mathbf{D}$ are just the inverses of the number of elements in each cluster (notation as above). In addition a vector $\mathbf{X}$ containing the feature version of the training examples will be helpful.

$$\mathbf{X} = \begin{pmatrix} \eta(\mathbf{x_1}) \\ \eta(\mathbf{x_2}) \\ \cdots\cdots \\ \eta(\mathbf{x_n}) \end{pmatrix} \tag{36}$$

From this one obtains on a purely formal level

$$\mathbf{X}^T\mathbf{SD} = (\sum_{\mathbf{x} \in S_1} \eta(\mathbf{x}), \sum_{\mathbf{x} \in S_2} \eta(\mathbf{x}), \dots, \sum_{\mathbf{x} \in S_k} \eta(\mathbf{x}))\mathbf{D} \tag{37}$$

which gives

$$(1/n_1 \sum_{\mathbf{x} \in S_1} \eta(\mathbf{x}), 1/n_2 \sum_{\mathbf{x} \in S_2} \eta(\mathbf{x}), \dots, 1/n_k \sum_{\mathbf{x} \in S_k} \eta(\mathbf{x}))$$

This does of course describe the vector of means in feature space albeit still containing the feature map explicitly. Thus, to construct the complete algorithm, it is still necessary to remove the dependence on the feature map.

Finally a vector $\mathbf{k}$ of scalar products between $\eta(\mathbf{x_t})$ and the feature version of the samples is going to be defined in terms of the kernel matrix K:

$$\mathbf{k} = \begin{pmatrix} K(\mathbf{x_t}, \mathbf{x_1}) \\ K(\mathbf{x_t}, \mathbf{x_2}) \\ \cdots\cdots\cdots \\ K(\mathbf{x_t}, \mathbf{x_n}) \end{pmatrix} \tag{38}$$

Hence $\mathbf{k}^T\mathbf{SD}$ is given by

$$(< \eta(\mathbf{x_t}), 1/n_1 \sum_{\mathbf{x} \in S_1} \eta(\mathbf{x}) >, < \eta(\mathbf{x_t}), 1/n_2 \sum_{\mathbf{x} \in S_2} \eta(\mathbf{x}) >, \dots,)$$

It is now possible to compute

$$n_j/(n_j + 1)\|\eta(\mathbf{x_t}) - \mathbf{c_j}\|^2 \\ = n_j/(n_j + 1)(\|\eta(\mathbf{x_t})\|^2 - 2 < \eta(\mathbf{x_t}), \mathbf{c_j} > +\|\mathbf{c_j}\|^2) \tag{39}$$

without involving the explicit use of the feature map whilst also including the indicator matrix:

$$n_j/(n_j + 1)(K(\mathbf{x_t}, \mathbf{x_t}) - 2(\mathbf{k}^T\mathbf{SD})_j + (\mathbf{DS}^T K(\mathbf{x_i}, \mathbf{x_j})\mathbf{SD})_{jj})$$

Note here that for brevity the j-th vector (jj-th matrix) elements have been indicated by subscripts.

12. Pseudo Code for the Main Algorithm

The Kernel Algorithm
Collecting together the above results the following kernel k-means algoritm is obtained:

begin initialize n, k, $c_1(p), c_2(p), \ldots, c_k(p)$
as described above, $E(p), S(p), D(p),$ iter.
iter times repeat
($\star$)
initialize $c_1(t), c_2(t), \ldots, c_k(t),$
$E(t), S(t), D(t).$
do randomly select a sample $x_t(t)$
    $i \leftarrow \text{argmin}_{1 \le l \le k} (K(x_t(t), x_t(t)) - 2(k^T(t)S(t)D(t))_l +$
    $(D(t)S^T(t) K(x_i, x_j)S(t)D(t))_{ll}))$ (classify x_t)
    if $n_i \ne 1$ **then** compute

$$\rho_j = \begin{cases} n_j/(n_j + 1)(K(x_t(t), x_t(t)) - 2(k^T(t)S(t)D(t))_j \\ \quad + (D(t)S^T(t)K(x_i, x_j)S(t)D(t))_{jj}) \ j \ne i \\[2ex] n_i/(n_i - 1)(K(x_t(t), x_t(t)) - 2(k^T(t)S(t)D(t))_i \\ \quad + (D(t)S^T(t)K(x_i, x_j)S(t)D(t))_{ii}) \ j = i \end{cases}$$

    if $\rho_m \le \rho_j$ for all j **then** transfer $x_t(t)$ to S_m
    recompute $E(t), c_i(t), c_m(t)$ and update the n_i in $D(t)$
    as well as the entries of $S(t)$
    until no change in $E(t)$ in n attempts
    if iter $\ne 0$ **then** iter = iter-1 **go to**($\star$) **else**
    if $E(t) \le E(p)$ **then** replace the
    pocket values by the temporary values
    return the pocket values
end

Here the t designates temporary values. The temporary values are finally transferred to the pocket values and thus constitute the output.
In addition the expression behind the brace describes the updates of the centres (means) as obtained in section 11. Using Gallant's method it is possible to fix a number of repetitions of the algorithm and then select the best one obtained. This is standard practice.

13. Experimental Results Reported

In [1] David Arthur and Sergei Vassilitski report the following.

Experiments:
They implemented and tested their k-means++ algorithm in C++ and compared it to k-means. They tested k = 10, 25, 50 with 20 runs each due to randomized seeding processes.

Data Sets Used:
They used four datasets. The first one was a synthetic data set containing 25 centers selected at random from a fifteen dimensional hypercube. They then added points of a Gaussian distribution of variance 1 around each center thus obtaining a good approximation to the optimal clustering around the original centers. The remaining datasets were chosen from real-world examples off the UC-Irvine Machine Learning Repository.

Results Reported:
They observed that k-means++ consistently outperformed k-means, both by achieving a lower target function value,

in some cases by several orders of magnitude, and also by having a faster running time. The D^2 seeding (the weighting given in (32)) was slightly slower than uniform seeding, but it still lead to a faster algorithm since it helped the local search converge after fewer iterations. The synthetic example was a case where standard k-means did very badly. Even though there was an "obvious" clustering, the uniform seeding would inevitably merge some of these clusters, and the local search would never be able to split them apart. The careful seeding method of k-means++ avoided this problem altogether, and it almost always attained the optimal clustering on the synthetic dataset. As far as applications go it should be mentioned that it is possible to exploit this algorithm for maximum likelihood applications, for details see [21], where the clustering algorithm is used to obtain an approximate solution of the maximum likelihood problem for normal mixtures.

The difference between k-means and k-means++ on the real world data sets was also substantial. Without exception k-means++ achieved a significant improvement over k-means. In every case, k-means++ achieved at least a 10 % improvement in accuracy over k-means, and it often performed much better. Indeed, on the Spam and Intrusion datasets, k-means++ achieved target function values 20 to 1000 times smaller than those achieved by standard k-means. Each trial also completed two to three times faster, and each individual trial was much more likely to achieve a good clustering Of course, these results cannot be directly transferred to the kernel algorithm, since that is dependent on the particular kernel employed. The choice of kernel again depends on the particular problem to be handled. Nevertheless, they provide good indications as to how to improve the classical algorithm in feature space.

14. Conclusion

By adopting the initialization suggested by Arthur and Vassilitski the kernel version of Duda's algorithm could be improved. In particular it seems that the well-known problems concerning getting stuck in local extrema have been avoided to some extent. This way the special advantages of the iterative algorithm originally described by Duda can be fully exploited. Indeed experimental results presented by Arthur and Vassilitski substantiate that claim. Moreover a trivial modification allows a quantification of improvements obtained after the initialization. Thus an easy application of trial and error methods using a ratchet is obtained. This is similar to a method used in supervised learning (Gallant's Pocket Algorithm). As far as applications go it should be pointed out that it is possible to exploit this algorithm for maximum likelihood applications, where the clustering algorithm is used to obtain an approximate solution of the maximum likelihood problem for normal mixtures. In this context it should also be mentioned that by Arthur and Vassilitski certain generalizations of the k-means++ algorithm are considered yielding only slight weaker results. The author also considered Bregman Divergencies and the so-called Jensen Bregman. This can be utilized to obtain a generalization by appealing to the Reproducing Kernel Hilbert Space, see section 3.

The experimental results described above give clear indications towards the advantages of careful seeding. However,

further investigations are needed to find suitable kernels for the particular problems considered. Thus there is still room for much future work.

References

[1] B.-J. Falkowski, "Maximum likelihood estimates and a kernel k-means iterative algorithm for normal mixtures", "USB Proceedings IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society", Online Singapore, doi:10.1109/IECON43393.2020.9254276.

[2] J. Shawe-Taylor, N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.

[3] B.-J. Falkowski, "On certain generalizations of inner product similarity measures", *Journal of the American Society for Information Science*, vol. 49, no. 9, 1998.

[4] G. Salton, *Automatic Text Processing*, Addison-Wesley Series in Computer Science, 1989.

[5] N. Aronszajn, "Theory of reproducing kernels", *AMS*, 1950.

[6] G. Wahba, "Support vector machines, reproducing kernel hilbert spaces, and randomized gacv", "Advances in Kernel Methods, Support Vector Learning", MIT Press, 1999.

[7] G. Wahba, "An introduction to reproducing kernel hilbert spaces and why they are so useful", "IFAC Publications", Rotterdam, 2003.

[8] A. Gretton, "Introduction to rkhs and some simple kernel algorithms", 2019, uCL.

[9] D. Knuth, *The Art of Computer Programming. Vol. 1, Fundamental Algorithms, 2nd Edition*, 1973.

[10] "Polynomial kernel", https://en.wikipedia.org/wiki/Polynomial_kernel, accessed: 2024-03-24.

[11] I. Schoenberg, "On certain metric spaces arising from euclidean spaces and their embedding in hilbert space", *Annals of Mathematics*, vol. 38, no. 4, 1937.

[12] I. Schoenberg, "Metric spaces and positive definite functions", *Transactions of the American Mathematical Society*, vol. 41, 1938.

[13] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 2013, reprinted.

[14] B.-J. Falkowski, "Mercer kernels and 1-cohomology", N. Baba, R. Howlett, L. Jain, eds., "Proc. of the 5th Intl. Conference on Knowledge Based Intelligent Engineering Systems and Allied Technologies (KES 2001)", IOS Press, 2001.

[15] B.-J. Falkowski, "Mercer kernels and 1-cohomology of certain semisimple lie groups", V. Palade, R. Howlett, L. Jain, eds., "Proc. of the 7th Intl. Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2003)", vol. 2773 of *LNAI*, Springer Verlag, 2003.

[16] A. Jain, "Data clustering: 50 years beyond k-means", *Pattern Recognition Letters*, 2009, doi:10.1016/j.patrec.2009.09.011.

[17] R. Duda, P. Hart, D. Stork, *Pattern Classification*, Wiley, 2017, reprinted.

[18] R. Chitta, "Kernel-clustering based on big data", Ph.D. thesis, MSU, 2015.

[19] "Kernel clustering", http://www.cse.msu.edu>cse902>ppt>Ker.., last visited: 20.01.2019.

[20] B.-J. Falkowski, "A kernel iterative k-means algorithm", "Advances in Intelligent Systems and Computing 1051, Proceedings of ISAT 2019", Springer-Verlag, 2019.

[21] B.-J. Falkowski, "Bregman divergencies, triangle inequality, and maximum likelihood estimates for normal mixtures", "Proceedings of the 2022 Intelligent Systems Conference (Intellisys)", vol. 1, Springer, 2022, doi:10.1007/978-3-031-16072-1_12.

[22] S. Gallant, "Perceptron-based learning algorithms", *IEEE Transactions on Neural Networks*, vol. 1, no. 2, 1990.

[23] D. Stork, "Solution manual to accompany pattern classification, 2nd ed.", PDF can be obtained via Wiley.

[24] D. Arthur, S. Vassilitski, "k-means++: The advantages of careful seeding", "Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)", 2007.

[25] S. Har-Peled, B. Sadri, "How fast is the k-means method?", "ACM-SIAM Symposium on Discrete Mathematics (SODA)", 2005.

[26] "Kernel k-means and spectral clustering", https://www.cs.utexas.edu/~inderjit/public_papers/kdd_spectral_kernelkmeans.pdf, last visited: 19.06.2018.

BERND-JÜRGEN FALKOWSKI obtained his Ph. D. in Mathematics from the Victoria University of Manchester (England). He was employed by the "Hochschule Stralsund" as a "Professor fuer Wirtschaftsinformatik" until 2010. Thereafter he lectured at the "Fachhochschule für Oekonomie und Management" in Augsburg and Muenchen.

His research interests include Artificial Intelligence, Statistics and Applications.

Evaluation of Equivalent Aacceleration Factors of Repairable Systems in a Fleet: a Process-Average-Based Approach

Renyan Jiang [1, 2], Kunpeng Zhang [1, 2], Xia Xu [2, 3], Yu Cao [1, 2*]

[1] China International Science & Technology Cooperation Base for Laser Processing Robotics, Wenzhou University, Wenzhou 325035, China.
[2] Zhejiang Provincial Innovation Center of Laser Intelligent Equipment Technology, Wenzhou, 325000, China.
[3] Penta laser (Zhejiang) Co., Ltd., Wenzhou 325000, China.
Corresponding Author: Yu Cao, Wenzhou University, E-mail: yucao@wzu.edu.cn

ABSTRACT: Research on repairable systems in a fleet is mainly concerned with modelling of the failure times using point processes. One important issue is to quantitatively evaluate the heterogeneity among systems, which is usually analyzed using frailty models. Recently, a fleet heterogeneity evaluation method is proposed in the literature. This method describes the heterogeneity with the relative dispersion of equivalent acceleration factors (EAFs) of systems, which is defined as the ratio of the mean times between failures (MTBFs) of a system and a reference system. A main drawback of this method is that the MTBFs of a specific system and the reference system are estimated at different times while the MTBF estimated at different time can be different. This paper aims to address this issue by proposing an improved method. The proposed method uses an "average process" as the reference process and estimates the MTBFs of systems and the reference system at a common time point. This leads to more robust MTBF estimates. Three datasets are analyzed to illustrate the proposed method and its superiority.

KEYWORDS: Repairable system; reliability modeling; MTBF; equivalent acceleration factor; fleet heterogeneity

1.   Introduction

The literature on repairable systems is vast and mainly concerned with modelling of the failure times using point processes [1-2]. The repairable systems that most of the literature deals with can be roughly divided into two categories: (a) multi-component repairable systems composed by non-identical components, and (b) repairable systems in a fleet composed by identical or similar units. In this paper, we focus on the repairable systems in a fleet.

The reliability modeling of repairable systems in a fleet involves a number of issues, including:

- Failure trend analysis [3-4] and failure pattern analysis [5-6]. The former deals with the behavior of the failure occurrence rate or failure intensity, e.g., increasing, decreasing or non-monotonic over time; and the latter aims to identify typical shape types of failure intensity function.

- Fleet heterogeneity evaluation and/or modeling [7-10]. Basic models are frailty models [11-13] and the equivalent acceleration factor model can be viewed as an exception [14].

- Reliability model selection [15-16] and development of models and/or methods for modeling the data with different types of information or covariates (e.g., failure causes and modes) and different types of censoring [17-26].

- Maintenance quality evaluation [27], maintenance decision optimization [14, 23, 28], and prediction of number of failures [29-30].

The focus of this paper is on the evaluation of heterogeneity of repairable systems in a fleet.

Different from the frailty models, Jiang et al. [14] recently propose a heterogeneity evaluation method for repairable systems in a fleet. It estimates the mean time between failures (MTBFs) of individual systems at the right end point of the observation window of each system

and defines the equivalent acceleration factor (EAF) of a specific system as the ratio between the MTBFs of this system and a reference system, which is the one whose failure number is the largest among all the systems. The fleet heterogeneity is measured by the ratio of the sample average and standard deviation of EAFs of all the systems. The smaller this ratio is, the greater the heterogeneity of the fleet is. Generally, the observation windows of systems are different so that the MTBFs of different systems are estimated at different time points while the MTBF estimated at different time is different and can be considerably overestimated when the failure number is small. As a result, the EAFs estimated from the above-mentioned heterogeneity evaluation method can be unreliable. This paper aims to address this research gap through proposing an improved method to estimate the EAFs and MTBFs of systems. The proposed method uses an "average system" as the reference system or process, whose mean cumulative function (MCF) equals to the MCF of the fleet. The EAF of a system is estimated as the ratio between the MCFs of the system and fleet (or reference system) at the right end point of the observation window of the system; and the MTBF of the system is estimated as the product of the EAF and fleet MTBF. A unique advantage of using the average process as the reference process is that any data point on the MCF of a specific unit always corresponds to a data point on the MCF of the average process. The reference process defined in [14] does not have such a property. This property makes it possible to assess the MTBFs at the same time. Three datasets are analyzed to illustrate the proposed method and its superiority.

The paper is organized as follows. The proposed method is presented in Section 2, compared with the frailty model in Section 3, and illustrated in Section 4. The paper is concluded in Section 5.

2. Proposed method

2.1. Nelson MCF estimator

Consider a fleet of nominally identical repairable systems, and each system is called a unit. Failure point processes of units are randomly censored on the right. Let t_{ij} denote the time to the jth failure of the ith unit, and $x_{ij} = t_{ij} - t_{i,j-1}$ ($t_{i,0} = 0$) denote the inter-failure time. The failure point process data are given by

$$\{t_{ij} \leq \tau_i, 1 \leq i \leq n, 1 \leq j \leq n_i\} \quad (1)$$

where τ_i is the censored time of the ith unit, and n_i is the failure number of the ith unit. The fleet is single censoring if $\tau_1 = \tau_2 = \cdots = \tau_n$; otherwise, it is multiple censoring. The total failure number of the fleet is $N = \sum_{i=1}^{n} n_i$, and the censoring time of the fleet is defined as

$$\tau = \max_{i}(\tau_i). \quad (2)$$

The total time on test (TTT, or T_3 for short) of the fleet at τ is $T_3(\tau) = \sum_{i=1}^{n} \tau_i$, and the fleet MTBF evaluated at τ is estimated as

$$\mu_F = T_3(\tau)/N. \quad (3)$$

A failure point process can be characterized by the MCF of the fleet, which is the mean cumulative number of failures per unit. The MCF can be used to predict the total number or cost of repairs of the fleet in a future period and to determine an optimum retirement age of a system [7]. To estimate the MCF of the fleet, the data given by (1) are sorted in an ascending order. Let t_k ($1 \leq k \leq N$) denote the kth smallest value of t_{ij}'s, and $s(t_k)$ denote the number of units at risk at time t_k. For a single censoring fleet, $s(t_k) = n$; for a multiple censoring fleet, $s(t_k)$ decreases with k and tends to 1. The Nelson MCF estimator is given by [7-8]

$$M_N(t_k) = M_N(t_{k-1}) + \frac{1}{s(t_k)}, \quad t_0 = 0, M_N(0) = 0. \quad (4)$$

Equation (4) is a staircase function with $M_N(t_k^-) = M_N(t_{k-1}^+)$. That is, the MCF at t_k has two values. To make the MCF at t_k unique, a smoothed MCF is defined as

$$M(t_k) = \frac{M_N(t_{k-1}) + M_N(t_k)}{2}. \quad (5)$$

An MCF-based fleet MTBF can be estimated as

$$\mu_{FN} = \frac{\tau}{M(\tau)}. \quad (6)$$

Thus, we have two the MTBF estimates obtained from (3) and (6), respectively. For a single censoring fleet, we have $T_3(\tau) = n\tau$ and $M(\tau) = N/n$. From (6) yields

$$\mu_{FN} = \frac{\frac{T_3(\tau)}{n}}{\frac{N}{n}} = \frac{T_3(\tau)}{N} = \mu_F. \quad (7)$$

In this case, (6) is consistent with (3). For a multiple censoring fleet, the MTBF estimated from (6) is generally different from the MTBF estimated from (3). As mentioned by Nelson [7], the Nelson MCF estimator for large t is not robust for a multiple censoring fleet since $s(t)$ decreases as t increases. Therefore, the fleet MTBF estimated by (3) is preferred.

2.2. Robust MCF estimator

To improve the Nelson MCF estimator, a robust estimator is recently proposed by [14]. Specific details are outlined as follows.

Suppose that an ordered dataset ($t_k, 1 \leq k \leq N$) is collected from a multiple censoring fleet with n units. An equivalent operating time t_k^* associated with t_k is defined by

$$t_k^* = \frac{T_3(t_k)}{n}. \quad (8)$$

Define an equivalent single censoring fleet with n units, whose censoring time is defined as

$$\tau^* = \frac{T_3(\tau)}{n}. \quad (9)$$

Its MCF is estimated by (4) and (5) with t_k and $s(t_k)$ being replaced by t_k^* and n, respectively. The MCF obtained in such a way is called as the robust MCF estimator. Its robustness results from the fact that the increment of the MCF is $1/n$, which is smaller than the MCF increment of the Nelson estimator, $1/s(t_k)$.

Based on the robust MCF, the fleet MTBF can be estimated as

$$\mu_{FJ} = \frac{\tau^*}{M(\tau^*)}. \tag{10}$$

It is noted that $M(\tau^*) = N/n$. Using this relation and (9) to (10) yields

$$\mu_{FJ} = \frac{T_3(\tau)}{N} = \mu_F. \tag{11}$$

That is, (11) is consistent with (3) and hence the appropriateness of the robust MCF estimator is confirmed.

2.3. Equivalent acceleration factors

[14] estimate the MTBF of the ith unit as

$$\mu_i = \frac{\tau_i}{n_i}. \tag{12}$$

The unit that has the largest failure number is selected as the reference unit, whose MTBF and censoring time are denoted as μ_R and τ_R, respectively. The EAF of the ith unit is defined as

$$\alpha_i = \frac{\mu_i}{\mu_R}. \tag{13}$$

Clearly, α_i can be viewed as a normalized MTBF, which is dimensionless.

Let μ_α and σ_α denote the mean and standard deviation of all the EAF values, respectively. The fleet heterogeneity is measured by

$$\rho_\alpha = \frac{\mu_\alpha}{\sigma_\alpha}. \tag{14}$$

[14] suggest 1.96 as the critical value of ρ_α. If $\rho_\alpha > 1.96$, the fleet is thought to be homogeneous; otherwise, heterogeneous.

Generally, the MTBF estimated at different time is different and it is negatively correlated with unit age. It is noted that μ_i is estimated at $\tau_i \leq \tau$ and μ_R is estimated at τ_R, which is generally different from τ_i. That is, μ_i and μ_R is estimated at different time points. When τ_i and τ_R are significantly different, the EAF estimated by (13) is unreliable. To improve, μ_i and μ_R should be estimated at the same time point τ_i. In this case, (13) is revised as

$$\alpha_i = \frac{\mu_i}{\mu_R(\tau_i)} \tag{15}$$

where $\mu_R(\tau_i)$ is the MTBF of the reference unit estimated at τ_i by interpolation or extrapolation. To facilitate the evaluation of $\mu_R(\tau_i)$ and obtain a more robust estimate of $\mu_R(\tau_i)$, it is necessary to redefine the reference unit. This will yield an improved EAF evaluation method.

2.4. Improved EAF evaluation method

We define a virtual unit as the reference unit, its failure process equals to the "average process" of the fleet, and its MCF is given by the Nelson estimator of the fleet MCF. In this case, we have

$$\mu_R(\tau_i) = \frac{\tau_i}{M(\tau_i)}, \mu_i = \frac{\tau_i}{M_i(\tau_i)} \tag{16}$$

where $M_i(.)$ is the MCF of the ith unit. If $\tau_i > t_{n_i}$, $M_i(\tau_i) = n_i$; if $\tau_i = t_{n_i}$, $M_i(\tau_i) = n_i - 0.5$. Using (16) to (15) yields

$$\alpha_i = \frac{M(\tau_i)}{M_i(\tau_i)}. \tag{17}$$

Assume that the EAF estimated from (17) is approximately a constant. That is, $\alpha_i(\tau_i) \approx \alpha_i(\tau)$. Under this assumption, the MTBFs of the units evaluated at τ can be estimated as

$$\mu_i = \alpha_i \mu_F \tag{18}$$

where μ_F is given by (3). Thus, *the MTBFs of all the units are evaluated at a common time τ.*

It is noted that (18) is the MTBF of the ith unit evaluated at τ while (12) is the MTBF of the ith unit evaluated at τ_i. The two estimates are usually different and their relative error is given by

$$\varepsilon_i = |1 - \frac{\mu_i(\tau_i)}{\mu_i(\tau)}|. \tag{19}$$

When τ_i is much smaller than τ, ε_i may be large, as to be illustrated by Figures 1 and 3 in Section 4.

3. Comparison of the EAF-based approach with frailty model

Traditionally, the heterogeneity of repairable systems in a fleet is modeled by a frailty model [11-13, 31]. In this section, we compare the EAF-based approach with the frailty model and illustrate the conclusions obtained from comparison analysis using a real-world example.

3.1. Frailty model

Consider the failure point processes given by (1). In the context of frailty analysis, each system in the fleet has its own failure intensity; and the effect of the heterogeneity on the failure process of a system is described a non-negative random variable Z called the frailty. The frailty model is defined in terms of the intensity function of a failure process, given by

$$m(t|Z) = Zm_0(t), \tag{20}$$

where $m_0(t)$ is the baseline failure intensity function. Equation (20) is actually a proportional intensity model [32]. According to (20), the intensity function of the ith unit can be written as

$$m_i(t) = z_i m_0(t) \tag{21}$$

where z_i is the frailty of the ith unit.

To specify a frailty model, one needs to specify the type of the repair process (i.e., minimal, imperfect or perfect repair) and the distribution of the frailty. For a minimal repair process, the non-homogeneous Poisson process with a power-law MCF is assumed. The power-law MCF is given by

$$M(t) = (t/\eta)^{\beta}. \tag{22}$$

The frailty is usually assumed to follow the gamma distribution with scale parameter θ and shape parameter $1/\theta$. The expectation of this distribution is one and the variance is θ. A large value of θ indicates that there exists heterogeneity between the systems. This frailty model has three parameters (i.e., β, η; θ) to be estimated. For the data given by (1), the parameters can be estimated using the maximum likelihood estimation method. Since z_i is unobserved, the ordinary likelihood function has to be replaced by the marginal likelihood obtained by unconditioning with respect to z_i.

Once the parameters are specified, the frailty z_i of the ith unit can be estimated. For the above-mentioned frailty model, the frailty of the ith unit is estimated by

$$z_i = (\tfrac{1}{\theta} + n_i)/[\tfrac{1}{\theta} + M(\tau_i)]. \tag{23}$$

3.2. Comparison of the EAF-based approach with the frailty model

We compare the EAF-based approach with the frailty model from four aspects: definitions of EAF and frailty, evaluation methods of heterogeneity, determination of fleet heterogeneity and interpretations of EAF and frailty.

Definition of EAF and frailty: Both the EAF and frailty are defined relative to an "average" system; the EAF is defined in terms of the MTBF, and the frailty is defined in terms of intensity function. From (23), the frailty is asymptotically reciprocally proportional to the EAF as $\theta \to \infty$. These imply that the two models are somehow similar and closely related.

Evaluation of heterogeneity: Parameter θ of the frailty model describes the heterogeneity and is estimated using a parametric approach. The approach needs to make the assumptions about the types of repair process and frailty distribution, and the parameter estimation needs to use an unconditional approach. These make this approach complex and the results unreliable due to possible misspecification. On the other hand, the EAF-based approach does not make any assumptions since it is essentially a non-parametric approach. Therefore, it is simpler and more reliable than the frailty model.

Determination of fleet heterogeneity: The EAF-based approach defines a critical value of ρ_{α} to objectively determine whether the heterogeneity exists while the frailty-based approach does not define such a critical value

so that the determination of fleet heterogeneity is subjective to some extent.

Interpretation of EAF and frailty: The EAF of a system is the ratio of two statistics (i.e., system MTBF and fleet MTBF) while the frailty is the ratio of two functions (i.e., system intensity and baseline intensity). Thus, the former can be viewed as being "observable" since the MTBF can be directly computed from the data while the latter is unobserved since it can be only estimated based on the data and assumed models.

3.3. A robust estimate for frailty of a system

It is noted that the frailty given by (23) is highly sensitive to the value of θ: when $\theta \to 0$, $z_i \to 1$; and when $\theta \to \infty$, $z_i \to \frac{n_i}{M(\tau)} = \frac{1}{\alpha_i}$. To make z_i robust, we define a non-parametric estimate of the frailty. Consider the single censoring case with $\tau_i = \tau, (1 \le i \le n)$. Integrating the two sides of (21) from zero to τ, we have

$$LHS = n_i, \quad RHS = z_i M(\tau). \tag{24}$$

Letting $LHS = RHS$ yields

$$z_i = \frac{n_i}{M(\tau)} = \frac{1}{\alpha_i}. \tag{25}$$

This estimate is independent of θ and can be interpreted as the ratio of the system's failure numbers and expected failure number of the fleet in a common observation window $(0, \tau)$. The frailty given by (25) is reciprocally proportional to the EAF. According to this definition, the frailty becomes "observable". The definition can be extended to the multiple censoring case: *the frailty of system i is the ratio of the system's failure numbers and expected failure number of the fleet evaluated at τ_i.*

3.4. Example 1

In this subsection, we illustrate the frailty-based and EAF-based approaches by a real-world example.

3.4.1. Data

The example comes from [31] and deals with the failure processes of 9 sugarcane harvesters, observed in a common period of 200 days. [31] Present the failure time data of the systems with a figure, from which we extract the data shown in the first two rows of Table 1.

Table 1: Failure numbers of sugarcane harvesters in 200 days

i	1	2	3	4	5	6	7	8	9	δ
n_i	11	14	14	15	19	14	11	13	16	0.1752
z_i, (23)	0.9989	0.9991	0.9991	0.9991	0.9994	0.9991	0.9989	0.9990	0.9992	0.0002
z_i, (25)	0.7795	0.9921	0.9921	1.063	1.346	0.9921	0.7795	0.9213	1.134	0.1799
μ_i	18.18	14.29	14.29	13.33	10.53	14.29	18.18	15.38	12.5	0.1704
α_i	1.283	1.008	1.008	0.941	0.743	1.008	1.283	1.085	0.882	0.1704

3.4.2. Results of the frailty-based approach

In [31], the authors estimated assume that the repair is perfect, the time to failure follows the Weibull distribution with shape parameter β and scale parameter η, and the frailty follows the gamma distribution with parameter θ. The maximum likelihood estimates (MLEs) of the parameters are $(\beta, \eta; \theta) = (1.32, 15.02; 0.56 \times 10^{-4})$. Since the estimate of θ is nearly equal to zero, the unobserved heterogeneity among the systems is deemed to be non-existent. The frailties estimated from (23) are shown in the third row of Table 1 and the frailties estimated from (25) are shown in the fourth row. The last column shows the coefficient of variation (δ) of the nine values of each row. It is expected that $\delta(z_i)$ should be close to $\delta(n_i)$. However, (23) does not meet this expectation while (25) meets the expectation, and hence its appropriateness is confirmed.

3.4.3. Results of the proposed approach

Since the observation windows of the systems are the same, it is easy to evaluate the MCF and MTBF of the fleet, which are $M(200) = 14.11$ and $\mu_F = 14.17$ days, respectively. It is also easy to calculate the MTBFs and EAFs of individual systems using the following relations

$$\mu_i = 200/n_i, \quad \alpha_i = M(200)/n_i \tag{26}$$

The results are shown in the last two rows of Table 1. The average and standard deviation of the EAFs are $\mu_\alpha = 1.0267$ and $\sigma_\alpha = 0.1749$, respectively; and their ratio is $\rho_\alpha = 5.870$, which is much larger than 1.96. Therefore, the fleet is homogeneous. This conclusion is consistent with the one of [31] though $\delta(z_i)$ obtained from (23) is significantly different from $\delta(z_i)$ obtained from (25).

4. Further illustrations

In this section, we further illustrate the proposed approach and its superiority using two examples, which have been analyzed by [14].

4.1. Example 2

4.1.1. Data

Consider a fleet of 10 units, and each has a power-law MCF given by (22) with parameters $\beta = 1.5$ and $\eta = 100$. A failure is restored by minimal repair; and the censoring time of each unit is randomly generated from the uniform distribution over (0, 500) so that $\tau_i > t_{n_i}$. The randomly generated data are shown in the upper part of Table 2 and the censoring times of units are shown in the last row. From the table yields four important statistics:

$$N = 46, \tau = 470.9, T_3(\tau) = 2601.7, \mu_F = 56.56. \tag{27}$$

Table 2: Failure processes of units

Unit	1	2	3	4	5	6	7	8	9	10
t_1	245	131		17	105		19	70	108	32
t_2	293	194		51	209		49	174	188	103
t_3	353	277		56	235			195	235	
t_4	356	278		188	255			215		
t_5	378	279		206	262			271		
t_6	401	324		216	270			356		
t_7	424			231	281					
t_8	466			249	331					
t_9				251						
t_{10}				338						
t_{11}				372						
τ_i	470.9	336.1	50.5	441.3	348.3	111.6	92.8	374.5	239.4	136.3

4.1.2. Evaluation of MTBF and EAF

The information that the approach of in [14], the authors estimated requires for estimating the MTBFs and EAFs of units are shown in the first three columns of Table 3; and the estimated MTBFs and EAFs are shown in the 4th and 5th columns. As can be seen, using the approach of [14] cannot estimate the MTBFs and EAFs of Units 3 and 6 due to $n_3 = n_6 = 0$.

Although no failure occurs for Units 3 and 6, these two units can and should be included in the computation of MCF since their censoring information has an influence on $s(t_k)$. The results obtained from the proposed approach are shown in the 6th to 8th columns. Different from the approach of [14], which first estimates μ_i (at τ_i) and then estimates α_i, the improved method first estimates α_i (at τ_i) and then estimates μ_i (at τ).

Table 3: MTBFs and EAFs of units obtained from two methods

Unit	Jiang et al. [14]				Improved approach			ε_i
	τ_i	n_i	μ_i (12)	α_i (13)	$M(\tau_i)$	α_i (17)	μ_i (18)	
1	470.9	8	58.86	1.467	10.13	1.267	71.65	0.1785
2	336.1	6	56.02	1.396	6.051	1.009	57.04	0.0180
3	50.50	0			0.4000	(0.7315)	(41.37)	
4	441.3	11	40.12	1.000	9.135	0.8304	46.97	0.1458
5	348.3	8	43.54	1.085	6.301	0.7876	44.55	0.0227
6	111.6	0			1.108	(2.027)	(114.6)	
7	92.80	2	46.40	1.157	0.7333	0.3667	20.74	1.237
8	374.5	6	62.42	1.556	7.635	1.272	71.97	0.1327
9	239.4	3	79.80	1.989	3.251	1.084	61.29	0.3019
10	136.3	2	68.15	1.699	1.251	0.6256	35.38	0.9261

Since the reference process is an "average process", it is possible to estimate the MTBFs and EAFs of Units 3 and 6 based on the following two assumptions:

(a) The average of EAFs of all the units equals to 1, and

(b) From (17) we can assume that $\alpha_i = cM(\tau_i)$ for $i = 3$ and 6.

The value of c can be estimated from the first assumption, which is 1.829. From the second assumption yields $\alpha_3 = 0.7315$ and $\alpha_6 = 2.027$. Further, from (18) yields $\mu_3 = 41.37$ and $\mu_6 = 114.6$. These estimates are bracketed in Table 3.

The last column shows the relative errors between the MTBFs estimated from (12) and (18). To examine the influence of censoring time τ_i (or failure number n_i) on the accuracy of MTBF estimate, we examine the plot of ε_i vs.

n_i shown in Figure 1. As seen from the figure, ε_i quickly increases as n_i decreases (or $1/n_i$ increases), implying that the MTBF estimates obtained from (12) and (18) are nearly the same when n_i is large. Since the estimate obtained from (12) is accurate when n_i is large, this actually validates the appropriateness of (18).
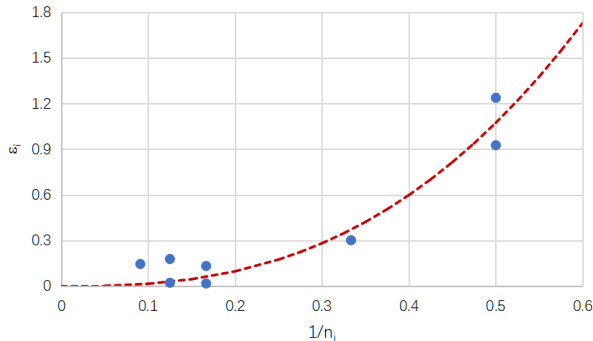

Figure 1: Plot of ε_i vs. n_i for Example 2

4.1.3. Estimates of fleet MTBF

As mentioned in Section 2, the fleet MTBF can be estimated by a TTT-based approach given by (3) and by an MCF-based approach given by (6). We illustrate that they are different for multiple censoring case.

From (27) and Table 3 we have $\tau = \tau_1 = 470.9$ and $M(\tau) = M(\tau_1) = 10.13$. Applying these to (6) yields $\mu_{FN} = 46.46$. On the other hand, from (3) we have $\mu_F = 56.56$. This implies that the MCF-based approach underestimates the fleet MTBF with a relative error of 17.85%. This also indirectly illustrates that the Nelson MCF estimator can be inaccurate for the multiple censoring case, and its accuracy can be measured by the relative error given by

$$\varepsilon_F = |1 - \frac{\mu_{FN}}{\mu_F}|. \qquad (28)$$

The larger the relative error is, the less accurate the Nelson MCF estimator is.

4.1.4. Heterogeneity evaluation

The computational results of the heterogeneity are shown in Table 4. The results obtained from the approach of [14] are shown in the second row, the results obtained from the improved method with the EAFs of Units 3 and 6 being excluded are shown in the third row, and the results obtained from the improved method with the EAFs of Units 3 and 6 being included are shown in the last row. As seen, the values of ρ_α for these three cases are quite different, but they all are larger than 1.96, implying that the fleet is homogeneous.

Table 4: Fleet heterogeneity measures obtained from different methods

Approach	Units 3 and 6	μ_α	σ_α	ρ_α
[14]	Excluded	1.419	0.3340	4.248
This paper	Excluded	0.9052	0.3153	2.871
	Included	1.000	0.4587	2.180

The difference in ρ_α can be explained by a large ε_F. Since Units 7, 9 and 10 only have 2, 3 and 2 failures, respectively, their MTBFs are considerably overestimated by (12). This reduces the dispersion of the MTBFs and EAFs and leads to a large value of ρ_α. This confirms the fact that the units' EAF estimates obtained from the proposed approach are more robust than those obtained from the approach of [14].

4.1.5. Distribution of EAF

The frailty is usually assumed to follow the gamma distribution with mean one. It is possible to check whether this assumption holds for the EAF. Fitting the EAFs shown in the 7th column of Table 3 to four typical 2-parameter distributions: gamma distribution with parameters u and v, Weibull distribution with parameters β and η, normal distribution with parameters μ and σ, and lognormal distribution with parameters μ_l and σ_l. The MLEs of the parameters are shown in Table 5, where $\theta_1 = u, \beta, \mu$ or μ_l; $\theta_2 =, \eta, \sigma, v$ or σ_l; and $\ln(L)$ is the log-likelihood value. As seen, the gamma distribution provides the best fit in terms of $\ln(L)$.

Table 5: MLEs of distributional parameters

	Example 2				Example 3			
	Gamma	Weibull	Normal	Lognormal	Gamma	Weibull	Normal	Lognormal
θ_1	5.498	2.434	1.000	-0.0936	4.463	2.005	1.229	0.0896
θ_2	0.1819	1.130	0.4352	0.4413	0.2753	1.397	0.6631	0.4580
$ln(L)$	-5.034	-5.423	-5.871	-5.074	-10.37	-11.50	-13.11	-9.459

The probability plot of a distribution is often used to check the appropriateness of the distribution. Let α_j ($1 \leq j \leq n$) denote the ordered EAFs sorted in ascending order, $p_j = (j - 0.3)/(n + 0.4)$, and $q_j = F^{-1}(p_j; u, 1)$ denote the inverse function of the gamma distribution with parameters u and 1. The gamma probability plot is given by

$$\alpha_j = vq_j. \qquad (29)$$

Figure 2 shows the plot of α_j vs. q_j for $u = 5.498$. Regression yields $v = 0.1879$, which is close to the MLE of v with a relative error of 3.29%. This confirms the appropriateness of the gamma distribution for fitting the EAF data.
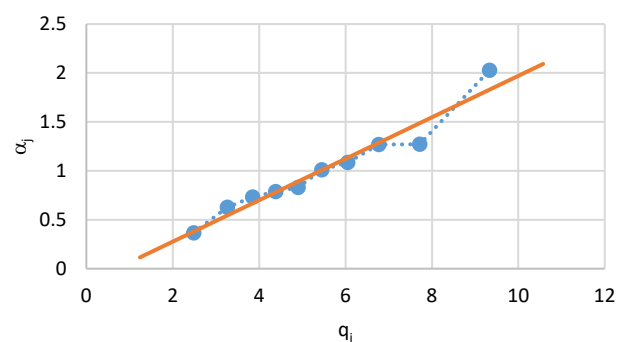

Figure 2: Plot of α_j vs. q_j for Example 2

4.2. Example 3

4.2.1. Data

The data shown in Table 6 come from Proschan [33] and deal with the inter-failure times (in days) of 13 air conditioning systems before overhauls. For this example, the censoring time meets the relation of $\tau_i = t_{n_i}$. From the table, we have

$$N = 192, \tau = 2074, T_3(\tau) = 18088, \mu_F = 94.21. \quad (30)$$

Table 6: Inter-failure times of air conditioning systems

$j\backslash i$	1	2	3	4	5	6	7	8	9	10	11	12	13
1	194	413	90	74	55	23	97	50	359	50	130	487	102
2	15	14	10	57	320	261	51	44	9	254	493	18	209
3	41	58	60	48	56	87	11	102	12	5		100	14
4	29	37	186	29	104	7	4	72	270	283		7	57
5	33	100	61	502	220	120	141	22	603	35		98	54
6	181	65	49	12	239	14	18	39	3	12		5	32
7		9	14	70	47	62	142	3	104			85	67
8		169	24	21	246	47	68	15	2			91	59
9		447	56	29	176	225	77	197	438			43	134
10		184	20	386	182	71	80	188				230	152
11		36	79	59	33	246	1	79				3	2
12		201	84	27		21	16	88				130	14
13		118	44			42	106	46					230
14			59			20	206	5					66
15			29			5	82	5					61
16			118			12	54	36					34
17			25			120	31	22					
18			156			11	216	139					
19			310			3	46	210					
20			76			14	111	97					
21			26			71	39	30					
22			44			11	63	23					
23			23			14	18	13					
24			62			11	191	14					
25						16	18						
26						90	163						
27						1	24						
28						16							
29						52							
30						95							

4.2.2. Estimation of MTBF and EAF

The results obtained from the approach of [14] are outlined in the 2nd to the 5th column of Table 7. The Nelson MCF estimators at unit censoring times are shown in the 6th column and the EAFs and MTBFs of units obtained from the improved methods are shown in the 7th and 8th columns, respectively. It is noted that $\tau = \tau_7 = 2074$ and $M(\tau) = M(\tau_7) = 23.74$. Applying these to (6) yields $\mu_{FN} = 87.38$, which is slightly smaller than the fleet MTBF shown in (30) with a relative error of 7.25%. An underestimate of μ_{FN} implies an overestimate of $M(\tau)$. Compared with Example 2, ε_F is relatively small.

Figure 3 shows the plot of ε_i vs. n_i. As expected, ε_i increases as n_i decreases. The average and maximum of ε_i's are 0.0683 and 0.2258, respectively, implying that the

MTBF estimates obtained from (12) and (18) are fairly close to each other for the current example.

Table 7: MTBFs and EAFs of units obtained from two methods for Example 3

| Unit	[14]				Improved approach			ε_i
	τ_i	n_i	μ_i (12)	α_i (13)	$M(\tau_i)$	α_i (17)	μ_i (18)	
1	493	6	82.16	1.378	4.269	0.7115	67.03	0.2258
2	1851	13	142.4	2.389	19.99	1.537	144.8	0.0167
3	1705	24	71.04	1.191	18.80	0.7835	73.81	0.0375
4	1314	12	109.5	1.837	12.95	1.080	101.7	0.0767
5	1678	11	152.5	2.559	18.22	1.656	156.0	0.0221
6	1788	30	59.60	1.000	19.28	0.6426	60.54	0.0155
7	2074	27	76.81	1.288	23.74	0.8791	82.82	0.0725
8	1539	24	64.12	1.075	16.23	0.6763	63.71	0.0065
9	1800	9	200.0	3.355	19.57	2.174	204.8	0.0234
10	639	6	106.5	1.786	6.111	1.018	95.95	0.1100
11	623	2	311.5	5.226	5.933	2.966	279.5	0.1145
12	1297	12	108.1	1.813	12.71	1.059	99.80	0.0830
13	1287	16	80.43	1.349	12.61	0.7879	74.23	0.0836
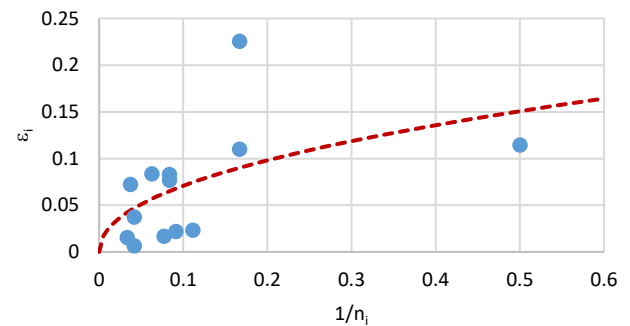
Figure 3: Plot of ε_i vs. n_i for Example 3

4.2.3. Heterogeneity evaluation

The fleet heterogeneity measure obtained from [14] is $\rho_\alpha = 1.715$, implying that the fleet is heterogeneous. The value of ρ_α obtained from the improved method is 1.780, which is very close to the above estimate. The similarity of the results obtained from the two approaches can be explained by the average failure number of units (i.e., N/n), which is 14.8. For a dataset with a large value of N/n, both the method of [14] and the improved method will give similar results and draw similar conclusions; otherwise, the results and conclusions are probably different.

4.2.4. Distribution of EAF

Fitting the EAFs shown in the 7th column of Table 7 to the four 2-parameter distributions yields the MLEs of the parameters shown in the RHS of Table 5. In terms of the log-likelihood value, the lognormal distribution provides the best fit to the data. To further validate, we examine the lognormal probability plot. Let $q_j = F^{-1}(p_j; 0, 1)$ denote the inverse function of the standard normal distribution. The lognormal probability plot is given by

$$q_j = \frac{\ln(\alpha_j) - \mu_l}{\sigma_l}. \quad (31)$$

Figure 4 shows the plot of q_j vs. $ln\ (\alpha_j)$. As can be seen, the plot is concave, implying that the 2-parameter lognormal distribution is not appropriate for fitting the EAF data.
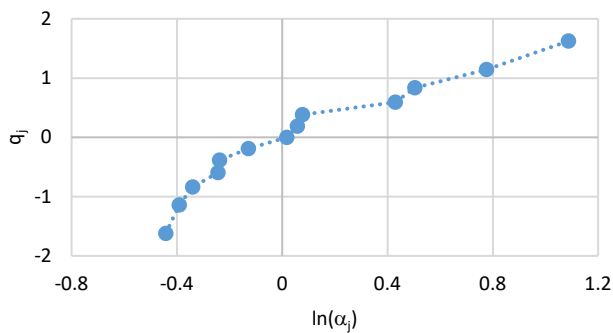

Figure 4: Plot of q_j vs. $\ln(\alpha_j)$ for EAF data of Example 3

Figure 4 indicates that urther analysis shows that the 3-parameter lognormal distribution with a location parameter γ can be appropriate for fitting the data. This is validated by the probability plot of the 3-parameter lognormal distribution shown in Figure 5. The least square estimates of the parameters are $\gamma = 0.5934$, $\mu_l = -1.016$, $\sigma_l = 1.259$, and the corresponding log-likelihood value is $\ln(L) = -6.854$, which is much larger than the log-likelihood value of the 2-parameter lognormal distribution. From the fitted model yields $\rho_\alpha = 0.8843$, and hence the heterogeneity of the fleet is confirmed.
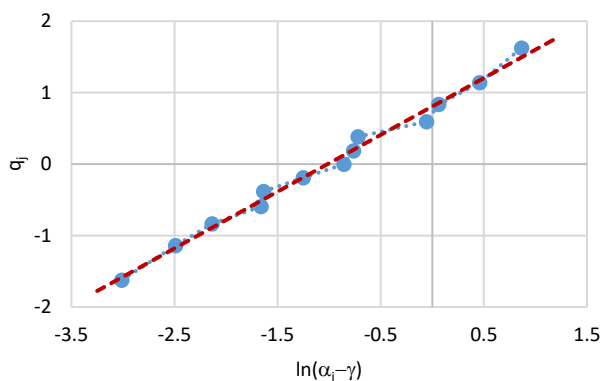
Figure 5: Probability plot of the 3-parameter lognormal distribution of the EAF for Example 3

5. Conclusions

In this paper, we have identified the problems of the heterogeneity evaluation method presented in [14] and proposed an improved method, whose main advantages are:

(a) it estimates the MTBFs of units at a common time point so that the results are much more reliable than those obtained from the method given in [14], and

(b) the MTBFs of zero-failure units can be inferred under two reasonable assumptions. This is different from those approaches that exclude the units with an

inadequate amount of failure data from the analysis [16].

These have been illustrated by three examples.

The main conclusions and/or findings of the paper have been:

- The MTBFs and EAFs of units estimated from the method [14] and the improved method are fairly consistent if the average failure number of units is large and can be considerably different otherwise.

- The fleet MTBF estimates obtained from the Nelson MCF estimator and TTT approach are different for multiple censoring data.

- The frailty of a system can be defined as the ratio of the system's failure numbers and expected failure number of the fleet evaluated in a common observation window. Such defined frailty is reciprocally proportional to the EAF.

- A simple distribution such as the 2-parameter gamma or lognormal distribution may be inappropriate for describing the EAF or frailty in the presence of heterogeneity.

In this paper, we have confined to the fleet composed by similar repairable systems rather than multi-component repairable systems and stressed MTBFs of units rather than their trends. The MTBF estimates of the units are based on the constant EAF assumption, whose reasonability requires further validation. The data considered in this paper only contain failure time information without other information such as failure causes, failure modes, maintenance types and censoring types. Analysis for the failure process with such information is a challenging issue and needs further research. Finally, a topic for future study is to develop a cluster analysis method for classifying the units of a heterogeneous fleet (e.g., automated guided vehicles [34]) into several homogeneous groups based on the EAFs of the units.

References

[1] Ascher H, Feingold H. *Repairable Systems – Modeling, Inference, Misconceptions and their Causes*. Marcel Dekker, New York, 1984.

[2] Lindqvist BH. "Maintenance of repairable systems," in *Complex system maintenance handbook*, pp. 235-261, 2008, doi: 10.1007/978-1-84800-011-7_10

[3] Lindqvist BH. "On the statistical modeling and analysis of repairable systems," *Statistical Science*, vol. 21, no. 4, pp. 532-551, 2007, doi: 10.1214/088342306000000448.

[4] Krivtsov VV. "Practical extensions to NHPP application in repairable system reliability analysis," *Reliability Engineering and System Safety*, vol. 92, no. 5, pp. 560-562, 2007, doi: 10.1016/j.ress.2006.05.002.

[5] Jiang R, Huang C. "Failure patterns of repairable systems and a flexible intensity function model," *International Journal of Reliability and Applications*, vol. 13, no. 2, pp. 81-90, 2012, doi: koreascience.kr/article/JAKO201217752421606.

[6] R. Jiang, Y. Guo. and strong, "Estimating Failure Intensity of a Repairable System to decide on its Preventive Maintenance or Retirement," *International journal of performability engineering*, vol. 10, no. 6, pp. 577-588, 2014, doi: 10.23940/ijpe.14.6.p577.mag.

[7] Nelson W. "Graphical analysis of system repair data," *Journal of Quality Technology*, vol. 20, pp. 24-35, 1988, doi: 10.1080/00224065.1988.11979080.

[8] Nelson W. *Recurrent Events Data Analysis for Product Repairs, Disease Recurrences, and Other Applications*, Society for Industrial and Applied Mathematics (SIAM): Philadelphia, 2003.

[9] Asfaw ZG, Lindqvist BH. "Unobserved heterogeneity in the power law nonhomogeneous Poisson process," *Reliability Engineering and System Safety*, vol. 134, pp. 59-65, 2015, doi: 10.1016/j.ress.2014.10.005

[10] Cui D., Sun Q., Xie M., "Robust statistical modeling of heterogeneity for repairable systems using multivariate gaussian convolution processes," *IEEE Transactions on Reliability*, vol. 72, no. 4, pp. 1493-1506, 2023, doi: 10.1109/TR.2023.3235889.

[11] Liu X., Vatn J., Dijoux Y., Toftaker H., "Unobserved heterogeneity in stable imperfect repair models," *Reliability Engineering and System Safety*, vol. 203, pp. 107039, 2020, doi: 10.1016/j.ress.2020.107039.

[12] Zaki R, Barabadi A, Barabady J, Qarahasanlou AN. "Observed and unobserved heterogeneity in failure data analysis," *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, vol. 236, no. 1, pp. 194-207, 2022, doi: 10.1177/1748006X211022538

[13] Brown B., Liu B., McIntyre S., Revie M., "Reliability evaluation of repairable systems considering component heterogeneity using frailty model," *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, SAGE journal*, vol. 237, no. 4, pp. 654-670, 2023, doi: 10.1177/1748006X221109341.

[14] Jiang R., Li F., Xue W., Cao Y., et al., "A robust mean cumulative function estimator and its application to overhaul time optimization for a fleet of heterogeneous repairable systems," *Reliability Engineering and System Safety*, vol. 236, pp. 109265, 2023, doi: 10.1016/j.ress.2023.109265.

[15] Garmabaki, AHS, Ahmadi A, Block J, Pham H, Kumar U. "A reliability decision framework for multiple repairable units," *Reliability Engineering & System Safety*, vol. 150, pp. 78-88, 2016, http://dx.doi.org/10.1016/j.ress.2016.01.020

[16] Garmabaki AHS, Ahmadi A, Mahmood YA, Barabadi A. "Reliability modelling of multiple repairable units," *Quality and Reliability Engineering International*, vol. 32, no. 7, pp. 2329-2343, 2016, doi: 10.1002/qre.1938

[17] Weckman GR, Shell RL, Marvel JL. "Modeling the reliability of repairable systems in the aviation industry," *Computers and Industrial Engineering*, vol. 40, pp. 51-63, 2001, doi: 10.1016/S0360-8352(00)00063-2.

[18] Giorgio M, Guida M, Pulcini G. "Repairable system analysis in presence of covariates and random effects," *Reliability Engineering and System Safety*, vol. 131, pp. 271-281, 2014, doi: 10.1016/j.ress.2014.04.009.

[19] Meeker WQ, Hong Y. "Reliability meets big data: opportunities and challenges," *Quality Engineering*, vol. 26, pp. 102-116, 2014, doi: 10.1080/08982112.2014.846119.

[20] Navas MA, Sancho C., Carpio J. "Reliability analysis in railway repairable systems," *International Journal of Quality & Reliability Management*, vol. 34, no. 8, pp. 1373-1398, 2017, doi: 10.1108/IJQRM-06-2016-0087.

[21] Hong Y, Zhang M, Meeker WQ. "Big data and reliability applications: the complexity dimension," *Journal of Quality Technology*, vol. 50, no. 2, pp. 135-149, 2018, doi: 10.1080/00224065.2018.1438007.

[22] Peng W, Shen L, Shen Y, Sun Q, "Reliability analysis of repairable systems with recurrent misuse-induced failures and normal-operation failures," *Reliability Engineering and System Safety*, vol. 171, pp. 87-98, 2018, doi: 10.1016/j.ress.2017.11.016.

[23] Si W, Love E, Yang Q. "Two-state optimal maintenance planning of repairable systems with covariate effects," *Computers & Operations Research*, vol. 92, pp. 17-25, 2018, doi: 10.1016/j.cor.2017.11.007.

[24] Liu X, Pan R. "Analysis of large heterogeneous repairable system reliability data with static system attributes and dynamic sensor measurement in big data environment," *Technometrics*, vol. 62, no. 2, pp. 206-222, 2020, doi: 10.1080/00401706.2019.1609584.

[25] Sharma G, Rai RN. "Failure modes based censored data analysis for repairable systems and its industrial perspective," *Computers & Industrial Engineering*, vol. 158, pp. 107439, 2021, https://doi.org/10.1016/j.cie.2021.107439

[26] Jiang R, Li F, Xue W, Lin L, Li X, Zhang K. "Identification and treatment of extreme inter-failure times from a fleet of repairable systems,". In G. Abdul-Nour et al. (eds.), *17th WCEAM Proceedings*, Lecture Notes in Mechanical Engineering, pp. 417-431, 2023, https://doi.org/10.1007/978-3-031-59042-9_34

[27] Jiang R, Xue W, Cao Y. "Analysis for influence of maintenance and manufacturing quality on reliability of repairable systems," In *Advances in Reliability and Maintainability Methods and Engineering Applications*, pp. 385-403. Cham: Springer Nature Switzerland, 2023, https://doi.org/10.1007/978-3-031-28859-3_15

[28] Jiang R. "Overhaul decision of repairable systems based on the power-law model fitted by a weighted parameter estimation method," *In Asset Intelligence through Integration and Interoperability and Contemporary Vibration Engineering Technologies*, Chapter 29, pp. 277-286, Springer, 2018. doi: 10.1007/978-3-319-95711-1_28.

[29] Block J, Ahmadi A, Tyrberg T, Kumar U. "Fleet-level reliability analysis of repairable units: a non-parametric approach using the mean cumulative function," *International Journal of Performability Engineering*, vol. 9, no. 3, pp. 333-344, 2013.

[30] Block J, Ahmadi A, Tyrberg T, Kumar U. "Fleet-level reliability of multiple repairable units: a parametric approach using the power law process," *International Journal of Performability Engineering*, vol. 10, no. 3, pp. 239-250, 2014, doi: 10.23940/ijpe.14.3.p239.mag

[31] Brito ÉS, Tomazella VLD, Ferreira PH. "Statistical modeling and reliability analysis of multiple repairable systems with dependent failure times under perfect repair," Reliability Engineering and System Safety, vol. 222, pp. 108375, 2022, doi: 10.1016/j.ress.2022.108375.

[32] Percy DF, Alkali BM. "Generalised proportional intensities models for repairable systems," *IMA Journal of Management Mathematics,*

vol. 17, pp. 171-185, 2006, doi: 10.1093/imaman/dpi034.

[33] Proschan F. "Theoretical explanation of observed decreasing failu re rate," *Technometrics,* vol. 5, no. 3, pp. 375-383, 1963, doi:10.1080/ 00401706.1963.10490105.

[34] Rashidi H, Matinfar F, Parand FA. "Automated guided vehicles-a review on applications, problem modeling and solutions," *Intern ational Journal of Transportation Engineering*, vol. 8, no. 3, pp. 261-2 78, 2021, doi: 10.22119/IJTE.2021.246669.1531.

RENYAN JIANG received his B.S. and M.S. degrees from Wuhan University of Technology, China, in 1982 and 1985, respectively and his Ph.D. in 1996 from University of Queensland, Australia. His research interests are in various aspects of quality, reliability and maintenance. He is the author or co-author of five reliability/maintenance related books and has published more than 250 papers.

XIA XU has done his bachelor's degree from Wuhan University of Technology in 2005. He is the leading craftsman at Penta Laser (Zhejiang) Co., Ltd. He invented 15 patents, published 9 papers, and won the first prize of Zhejiang Machinery Industry Sci. & Tech.

KUNPENG ZHANG received his B.S. (2013) degree from Nanyang Institute of Technology, M.S. (2016) and Ph.D. (2021) degrees from Xiamen University. His research interests are micro/nano 3D printing, microfluidic and the reliability engineering and system safety.

YU CAO received his B.S. (2003) degree from Harbin Institute of Technology (HIT), M.S. (2006) and Ph.D. (2009) degrees from Huazhong University of Science & Technology (HUST). His research interests are laser material processing, Reliability Engineering and System Safety.

Biclustering Results Visualization of Gene Expression Data: A Review

Haithem Aouabed[*][1] [iD]**, Mourad Elloumi** [2]**, Fahad Algarni** [2]

[1]University of Sfax, Department of Computer Sciences, Faculty of Economic Sciences and Management, Sfax, 3018, Tunisia

[2]University of Bisha, Department of Computer Sciences and artificial intelligence, College of Computing and Information Technology, Bisha, 67714, Saudi Arabia

*Corresponding author: Haithem Aouabed, Mahdia road km 19 Sfax, +21626399303 &haithem.abdi@gmail.com

ABSTRACT: Biclustering is a non-supervised data mining method used to analyze gene expression data by identifying groups of genes that exhibit similar patterns across specific groups of conditions. Discovering these co-expressed genes (called biclusters) can aid in understanding gene interactions in various biological contexts. Biclustering is characterized by its bi-dimensional nature, grouping both genes and conditions in the same bicluster and its overlapping property, allowing genes to belong to multiple biclusters. Biclustering algorithms often produce a large number of overlapping biclusters. Visualizing these results is not a straightforward task due to the specific characteristics of biclusters. In fact, biclustering results visualization is a crucial process to infer patterns from the expression data. In this paper, we explore the various techniques for visualizing multiple biclusters simultaneously and we evaluate them in order to help biologists to better choose their appropriate visualization techniques.

KEYWORDS: Biclustering algorithms, Biclusters, Overlaps, Visualization, Visualization techniques

1. Introduction

Gene expression profiles, generated by advanced high-throughput technologies like microarrays, are depicted in a matrix format where rows correspond to genes, columns to experimental conditions and each matrix entry to the expression level of a gene under a specific condition. Clustering has been the primary technique for analyzing such voluminous genomic data, focusing on grouping genes (rows) that show similar expression across all conditions (columns), as noted by [1]. Traditional clustering methods including hierarchical clustering [2] and k-means clustering [3] have proven effective in gene expression analysis. However, to glean novel insights from biological data such as identifying genes associated with cancer progression, determining functions of unknown genes or developing new treatment approaches, it's essential to perform clustering across both dimensions: genes and conditions. Indeed, the field of biclustering or co-clustering has emerged as a valuable tool in genomic data analysis. This machine learning technique identifies groups of biological entities, such as genes, that display comparable behaviour under specific conditions. It is first used to analyze gene expression data in 2000 by [4]. Biclustering differs from traditional clustering in two key theoretical aspects: bi-dimensionality which involves

grouping genes and conditions together and overlap which permits genes to be part of multiple biclusters at the same time. In [5], the author conducted a comprehensive review of various biclustering algorithms, categorizing them based on their search methodologies.

Visualizing biclustering output allows for the identification of co-regulated gene clusters and experimental conditions with similar gene expression profiles. In fact, examining biclustering results visually provides a deeper understanding of the underlying relationships and trends within the expression data [6]. Nevertheless, due to the unique attributes of biclustering which are bi-dimensionality and the potential for overlaps, the representation of gene expression data often results in numerous intersecting biclusters. These are challenging to display comprehensively in an informative way in a single visual representation. Indeed, encapsulating the results of biclustering into a single, coherent visual format is far from straightforward. Finding novel insights from vast, intricate multi-dimensional datasets necessitates an effective synergy of data processing algorithms and the power of interactive visualization tools [7-11]. Such a blend has been effectively applied to biological datasets, exemplified by the biclustering of gene expression data [12]. While heatmaps [1] and parallel coordinates [13]

remain the go-to methods for visualizing individual biclusters [14-16], the real challenge emerges when attempting to concurrently visualize multiple biclusters on a single screen for bioinformaticians and analysts [17], [18].

In this review, we conduct an examination of the visualization methods applied to biclustering outcomes derived from gene expression data [17]. We focus on biclustering results visualization techniques that can show more than one bicluster in the same screen. The structure of the review is outlined as follows: First, we provide an overview of the biclustering concept as it applies to gene expression data. Second, we make a survey on the current methods available for visualizing several biclusters concurrently. Then, we evaluate these methods according to a set of predefined criteria. Next, we demonstrate practical applications of these methods through various tools and discuss the datasets employed for their validation. Finally, we offer our conclusion.

2. Biclustering of gene expression data concept

We start by giving a definition of the biclustering concept.

2.1. Definition

A *bicluster* is a group of genes that exhibit consistent patterns of expression across a specific set of conditions. These genes have similar expression levels or follow identical trends within these conditions [19]. We note that biclusters can *overlap*, meaning that individual genes or conditions may be part of multiple biclusters at the same time.

Formally, a bicluster can be defined as follows: Let I={1, 2, . . . , n} be a set of indices of n genes, J={1, 2, . . . ,m} be a set of indices of m conditions and $M (I, J)$ be a data matrix associated with I and J. A bicluster associated with the data matrix $M (I, J)$ is a couple (I', J') such that $I'⊆I$ and $J'⊆J$.

The *biclustering problem* can be formulated as follows: Given a data matrix M, construct a group of biclustersB_{opt} associated with M such that:

$$f(B_{opt}) = \max_{B \in BC(M)} f (B), \qquad (1)$$

where f is a function that evaluates the quality, or coherence, of a bicluster group and $BC(M)$ represents the set of all potential bicluster groups associated with M[20], [21]. Biclustering is an NP-hard problem [4], [22]. In fact, NP-hard problems (i.e., non-deterministic polynomial time problems) are a class of challenging problems that are often considered intractable, meaning that there is no known efficient algorithm for solving them a priory. The combinatorial nature of the search space and the multiple optimization criteria involved make biclustering of gene expression data an NP-hard problem.

2.2. Groups of biclusters

Bicluster groups can be classified as follows [22]. See Figure 1:
- Particular bicluster (a).
- Bicluster groups with unique rows and columns (b).
- Checkerboard bicluster groups without overlap (c).
- Bicluster groups with unique rows (d).
- Bicluster groups with unique columns (e).
- Tree-structured bicluster groups without overlap (f).
- Non-overlapping bicluster groups without exclusive membership (rows or columns) (g).
- Hierarchical bicluster groups with overlap (h).
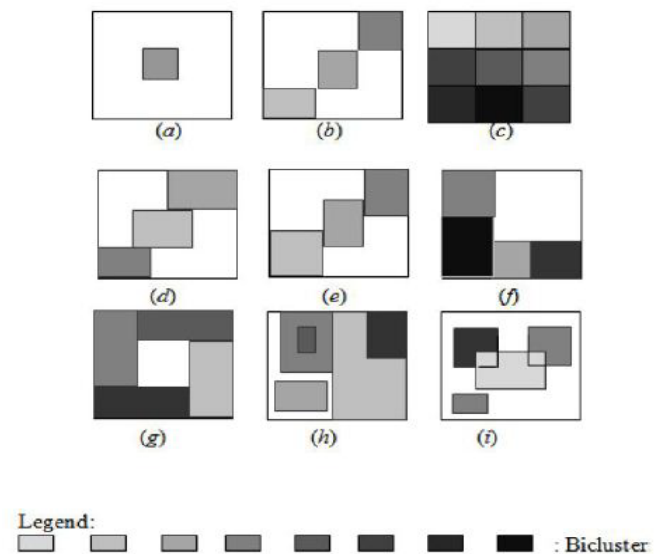- Randomly placed overlapping group of biclusters (i).

Figure 1: Types of bicluster groups [20]

We focus our review on how to visualize more than one bicluster with overlaps in the same screen (Figure 1(h) and Figure 1(i)).

2.3. Biclustering search methods

Biclustering is a computationally complex problem, often classified as NP-hard [4], [22]. As a result, heuristic approaches are typically employed to find approximate solutions. Given the variety of biclustering algorithms based on different search strategies, the following categorization can be identified [22], [23]:

- *Iterative row/column clustering*: This approach is a straightforward method that involves applying clustering algorithms to both the rows and columns of the expression matrix and then combining the results to identify biclusters. It is also known as two-way clustering [24], [25] or conjugated clustering [26]. This approach inherits the same benefits and drawbacks of clustering algorithms. Examples of algorithms that follow this approach include ITWC (Interrelated Two-Way Clustering) [25], CTWC (Coupled Two-Way Clustering) [24] and DCC (Double Conjugated Clustering) [26].

- *Divide and conquer*: This approach begins with a single bicluster encompassing the entire data matrix. It then recursively divides this matrix into two submatrices, creating two new biclusters. This process continues until a predetermined number of biclusters are generated that meet specific criteria. By breaking down the problem into smaller subproblems, this approach aims to accelerate the search for solutions. While this method is known for its speed, it can potentially ignore valuable biclusters if they are divided before being identified [21]. Examples of algorithms that follow this approach include the Hartigan biclustering algorithm [27] and Bimax[28].

- *Greedy iterative search*: This approach builds a solution incrementally using a specified quality measure. In the context of biclustering, at each step, submatrices of the data matrix are constructed by adding/removing rows or columns to maximize/minimize a particular function. This process continues until no further modifications can be made to any submatrix [20]. This approach shares the same strengths and weaknesses as the divide-and-conquer method. While it may make suboptimal choices and miss good biclusters, it can be very fast [21]. Examples of algorithms that follow this approach include CCA [4], OPSM [29], xMOTIFs[30], ISA [31], MSSRCC [32], QUBIC [33], COALESCE [34], CPB [35] and LAS [36].

- *Exhaustive bicluster enumeration*: This approach exhaustively explores all potential bicluster groups to identify the optimal solution that maximizes a specific evaluation function. Despite the capability of finding the best results, this approach is computationally expensive (i.e., time consuming). To alleviate this, biclustering algorithms often incorporate restrictions on the size or number of biclusters or employ pre- and post-filtering techniques [28]. Examples of algorithms that follow this approach include SAMBA [37], BiBit[38] and DeBi[39].

- *Distribution parameter identification*: This approach employs a statistical model to estimate distribution parameters and generate data by iteratively minimizing a specific criterion. Algorithms that follow this approach are capable of identifying the optimal biclusters, if they exist. However, due to their high computational complexity, they are often limited to analyzing biclusters with a specific size [21]. Examples of algorithms that follow this approach include Plaid model [40], Spectral biclustering[41], BBC [42] and FABIA [43].

Table 1 is a summary of biclustering search methods algorithms describing their characteristics.

Table 1: Evaluation of biclustering search methods

Biclutering search method	Algorithms	Advantages	Disadvantages
Iterative row/column clustering	ITWC [25] CTWC [24] DCC [26]	Find good results (i.e., clusters) Very fast	Sensitivity to noise datasets Scalability issues to large datasets
Divide and conquer	Hartigan algorithm [27] Bimax[28]	Very fast	Ignore good biclusters
Greedy iterative search	CCA [4] OPSM [29] xMOTIFs[30] ISA [31] MSSRCC [32] QUBIC [33] COALESCE [34] CPB [35] LAS [36]	Very fast	Ignore good biclusters
Exhaustive bicluster enumeration	SAMBA [37] BiBit[38] DeBi[39]	Find best solutions (i.e., biclusters)	Very Slow Time consuming
Distribution parameter identification	Plaid model [40] Spectral biclustering[41] BBC [42] FABIA [43]	Find best solutions (i.e., biclusters)	High complexity Time consuming

Most of these biclustering algorithms already mentioned in the previous subsection generated generally big size biclusters as a result. In the next section, we present the techniques used to visualize biclustering results of gene expression data [17].

3. Review of previous research

3.1. Heatmaps visualization

A heatmap serves as a bi-dimensional graphical representation that illustrates data values within a matrix structure. For gene expression data, the x-axis is allocated for conditions (columns) and the y-axis for genes (rows). Each matrix element a_{ij} denoting the expression level of the ith gene under the jth condition. It is depicted as a colored square (pixel) with the color intensity corresponding to a predefined scale. Typically, shades of green, red and black colors are chosen to align with the standard fluorescent dyes used in DNA microarrays: green signifies reduced expression or down-regulation, red indicates elevated expression or up-regulation and black represents a neutral expression level. To visualize a bicluster, its associated rows and columns are repositioned, generally to the top-left corner of the matrix [44]. Visualization techniques such as reordering or duplication are employed to display multiple biclusters within a single view. An example of heatmaps displaying biclusters is depicted in Figure 2.
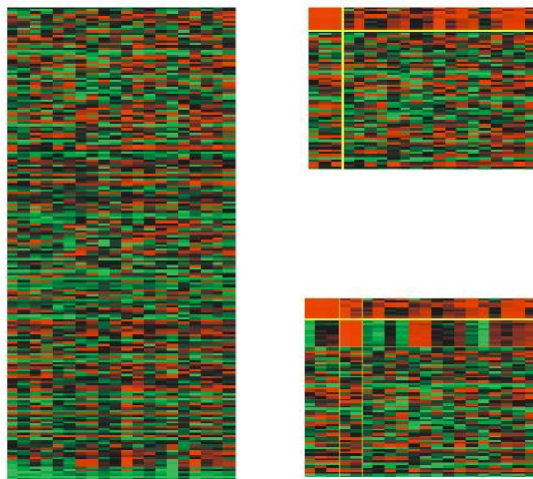

Figure 2: Heatmap representation of gene expression data (on the left). A bicluster at the upper left corner (top right). Two biclusters on the diagonal of the matrix (bottom right) [45]

3.1.1. Reordering techniques

To simultaneously visualize multiple biclusters, reordering is a viable strategy for heatmap representations. The literature presents various algorithms for this purpose.

In [46], the author proposed a heuristic iterative method that approaches the visualization of overlapping biclusters as an optimization challenge. This method introduces a reordering technique that draws parallels with the *hypergraph vertex ordering dilemma*, an extension of

the classic *minimal linear arrangement* or *graph ordering* problem. Initially, the heatmap matrix is transformed into a hypergraph which is then converted into a weighted undirected graph following a starting order that aligns with one of three predetermined configurations: a linear path, a singular loop or multiple loops. Then, the minimum linear arrangement problem (i.e., *MinLA*) is employed on the newly formed graph to discover an improved order. Next, the hypergraph is transformed into a different graph using the newly determined order. This iterative process continues until a satisfactory order is achieved or no further enhancements are observed. An illustrative example of this algorithm in action is presented in Figure 3.
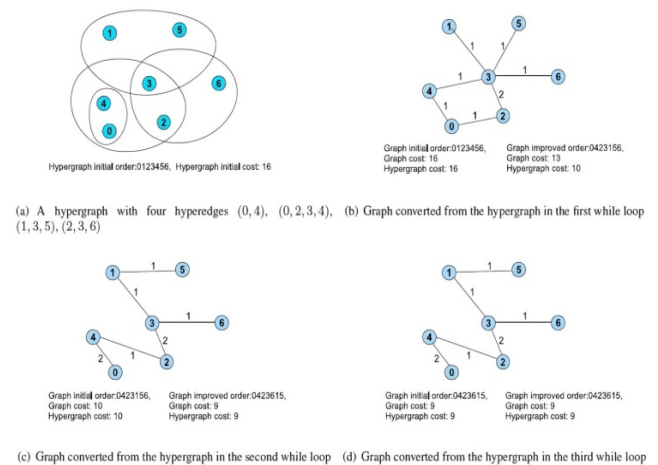

Figure 3: A practical demonstration of the reordering algorithm [46]

In [47], the algorithm presented aims to optimize the layout to enhance the visualization of the largest contiguous sections of biclusters from a gene expression matrix. Initially, the data is depicted as a *binary matrix* with rows representing genes or conditions and columns representing biclusters. The reordering approach is independently applied to both rows and columns to improve the visual quality of the biclusters. This optimization process counts four stages: The first stage named *simplify* eliminates redundant rows to reduce the problem's complexity. The second stage named *prearrange* seeks an optimal starting point for optimization by sequentially adding rows to a new order, ensuring each is positioned ideally. The third stage named *arrange* is the core of the algorithm where it aims to maximize an alignment score using a greedy strategy that repositions parts of a bicluster and permutes its constituent elements (genes or conditions) for better alignment. The final stage named *complexity* reintroduces the previously excluded rows into their new positions, thereby restoring the original problem's scale. An illustration of this technique's workflow is provided in Figure 4.
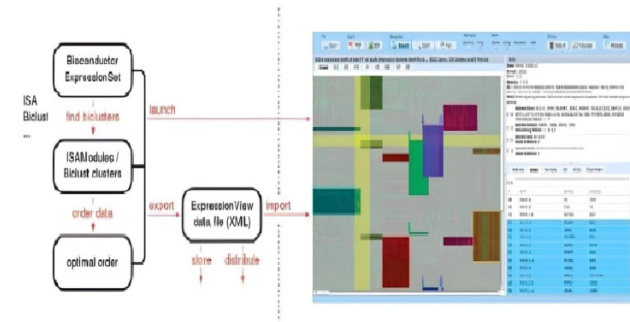
Figure 4: Different tasks of the proposed technique. Analysis part (on the left). Heatmap visualization (on the right) [47].

3.1.2. Duplication techniques

In certain instances within heatmap visualizations, reordering techniques can't display all biclusters adequately. It becomes necessary to replicate rows and columns to present the biclusters as continuous segments within a single heatmap. This duplication strategy has been proposed in various studies to enhance the clarity and continuity of bicluster representation.

The algorithm introduced by [48] aims to visualize biclusters and their overlaps as continuous regions within a single heatmap. Its core concept involves duplicating rows and columns to accurately depict overlapping biclusters. This approach is influenced by the *hypergraph superstring challenge* which refers to the physical mapping of genomes, as investigated by [49]. The algorithm presents a technique to limit the duplication of rows and columns as much as possible. This technique is executed separately on both rows and columns. It employs a data structure known as a *PQ tree*[50] which is helpful to arrange all potential columns to be adjacent, duplicating them if necessary, to form contiguous biclusters. Additionally, it utilizes a sequence of *REDUCE* operations that hierarchically organize the rows, thereby enhancing the overall quality of the visualization. An illustrative example employing two distinct expression matrices is demonstrated in Figure 5.
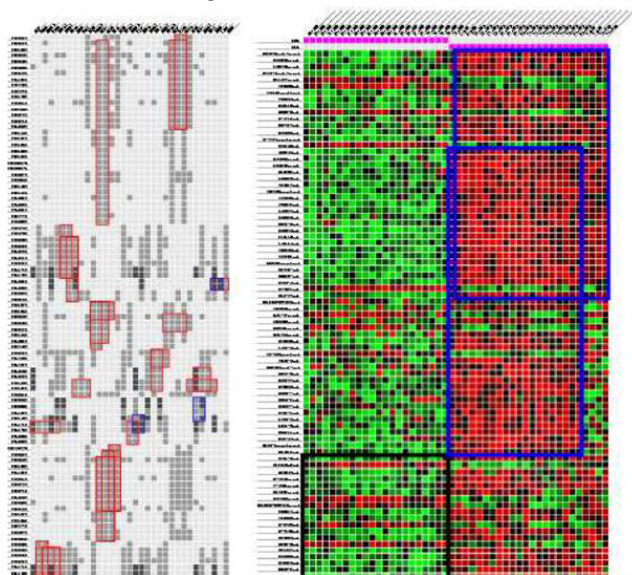

Figure 5: A number of biclusters were depicted [48]. Red rectangles (on the left) and blue rectangles (on the right)

In [51], the author developed a biclustering layout algorithm alongside an interactive visualization interface for illustrating multiple biclusters. The initial phase of their algorithm involves translating the heatmap into grayscale values through linear interpolation, spanning from the minimum to the maximum values within the data matrix. Next, each bicluster is designated with a unique color. When selected, biclusters are highlighted in a *semi-transparent yellow hue* which merges additively in areas of overlap, although users retain the option to customize their colors. To enable analysts to selectively visualize biclusters in an adjacent manner, the algorithm incorporates both reordering and duplication strategies for rows and/or columns. This interactive feature significantly reduces the occurrence of duplicates and marginally enhances the method's scalability. Figure 6 illustrates varied heatmap visualizations drawn from three distinct datasets containing multiple biclusters.
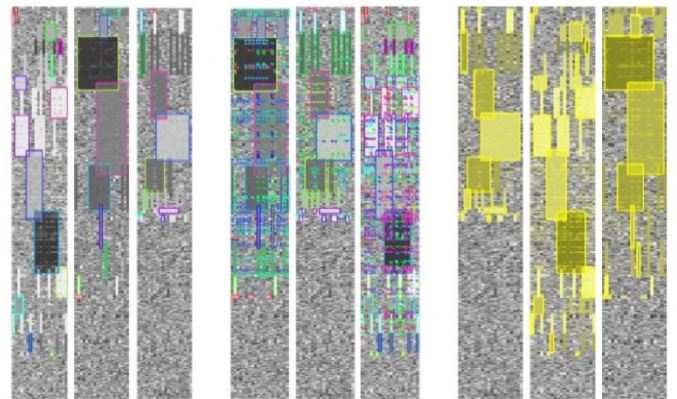

Figure 6: Bicluster results visualization. Each bicluster is depicted by its main rectangle (on the left). All biclusters are shown (in the middle). Representation with emphasized biclusters (on the right) [51]

Despite being the most common technique for visualizing single biclusters, heatmaps have limitations in terms of geometry especially when displaying biclusters with high levels of overlap.

3.2. Parallel coordinates visualization

Parallel coordinates are employed as a visualization method for representing complex, high-dimensional data sets. In this technique, each dimension is associated with a vertical line and individual data points are connected across these lines to form a *polyline* that reflects their multi-dimensional values. This approach has been adapted for the visualization of gene expression data as well. To depict gene profiles within an m-dimensional framework, *m* parallel and equidistant vertical lines are drawn, each symbolizing a different experimental condition. The gene profiles are then plotted as polylines across these lines, with the position of each point on a line corresponding to the gene's expression level under that particular condition. An example of this visualization technique, using parallel coordinates, is provided in Figure 7.
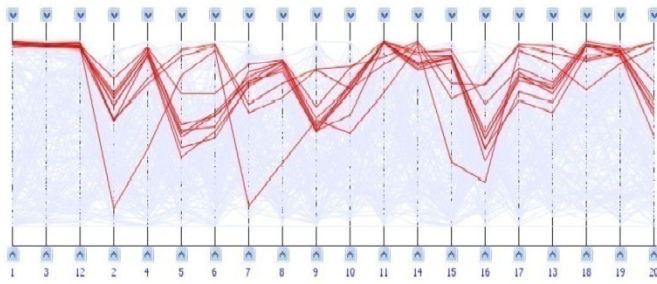
Figure 7: Parallel coordinates visualization. Polylines of significant genes highlighted in red [45]

In [51], the author implemented a series of transformations to their heatmap data representation in order to facilitate the simultaneous visualization of multiple biclusters using parallel coordinates. In this adaptation, the matrix of rows is represented as lines within the parallel coordinates framework. The vertical axes are positioned to correspond with the columns from the heatmap. To depict the conditions associated with a bicluster, the method computes the *mean vertical location* of all lines within a bicluster, establishing reference points known as *centroids*. These lines are then adjusted to intersect at the centroids. Next, the biclusters are rendered in a semi-transparent black hue. The color scheme utilized for the heatmap biclusters is replicated in the parallel coordinates display. To reduce visual clutter, lines not part of the highlighted biclusters can be dimmed by the user. An illustration of this parallel coordinates visualization for two biclusters is presented in Figure 8.
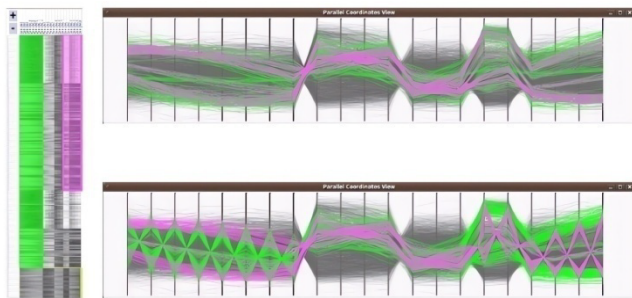

Figure 8: Two biclusters visualized as heatmaps (on the left) and next mapped to the parallel coordinates visualization without centroids (top right) then with centroids (bottom right) [51]

Parallel coordinates are a suitable method for visualizing large biclusters or individual biclusters. However, the cluttering of polylines due to overlapped biclusters can hinder the effectiveness of this technique in displaying multiple biclusters in a single view.

In general, scalability is the primary limitation of these methods (i.e., heatmaps and parallel coordinates), whether due to the abundance of biclusters or to high rates of overlap [45].

3.3. Bubble map visualization

In [45] and [52], the author presented an approach that involves the depiction of biclusters as circular entities named bubbles. The color coding of these bubbles corresponds to the sets of biclusters generated through a biclustering algorithm, with the capability to display up to three sets simultaneously. The intensity of the color, or brightness, indicates the uniformity within a bicluster. The size of each bubble is determined by the product of the number of genes and the number of conditions that constitute the bicluster. The placement of each bubble is based on a two-dimensional projection derived from multidimensional points which are the rows and columns that make up the bicluster. While this visualization method intuitively represents the arrangement of biclusters, the overlapping of bubbles does not precisely mirror the actual overlaps between biclusters. Rather, it serves as an approximation of their similarity. This technique is often employed to complement other methods, aiding in the comprehension of the general patterns observed in biclustering analyses. An illustration of this visualization method is provided in Figure 9.
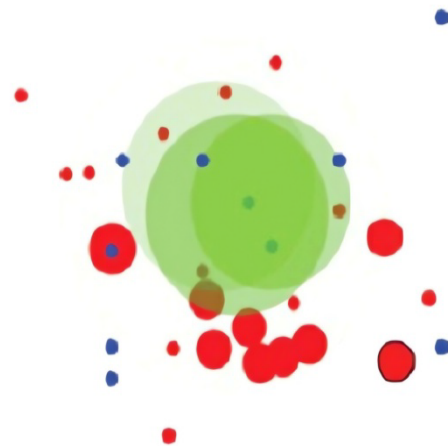

Figure 9: Bubblemap visualization of the outcomes of three biclustering algorithms [19]

Due to the limitations of heatmaps and parallel coordinates in visualizing a large number of biclusters, particularly with high levels of overlap [45], more advanced visualization approaches have been introduced. These approaches combine traditional gene expression visualization techniques (such as heatmaps and/or parallel coordinates) with set visualization techniques [53] like Venn-like diagrams [45], node-link diagrams [54] and two-dimensional matrix representations [55][56]. The following provides a description of these innovative techniques.

3.4. Venn-like diagrams visualization

Euler and Venn diagrams stand as some of the earliest techniques for illustrating sets and their interconnections. These diagrams were conceptualized by the British mathematician and philosopher, John Venn in the 18th century and have been widely adopted as effective tools for teaching concepts of set theory and logical relationships in education [57]. Utilizing a *proportional-*

area model, where the depicted areas correspond to the magnitude of a set and its intersections, sets are symbolized by enclosed *shapes* on a plane, typically *circles* and the relationships between sets are demonstrated through the *overlapping* of these shapes. They offer a versatile means to represent all conceivable set relationships, including intersection, inclusion and exclusion, due to the absence of limitations on the representation of overlaps. Venn diagrams, which are a specialized variant of Euler diagrams, are capable of representing every conceivable set intersection, regardless of whether they are non-empty or not.

In [45], the author described a novel visualization method based on Venn diagrams was employed, where biclusters are visualized as non-uniform shapes termed as *hulls* and their intersections are indicated by the *hulls' overlaps.* To represent genes and conditions that are unique to a single bicluster or shared among certain intersections, symbols known as *glyphs* are used. Each glyph is designed as a *pie chart*, segmented into sectors that represent the count of biclusters associated with the genes and conditions. The dimension of a glyph is indicative of the size of its respective group. The graphical representation is organized using a *force-directed algorithm* where biclusters are illustrated as flexible dynamic groups of genes and conditions. Specific genes and conditions related to a single bicluster or overlaps between a set of biclusters are illustrated through heatmaps and/or parallel coordinates which are displayed separately under request. While this approach effectively identifies a considerable number of biclusters with minimal rate of overlaps, its performance may degrade when faced with datasets containing a high degree of biclusters overlap. Figure 10 shows an illustration of this visualization technique.

3.5. Node-link diagrams visualization

A node-link diagram is a graphical representation, either in 2D or 3D consisting of *nodes* and connecting *edges*. This visualization technique represents various entities as nodes, also known as *vertices* and the connections between these entities as *edges* or *links*. Typically, nodes are symbolized by geometric shapes such as *circles* while the connections are depicted by *lines*. Creating a clear and informative graph requires careful consideration of nodes placement and edges routing, particularly when dealing with a large number of elements. Force-directed layout is commonly employed to address this challenge.

In [54], the author depicted biclusters and their intersections through a node-link *graph*. Here, biclusters form the nodes while the shared genes and conditions among them are represented as *edges* or *bands*. Each bicluster is visualized as a heatmap matrix where rows correspond to genes and columns represent conditions. Overlaps between biclusters are represented by

connecting *bands* that link the corresponding heatmaps at the positions of shared genes and conditions. The thickness of these bands indicates the degree of overlap, with thicker bands signifying more shared elements. The graph layout employs a *force-directed algorithm* where biclusters with overlapping elements are drawn closer together. When a bicluster is selected, it reveals detailed information such as its designation or the identifiers of its genes and conditions. This visualization approach is highly interactive and straightforward since the design is based on the heatmaps visualization. The bands that indicate overlap offer the user a detailed view of the common genes and conditions found in each pair of biclusters. However, this method's visualization of overlaps on a one-to-one basis makes it challenging to distinguish multiple bicluster overlaps easily. Moreover, the technique's scalability is limited with an increase in overlap levels. In fact, the bands become overly congested, making it difficult to gain a comprehensive view of the biclustering outcomes. An example of this bicluster visualization method is presented in Figure 11.
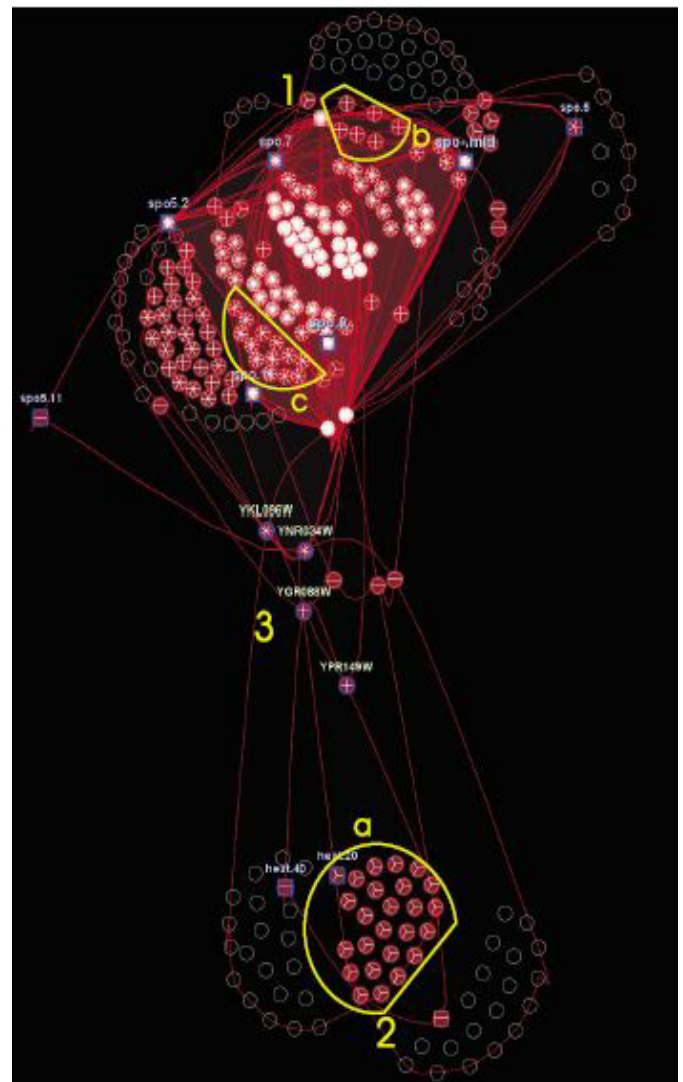

Figure 10: 50 biclusters visualization. Three sets of biclusters and their intersections are easily identifiable using the hulls representation (groups 1, 2 and 3) [45]
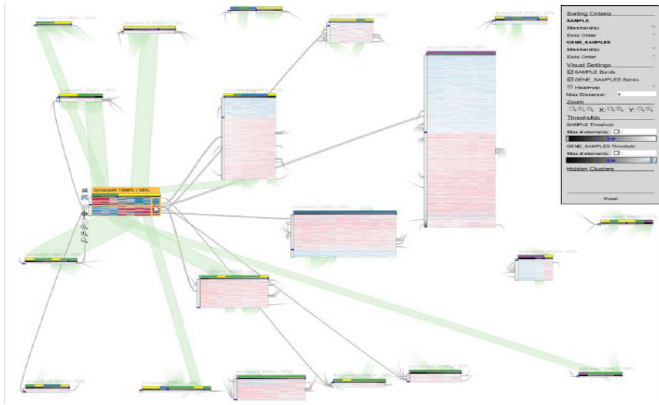
Figure 11: 20 biclusters are depicted. Nodes represent the biclusters displayed as heatmaps and edges connect the corresponding heatmaps at the positions of shared rows and columns [54]

3.6. Two-dimensional matrix visualization

In [55], [56] the author proposed a visualization method that takes into account the special characteristics of biclustering which are overlaps and bi-dimensionality. The primary objectives of the method are:

- Visually represent biclusters of varying sizes and degrees of overlap.
- Maintain both elements (i.e., genes and/or conditions) and biclusters information within a single view, preventing context loss.
- Avoid information simplification or duplication. While alternative approaches might offer clearer visualizations, they often ignore interesting information or introduce ambiguities.
- Provide interactive features that enable diverse perspectives and facilitate exploratory analysis.
- Increase scalability. An effective bicluster visualization method should accommodate large datasets, numerous biclusters and extensive overlaps between biclusters.

To achieve these objectives, the authors developed a visualization technique that lay out the generated biclusters as a *two-dimensional matrix*. Each bicluster is represented as a *column* and overlaps between sets of biclusters are depicted as *rows*. This method combines a modified set visualization technique for matrix layout with a traditional heatmap approach for visualizing individual biclusters and their overlaps as gene expression matrices [58], [53]. A user interface is implemented to query the biclusters intersection matrix and visualize matching results. The proposed technique is implemented in a web-based interactive visualization tool called *VisBicluster* which supports features like sorting, zooming and on-demand details. While applicable to any type of overlapping groups, the primary focus of this technique is on representing biclusters derived from gene expression data.

This approach emphasizes overlaps, making their identification and selection straightforward within the defined matrix-based visualization. By avoiding element crossings (lines, shapes, etc.), this method minimizes visual clutters. VisBicluster's scalability is remarkable since it can efficiently display large numbers of highly overlapped biclusters simultaneously. The tool also incorporates linking and brushing techniques for inspecting selected data subsets from different perspectives. Figure 12 illustrates this visualization technique.
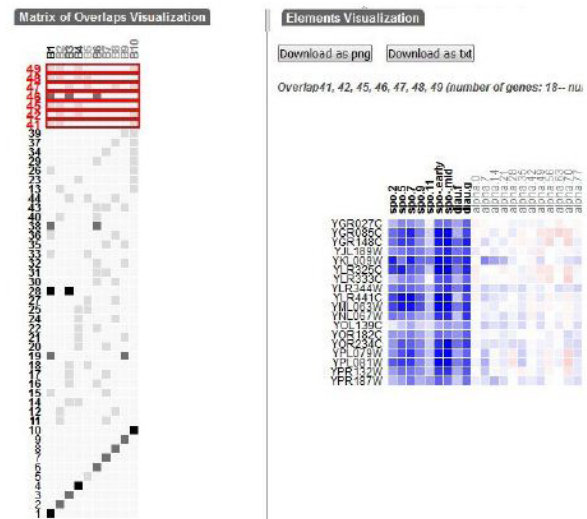

Figure 12: Plaid model biclustering algorithm [40] result visualization for yeast *Saccharomyces cerevisiae* expression data [56]. Overlaps are visualized as a two-dimensional matrix which is clustered by *Levenshtein distance*[59], an algorithm that arranges the most similar bicluster overlaps near each other (on the left). Genes and conditions of a set of similar selected overlaps are represented as a heatmap (on the right).

4. Critique of visualization methods

Our evaluation of the surveyed techniques focused on three key aspects:

- Reducing *overlaps* among biclusters.
- Maximizing the number of biclusters visualized within a single view (i.e., *scalability*).
- Ensuring *clear visibility* of both biclusters and their overlapping regions.

Due to their geometric limitations, heatmaps and parallel coordinates often struggle to efficiently visualize biclusters, particularly when evaluated against the criteria of overlap minimization and scalability [45]. Heatmaps, in particular, are typically unbalanced in terms of dimensions, with many more genes (around $10^{[3-4]}$ rows) than conditions (around $10^{[1-2]}$ columns) [19]. So, replication techniques used to visualize biclusters can lead to large matrices when visualizing multiple biclusters, making overlap perception difficult and limiting scalability. Additionally, the common use of a green-black-red color scale in heatmaps can hinder human perception of expression levels [7], [19].

Parallel coordinates often suffer from cluttering due to overlapping polylines when visualizing multiple biclusters simultaneously. This makes it difficult to perceive overlaps and limits scalability. Individual biclusters can be easily interpreted due to the human brain's ability to recognize patterns like parallel lines, mirror effects and changes in slope [19]. However, visualizing several large biclusters with high rates of overlap in the same parallel coordinates can be in some cases impossible.

By combining heatmaps and/or parallel coordinates with more sophisticated sets visualization techniques like Venn diagrams [45], node-link diagrams [54] or two-dimensional matrix visualization [55], [56], we can confirm that the scalability and clarity of drawn biclusters are improved significantly. Representing biclusters and their overlaps as abstract elements such as hulls [45], bands between heatmaps [54] or cells in a matrix [56] can simplify the visualization. So, focusing on intersections between visualized elements (i.e., biclusters) in a global overview while providing details (i.e., gene expression levels) in a separate view as heatmaps or parallel coordinates, will alleviate the representation remarkably.

Overlap is a key aspect when visualizing biclusters and overlap-centered tasks like how many overlaps between a specific number of biclusters or what are the biclusters involved in a certain overlap, can help analysts to gain insights from visualized complex data. Visualization tools that help interpret complex analysis results without distorting or losing the original data context are crucial for understanding data. In fact, VisBicluster offers an overlap-centered solution that aids in understanding biclustering results, providing scalability for analyzing real gene expression data [55], [56].

While these three novel visualization methods (i.e., BicOverlapper, Furby and VisBicluster) offer high rate of scalability, they can still be overwhelmed by a large number of biclusters and overlaps. So, in some cases, it may be impossible to visualize all biclusters within a single view. Table 2 provides a summary of biclustering visualization techniques, outlining their key characteristics.

Table 2: Evaluation of biclustering visualization techniques visualization techniques

Biclustering visualization technique	Visualization methods	Dealing with overlaps	Scalability	Clarity of visualization	Time complexity
Heatmap in [46]	Heatmap	Reordering of rows and columns	Low	Low	$O(\|X\|^2\|V\|^2) + O(\|X\|\|V\|)O(MinLA)^*$ $O(\|X\|^2\|V\|^3) + O(\|X\|\|V\|)O(MinLA)^{**}$ $O(\|X\|^2\|V\|^{2d}) + O(\|X\|\|V\|)O(MinLA)^{***}$
Heatmap in [47]	Heatmap	Reordering of rows and columns	Low	Low	$O(m^\alpha)$ with $\alpha \in [1.6, 2]^+$ $O(n^\alpha)$ with $\alpha \in [2.5, 2.7]^+$
Heatmap in [48]	Heatmap	Replication of rows and columns	Low	Low	$O(mn^2 + n^2 \log n)^\#$
Heatmap in [51]	Heatmap	Replication of rows and columns	Low	Low	−
Parallel coordinates plots [51]	Parallel coordinates plots	Color polylines	Low	Low	−
Bubblemap[45] and [52]	circles	Intersections between circles	Medium	High	−
Venn diagrams [45]	Hulls, pie charts, Heatmap, parallel coordinates	Hulls intersections, Glyphs with pie chart sectors	High	High	$O(n^3)^\$$

Node-link diagrams [54]	Heatmap, bands	Bands between rows and columns of heatmaps	High	High	_
Two-dimensional matrix visualization [55] [56]	Heatmap, two-dimensional matrix	Cells of the two-dimensional matrix	High	High	$O(n*m)^{\&}$

Where:

V is the set of vertices and X is the set of hyperedges.

*If we convert hyperedges into paths or cycles.

**If we convert each hyperedge into maximum valid cycles.

***If we convert each hyperedge into d arbitrary valid cycles.

$O(MinLA)$ is the time complexity of the minimum linear arrangement algorithm.

+m is the number of biclusters and n is the number of elements.

#n is the number of biclusters and m is the number of rows and columns in all the biclusters.

n isthe number of nodes (genes and conditions).

&n is the number of biclusters and m is the total number of overlaps between biclusters.

5. Tools and datasets

5.1. Tools

Several useful tools exist that incorporate many of the biclustering visualization methods discussed in this review. Table 3 provides a summary of the key features of these tools.

- BiVoc: It is a C++ implementation that incorporates two primary programs: one for the layout algorithm and another for generating the corresponding visualization image. To address the potential issue of numerous rows and columns due to replication, a user-friendly web interface allows for the selection of specific biclusters to display [48].
- ExpressionView: It is an R package that incorporates the developed ordering method and provides interactive visualization of bicluster results as heatmaps in a Flash applet format [47].
- Bicluster Viewer: It offers heatmap and parallel coordinates visualizations for representing biclusters as contiguous blocks. It also includes a range of interactive features [51].
- Biclust: It incorporates various biclustering algorithms and offers several visualization methods, including the Bubbleplot graphical representation that depicts biclusters as circles [52].
- BicOverlapper: It is a Java package that enables the visualization of bicluster sets using Venn-like diagrams, the representation of microarray data matrices or individual biclusters as heatmaps and/or parallel coordinates and the visualization of transcription regulatory networks. It also supports the integration of these various visualization techniques for comprehensive data analysis [45].
- Furby: It is a Java implementation of the node-link diagram technique for visualizing biclusters[54]. It incorporates several features including ordering, zooming, adjustable thresholds for different defined values, etc.
- VisBicluster: It is a web application built using JavaScript and the D3 library[60]. It enables the visualization of biclusters and their potential overlaps using the two-dimensional matrix method. Single biclusters or overlaps between two or more biclusters are depicted by heatmaps in a separate view. The software includes various integrated features such as ordering, filtering, zooming, etc. Linking and brushing between visualization techniques in VisBicluter are supported [55], [56].

Table 3: Characteristics of biclustering visualization tools

Tool	Heatmap	Parallel coordinates	Other visualizations	Degree of interactivity	accessibility	Available at
BiVoc [48]	Yes	No	_	Medium	free	http://bioinformatics.cs.vt.edu/~murali/papers/bivoc
ExpressionView [47]	Yes	No	_	Medium	free	http://www.unil.ch/cbg/ExpressionView

Bicluster Viewer [61]	Yes	Yes	_	High	commercial	http://www.simtech.uni-stuttgart.de
Biclust [52]	Yes	Yes	Bubbleplot, beplot, boxplot	Low	free	https://cran.r-project.org/web/packages/biclust/index.html
BicOverlapper [45]	Yes	Yes	Venn-like diagrams, TRN, word cloud	Very high	free	http://vis.usal.es/bicoverlapper
Furby [54]	Yes	No	Bar chart, histogram	Very high	free	http://furby.caleydo.org
VisBicluter [55] [56]	Yes	No	Two-diensional matrix, cells	Very high	free	http://vis.usal.es/~visusal/visbicluster

Table 4: Gene expression datasets utilized to evaluate biclustering visualization techniques

Name	Genes	Experimental conditions	Reference
Yeast *Saccharomyces cerevisiae* microarray data	2467	79	[1]
Prostate cancer tissue dataset	54675	19	[62]
Human lung carcinomas dataset	12600	203	[63]
Multiple tissue types dataset	5565	102	[64]

5.2. Datasets

Multiple gene expression datasets were employed to assess biclustering visualization techniques. Table 4 provides a list of some of these datasets.

6. Conclusion

While clustering focuses on identifying groups of similar elements within a dataset (gene expression data) by applying algorithms on one dimension either rows (i.e. genes) or columns (i.e. conditions), biclustering seeks to uncover patterns that exist simultaneously across both rows and columns. This added complexity (for biclustering case) necessitates more advanced visualization methods to effectively analyze gene expression data. By providing insights into the underlying relationships between genes and conditions, these visualizations can help bioinformaticians extract valuable knowledge. We present in this paper a global review that summarizes the most mentioned techniques to visualize results of biclustering of gene expression data in the literature and we evaluate them. We can mention that visualization issues such as scalability and overlaps between biclusters can be considered as open directions for academicians and researchers. As a possible solution to simplify the complexity of visualization of biclustering results, we think that the combination between traditional visualization techniques like heatmaps or parallel coordinates and one of the novel set visualization techniques mentioned in the literature [53], can be useful [56].

Conflict of Interest

The authors declare no conflict of interest.

References

[1] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns," *Proceedings of the National Academy of Sciences*, vol. 95, no. 25, 1998, doi.org/10.1073/pnas.95.25.1486

[2] R.. Sokal, C.. Michener, "A statistical method for evaluating systematic relationships," *Univ. Kansas, Sci. Bull.*, vol. 38, , pp. 1409–1438, 1958.

[3] J.A. Hartigan, M.A. Wong, Algorithm AS 136: A K-Means

Clustering Algorithm, 1979, doi.org/10.2307/2346830

[4] Y. Cheng, G.M. Church, "Biclustering of expression data," *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, vol. 8, pp. 93–103, 2000.

[5] S.C. Madeira, A.L. Oliveira, "Biclustering algorithms for biological data analysis: a survey," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 24–45, 2004, doi:10.1109/TCBB.2004.2.

[6] B. Pontes, R. Giráldez, J.S. Aguilar-Ruiz, "Biclustering on expression data: A review," *Journal of Biomedical Informatics*, vol. 57, pp. 163–180, 2015, doi:10.1016/j.jbi.2015.06.028.

[7] C. Ware, Information visualization : perception for design, Morgan Kaufman, 2004.

[8] B.J. Fry, Computational information design, Massachusetts Institute of Technology Cambridge, MA, USA, 2004.

[9] J.J. Thomas, K.A. Cook, Illuminating the path, IEEE Computer Society, 2005.

[10] D. Keim, K. Jörn, G. Ellis, M. Florian, Mastering the information age : solving problems with visual analytics, Eurographics Association, 2010.

[11] A. Holzinger, Human-Computer Interaction and Knowledge Discovery (HCI-KDD): What Is the Benefit of Bringing Those Two Fields to Work Together?, Springer, Berlin, Heidelberg, vol.8127, pp 319–328, 2013, doi:10.1007/978-3-642-40511-2_22.

[12] W. Ayadi, M. Elloumi, Biological Knowledge Visualization, John Wiley & Sons, Inc., Hoboken, New Jersey: 651–661, 2011.

[13] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985, doi:10.1007/BF01898350.

[14] D. Gonçalves, R.S. Costa, R. Henriques, "Context-situated visualization of biclusters to aid decisions: going beyond subspaces with parallel coordinates," *ACM International Conference Proceeding Series*, no. 9, pp. 1–5, doi:10.1145/3531073.3531124.

[15] N.K. Verma, T. Sharma, S. Dixit, P. Agrawal, S. Sengupta, V. Singh, "BIDEAL: A Toolbox for Bicluster Analysis—Generation, Visualization and Validation," *SN Computer Science*, vol. 2, no. 1, 2021, doi:10.1007/S42979-020-00411-9.

[16] M. Sözdinler, "A Review of Visualization Methods and Tools for the Biclustering," *International Journal of Innovative Science and Research Technology*, vol. 6, 2021, doi.org/10.48550/arXiv.2111.12154.

[17] H. Aouabed, R. Santamaria, M. Elloumi, "Visualizing biclustering results on gene expression data: A survey," *ACM International Conference Proceeding Series*, pp. 170–179, 2021, doi:10.1145/3473258.3473284.

[18] H. Aouabed, M. Elloumi, R. Santamaría, "An evaluation study of biclusters visualization techniques of gene expression data," *Journal of Integrative Bioinformatics*, vol. 18, no. 4, 2021, doi:10.1515/JIB-2021-0019/MACHINEREADABLECITATION/RIS.

[19] R. Santamaria, Visual analysis of gene expression data by means of biclustering, University of Salamanca, Spain, 2009.

[20] A.V. Freitas, W. Ayadi, M. Elloumi, J. Oliveira, J. Oliveira, J.-K. Hao, Survey on Biclustering of Gene Expression Data, John Wiley & Sons, Inc., Hoboken, New Jersey: 591–608, 2012, doi:10.1002/9781118617151.ch25.

[21] H. Ben Saber, M. Elloumi, "Dna Microarray Data Analysis: a New Survey on Biclustering," *International Journal for Computational Biology*, vol. 4, no. 1, pp. 21, 2015, doi:10.34040/ijcb.4.1.2014.36.

[22] S.C. Madeira, A.L. Oliveira, "Biclustering algorithms for biological data analysis: a survey," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 24–45,

2004.

[23] V.A. Padilha, R.J.G.B. Campello, "A systematic comparative evaluation of biclustering techniques," *Padilha Campello BMC Bioinforma.*, vol. 18, , 2017, doi:10.1186/s12859-017-1487-1.

[24] G. Getz, E. Levine, E. Domany, "Coupled two-way clustering analysis of gene microarray data.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 97, no. 22, pp. 12079–84, 2000, doi:10.1073/pnas.210134797.

[25] C. Tang, L. Zhang, A. Zhang, M. Ramanathan, "Interrelated two-way clustering: an unsupervised approach for gene expression data analysis," in *Proceedings 2nd Annual IEEE International Symposium on Bioinformatics and Bioengineering (BIBE 2001)*, IEEE: 41–48, 2001, doi:10.1109/BIBE.2001.974410.

[26] S. Busygin, G. Jacobsen, E. Krämer, "Double Conjugated Clustering Applied to Leukemia Microarray Data," *IN 2ND SIAM ICDM, WORKSHOP ON CLUSTERING HIGH DIMENSIONAL DATA*, 2002.

[27] J.A. Hartigan, "Direct Clustering of a Data Matrix," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 123, 1972, doi:10.2307/2284710.

[28] A. Prelić, S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, E. Zitzler, "A systematic comparison and evaluation of biclustering methods for gene expression data," *Bioinformatics*, vol. 22, no. 9, pp. 1122–1129, 2006, doi:10.1093/bioinformatics/btl060.

[29] A. Ben-Dor, B. Chor, R. Karp, Z. Yakhini, "Discovering local structure in gene expression data," *Proceedings of the Sixth Annual International Conference on Computational Biology - RECOMB '02*, pp. 49–57, 2002, doi:10.1145/565196.565203.

[30] T.M. Murali, S. Kasif, "Extracting conserved gene expression motifs from gene expression data.," *Pacific Symposium on Biocomputing.*, vol. 88, , pp. 77–88, 2003, doi:10.1142/9789812776303_0008.

[31] S. Bergmann, J. Ihmels, N. Barkai, "Iterative signature algorithm for the analysis of large-scale gene expression data," *Physical Review E*, vol. 67, no. 3 1, pp. 031902/1-031902/18, 2003, doi:10.1103/PhysRevE.67.031902.

[32] H. Cho, I.S. Dhillon, Y. Guan, S. Sra, "Minimum Sum-Squared Residue Co-clustering of Gene Expression Data," in *Proceedings of the 2004 SIAM International Conference on Data Mining*, Society for Industrial and Applied Mathematics, Philadelphia, PA: 114–125, 2004, doi:10.1137/1.9781611972740.11.

[33] G. Li, Q. Ma, H. Tang, A.H. Paterson, Y. Xu, "QUBIC: A qualitative biclustering algorithm for analyses of gene expression data," *Nucleic Acids Research*, vol. 37, no. 15, 2009, doi:10.1093/nar/gkp491.

[34] C. Huttenhower, K. Tsheko Mutungu, N. Indik, W. Yang, M. Schroeder, J.J. Forman, O.G. Troyanskaya, H.A. Coller, "Detailing regulatory networks through large scale data integration," *Bioinformatics*, vol. 25, no. 24, pp. 3267–3274, 2009, doi:10.1093/bioinformatics/btp588.

[35] D. Bozdag, J.D. Parvin, U. V Catalyurek, "A biclustering method to discover co-regulated genes using diverse gene expression datasets," *International Conference on Bioinformatics and Computational Biology*, vol. 5462 LNBI, , pp. 151–163, 2009, doi:10.1007/978-3-642-00727-9_16.

[36] A.A. Shabalin, V.J. Weigman, C.M. Perou, A.B. Nobel, "Finding large average submatrices in high dimensional data," *The Annals of Applied Statistics*, vol. 3, no. 3, pp. 985–1012, 2009, doi:10.1214/09-AOAS239.

[37] A. Tanay, R. Sharan, R. Shamir, "Discovering statistically significant biclusters in gene expression data.," *Bioinformatics*, vol. 18 Suppl 1, , pp. S136-S144, 2002, doi:10.1093/bioinformatics/18.suppl_1.S136.

[38] D.S. Rodriguez-Baena, A.J. Perez-Pulido, J.S. Aguilar-Ruiz, "A biclustering algorithm for extracting bit-patterns from binary datasets," *Bioinformatics*, vol. 27, no. 19, pp. 2738–2745, 2011, doi:10.1093/bioinformatics/btr464.

[39] A. Serin, M. Vingron, "DeBi: Discovering Differentially Expressed Biclusters using a Frequent Itemset Approach.," *Algorithms Molocular Biology*, vol. 6, no. 1, pp. 18, 2011, doi:10.1186/1748-7188-6-18.

[40] L. Lazzeroni, A. Owen, "Plaid Models for Gene Expression Data," *CEUR Workshop Proc.*, vol. 1542, , pp. 33–36, 2000, doi:10.1017/CBO9781107415324.004.

[41] Y. Kluger, R. Basri, J.T. Chang, M. Gerstein, "Spectral Biclustering of Microarray Data: Coclustering Genes and Conditions," *Genome Research*, vol. 13, pp. 703–716, 2003, doi:10.1101/gr.648603.graph.

[42] J. Gu, J.S. Liu, "Bayesian biclustering of gene expression data," *BMC Genomics*, vol. 9 Suppl 1, pp. S4, 2008, doi:10.1186/1471-2164-9-S1-S4.

[43] S. Hochreiter, U. Bodenhofer, M. Heusel, A. Mayr, A. Mitterecker, A. Kasim, T. Khamiakova, S. van Sanden, D. Lin, W. Talloen, L. Bijnens, H.W.H. Göhlmann, Z. Shkedy, D.A. Clevert, "FABIA: Factor analysis for bicluster acquisition," *Bioinformatics*, vol. 26, no. 12, pp. 1520–1527, 2010, doi:10.1093/bioinformatics/btq227.

[44] S. Barkow, S. Bleuler, A. Prelić, P. Zimmermann, E. Zitzler, "BicAT: A biclustering analysis toolbox," *Bioinformatics*, vol. 22, no. 10, pp. 1282–1283, 2006, doi:10.1093/bioinformatics/btl099.

[45] R. Santamaría, R. Therón, L. Quintales, "A visual analytics approach for understanding biclustering results from microarray data," *BMC Bioinformatics*, vol. 9, no. 1, pp. 247, 2008, doi:10.1186/1471-2105-9-247.

[46] R. Jin, Y. Xiang, D. Fuhry, F.F. Dragan, "Overlapping Matrix Pattern Visualization: A Hypergraph Approach," in *2008 Eighth IEEE International Conference on Data Mining*, IEEE: 313–322, 2008, doi:10.1109/ICDM.2008.102.

[47] A. Luscher, G. Csardi, A. Morton de Lachapelle, Z. Kutalik, B. Peter, S. Bergmann, "ExpressionView--an interactive viewer for modules identified in gene expression data," *Bioinformatics*, vol. 26, no. 16, pp. 2062–2063, 2010, doi:10.1093/bioinformatics/btq334.

[48] G.A. Grothaus, A. Mufti, T. Murali, "Automatic layout and visualization of biclusters," *Algorithms for Molecular Biology*, vol. 1, no. 1, pp. 15, 2006, doi:10.1186/1748-7188-1-15.

[49] S. Batzoglou, S. Istrail, Physical Mapping with Repeated Probes: The Hypergraph Superstring Problem, Springer, Berlin, Heidelberg: 66–77, 1999, doi:10.1007/3-540-48452-3_5.

[50] K.S. Booth, G.S. Lueker, "Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms," *Journal of Computer and System Sciences*, vol. 13, no. 3, pp. 335–379, 1976, doi:10.1016/S0022-0000(76)80045-1.

[51] J. Heinrich, R. Seifert, M. Burch, D. Weiskopf, BiCluster Viewer: A Visualization Tool for Analyzing Gene Expression Data, Springer, Berlin, Heidelberg: 641–652, 2011, doi:10.1007/978-3-642-24028-7_59.

[52] S. Kaiser, R. Santamaria, T. Khamiakova, M. Sill, R. Theron, L. Quintales, F. Leisch, E. De, T. Maintainer, "biclust: BiCluster Algorithms. R package version 1.0.2," 2013.

[53] H. Aouabed, R. Santamaría, M. Elloumi, Suitable Overlapping Set Visualization Techniques and Their Application to Visualize Biclustering Results on Gene Expression Data, Springer, Cham: 191–201, 2018, doi:10.1007/978-3-319-99133-7_16.

[54] M. Streit, S. Gratzl, M. Gillhofer, A. Mayr, A. Mitterecker, S. Hochreiter, "Furby: fuzzy force-directed bicluster visualization.," *BMC Bioinformatics*, vol. 15 Suppl 6, no. Suppl 6, pp. S4, 2014, doi:10.1186/1471-2105-15-S6-S4.

[55] H. Aouabed, R. Santamaria, M. Elloumi, "VisBicluster: A Matrix-Based Bicluster Visualization of Expression Data," *Journal of Computational Biology*, pp. cmb.2019.0385, 2020, doi:10.1089/cmb.2019.0385.

[56] H. Aouabed, M. Elloumi, "Visualizing Biclusters of Gene Expression Data and Their Overlaps Based on a Two-Dimensional Matrix Technique," *Computational Biology and Bioinformatics 2023*, vol. 11, no. 2, pp. 19–32, 2023, doi:10.11648/J.CBB.20231102.11.

[57] M.E. Baron, "A Note on the Historical Development of Logic Diagrams: Leibniz, Euler and Venn," *The Mathematical Gazette*, vol. 53, no. 384, pp. 113, 1969, doi:10.2307/3614533.

[58] A. Lex, N. Gehlenborg, H. Strobelt, R. Vuillemot, H. Pfister, "UpSet: Visualization of intersecting sets," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1983–1992, 2014, doi:10.1109/TVCG.2014.2346248.

[59] V.I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals," *Sov. Phys. Dokl. Vol. 10, p.707*, vol. 10, , pp. 707, 1966.

[60] M. Bostock, V. Ogievetsky, J. Heer, "D³ Data-Driven Documents," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 2301–2309, 2011, doi:10.1109/TVCG.2011.185.

[61] J. Heinrich, M. Burch, R. Seifert, D. Weiskopf, "BiCluster Viewer : A Visualization Tool for Analyzing Gene Expression Data BiCluster Viewer : A Visualization Tool for Analyzing Gene Expression Data," 2011.

[62] S. Varambally, J. Yu, B. Laxman, D.R. Rhodes, R. Mehra, S.A. Tomlins, R.B. Shah, U. Chandran, F.A. Monzon, M.J. Becich, J.T. Wei, K.J. Pienta, D. Ghosh, M.A. Rubin, A.M. Chinnaiyan, "Integrative genomic and proteomic analysis of prostate cancer reveals signatures of metastatic progression," *Cancer Cell*, vol. 8, no. 5, pp. 393–406, 2005, doi:10.1016/j.ccr.2005.10.001.

[63] A. Bhattacharjee, W.G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E.J. Mark, E.S. Lander, W. Wong, B.E. Johnson, T.R. Golub, D.J. Sugarbaker, M. Meyerson, "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses," *Proceedings of the National Acad Sciences U. S. A.*, vol. 98, no. 24, 13790–13795, 2001, doi:10.1073/pnas.191502998.

[64] A.I. Su, M.P. Cooke, K.A. Ching, Y. Hakak, J.R. Walker, T. Wiltshire, A.P. Orth, R.G. Vega, L.M. Sapinoso, A. Moqrich, A. Patapoutian, G.M. Hampton, P.G. Schultz, J.B. Hogenesch, "Large-scale analysis of the human and mouse transcriptomes," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 7, 4465–4470, 2002, doi:10.1073/pnas.012025199.

HAITHEM AOUABED received an Undergraduate Degree in Mathematics in 2006, and a Master's Degree in Computer Science in 2010, from the Faculty of Economic Sciences and Management, Sfax, Tunisia. He also received a Master's Degree in Computer Science in 2012 and a PhD Degree in Computer Science in 2024, from the University of Sfax, Tunisia. His research interests include Bioinformatics and Information Visualization, especially the integration of different source

data, analysis algorithms and representations for a better understanding of biological problems.

MOURAD ELLOUMI received an Undergraduate Degree in Mathematics and Physics in 1984, and a master's degree in computer engineering in 1988, from the Faculty of Sciences of Tunis, Tunisia. He also received a Master's Degree in Computer Science in 1989 and a PhD Degree in Computer Science in 1994, from the University of Aix-Marseilles III, France. Then, he received a *Habilitation* for conducting research in Computer Science in 2003, from the National School of Computer Science, Tunis, Tunisia. He is currently a Full Professor in Computer Science, College of Computing and Information Technology, University of Bisha, Saudi Arabia. Professor Mourad Elloumi is the author/co-author of more than 80 publications in international journals, books and conference proceedings. He was a Guest Editor of a special issue on biological knowledge discovery and data mining, *Knowledge Based Systems* Journal (Elsevier 2002), a Guest Editor of a special issue on pattern finding in Computational Molecular Biology, *Recent Patents on DNA and Gene Sequence* Journal (Bentham Science 2012), a Co-Editor of the proceedings of two international conferences and Editor/Co-Editor of five books, respectively, on Algorithms in Computational Molecular Biology (Wiley 2011), Biological Knowledge Discovery (Wiley 2014), Pattern Recognition in Computational Molecular Biology (Wiley 2015), Algorithms for Next-Generation Sequencing Data (Springer 2017), Deep Learning for Biomedical Data Analysis (Springer 2021). His research interests include Algorithmics, Computational Molecular Biology, Knowledge Discovery and Data Mining, and Deep Learning.

FAHAD ALGARNI received a bachelor's degree (Hons.) from the Department of Computer Science, King Abdulaziz University, the M.I.T. degree in computer networks from La Trobe University, Melbourne, Australia, and the Ph.D. degree from the Clayton School of Information Technology, Monash University, Melbourne, Australia. He is currently the Dean of the College of Computing and Information Technology, University of Bisha, Saudi Arabia. His research interests include Wireless Sensor Networks; Cloud Computing, Systems, Design, and Reliability, the IoT, IoV, IoD, Cybersecurity and Bioinformatics.

JENRS

Enhancing Mental Health Support in Engineering Education with Machine Learning and Eye-Tracking

Yuexin Liu[*,1] ⬤, **Amir Tofighi Zavareh**[1], **Ben Zoghi**[2]

[1]Department of Engineering Technology and Industrial Distribution, Texas A&M University, College Station, 77843, USA

[2]Lyle School of Engineering, Southern Methodist University, Dallas, 75205, USA

[*]Corresponding author: Yuexin Liu, Email: Yuexin.liu@tamu.edu

ABSTRACT: Mental health concerns are increasingly prevalent among university students, particularly in engineering programs where academic demands are high. This study builds upon previous work aimed at improving mental health support for engineering students through the use of machine learning (ML) and eye-tracking technology. A framework was developed to monitor mental health by analyzing eye movements and physiological data to provide personalized support based on student behavior. In this extended study, baseline data were analyzed to explore the correlations between emotions and physiological biomarkers. Key findings indicate that emotions such as Anger and Fear are positively correlated with increased physical activity, while Sadness is associated with elevated respiratory rates. A strong positive correlation between Electrodermal Activity (EDA) and Happiness was also identified, indicating physiological markers linked to positive emotional states. Temporal patterns were observed, with heightened emotional tagging occurring more frequently in the evening. These findings deepen the understanding of how emotional states manifest through physiological changes, providing a foundation for enhancing real-time, personalized mental health interventions. The results contribute to a more comprehensive framework for supporting student well-being and academic performance within engineering education.

KEYWORDS: Mental Health, Philosophy of Engineering Education, Data Correlation, Factor Analysis, Machine Learning, Electrodermal Activity

1. Introduction

Mental health concerns have become increasingly prominent among university students, especially within engineering programs, where students are subject to intense academic demands and pressure [1, 2]. Traditionally, engineering education has focused on the development of technical knowledge and skills [3, 4], often placing limited emphasis on the physical and mental well-being of students. As a result, many students in engineering experience high levels of stress, which can negatively impact their academic performance, personal lives, and overall professional growth.

Current methods for assessing mental health, such as surveys and self-reported questionnaires, have limitations in terms of accuracy and bias. These methods may not fully capture the extent of students' mental health challenges, as some may feel reluctant to disclose personal issues due to stigma, fear of judgment, or concerns about potential academic consequences [5]. While initiatives like counseling services, awareness campaigns, and stress management workshops have been introduced to address these issues , there is a pressing need for more objective and effective methods to monitor and improve the mental health of engineering students [6].

Predictive analytics, particularly when applied through machine learning (ML) models, presents a promising solution for addressing mental health challenges. In healthcare,

ML techniques have shown potential to transform mental health assessment and intervention by providing objective insights into emotional and psychological states. Building on this, the conference paper Improving Mental Health Support in Engineering Education Using Machine Learning and Eye-Tracking introduced a framework leveraging ML and eye-tracking technology to monitor student well-being. This innovative approach aimed to fill gaps in mental health monitoring by providing data-driven insights and personalized interventions.

This study introduces insights based on data from a 10-week study involving 18 participants. It includes detailed correlations between emotional states—such as Anger, Fear, Happiness, and Sadness—and physiological markers, including physical activity (actigraphy), respiratory rate, and Electrodermal Activity (EDA). Results reveal that increased actigraphy counts are associated with Anger and Fear, while Sadness correlates with elevated respiratory rates. Furthermore, a strong positive relationship between EDA and Happiness is observed. These findings offer a deeper understanding of the physiological responses to emotional states, improving the system's ability to provide real-time, personalized mental health support.

This paper is structured as follows: the next section provides background information on machine learning algorithms and mental health prediction. The methodology section outlines the strategy used for data collection and analysis. The results section presents detailed findings on

the emotional and physiological correlations, followed by a discussion of the potential impact on future mental health interventions. The paper concludes by summarizing the findings and offering recommendations for future research directions.

2. Background

2.1. Machine Learning

Machine learning (ML) refers to the application of statistical and probabilistic methods to build systems that can learn and improve from experience [7, 8]. This capability makes ML a powerful tool for predicting mental health outcomes by analyzing large amounts of complex data, leading to the development of intelligent automated systems that can offer personalized insights. Several algorithms such as support vector machines (SVM), random forests, and artificial neural networks (ANNs) have proven effective in predicting future outcomes and categorizing data. In the healthcare sector, ML is widely applied in various forms, including supervised learning, unsupervised learning, and deep learning. There are also hybrid methods like semi-supervised learning, which combines elements of both supervised and unsupervised learning, as well as reinforcement learning [9].

Supervised learning (SL) is commonly used in healthcare for disease prediction, where the algorithms are trained on a dataset that is pre-labeled with known outcomes. These models are then tested on unseen data to evaluate their predictive performance. On the other hand, unsupervised learning (UL) works without labeled data and is designed to detect patterns and clusters in datasets. Techniques such as k-means clustering, hierarchical clustering, and principal component analysis (PCA) are commonly employed to identify meaningful patterns within the data. In healthcare, UL is valuable for uncovering hidden structures in medical imaging or genetic data, which can help identify subtypes of diseases and facilitate personalized treatment strategies. Additionally, UL is useful for anomaly detection and feature reduction, contributing to more accurate medical diagnoses.

2.2. Deep Learning

Deep learning (DL) [10] is a subset of ML that uses artificial neural networks with multiple layers of nodes to learn intricate representations from raw data. This approach mimics the way the human brain processes information, enabling the discovery of complex relationships within high-dimensional datasets like electronic health records (EHRs). While DL models are highly effective in finding patterns, their multi-layered structure can make it difficult to interpret the decision-making process behind their outputs. Despite this challenge, DL has shown great potential in analyzing vast amounts of healthcare data.

2.3. Reinforcement Learning

Reinforcement learning (RL) is an area of ML where intelligent agents learn to make decisions by interacting with their environment and receiving feedback in the form of rewards or penalties. RL has been successfully applied

in areas like robotics [11, 12, 13], gaming [14, 15], education [16] and finance. Recently, RL has gained attention as a promising method for developing personalized mental health interventions [17, 18, 19]. Mental health disorders, being a major source of disability worldwide, often require tailored treatments. RL can create individualized interventions by adapting to patient needs over time based on continuous feedback. One key strength of RL is its ability to work with dynamic and complex data, such as EHRs and wearable device outputs. It is particularly effective in handling incomplete or noisy data, which is frequently encountered in mental health studies. However, implementing RL in mental health care poses challenges related to privacy, transparency, and the interpretability of models. Despite these obstacles, RL holds great promise for creating adaptive, patient-centered treatment plans.

3. Methodology

In recent years, continuous and real-time monitoring technologies have gained significant traction due to the potential to enhance cognitive and behavioral performance while reducing healthcare costs. With the increasing need to monitor mental health, researchers have been investigating different technologies to develop efficient monitoring systems [20, 21]. Among these, eye-tracking technology has shown promise for its ability to track mental health indicators [22]. By applying computational methods to the extensive physiological data collected through eye-tracking, intelligent systems like IntelEye [23] can extract meaningful patterns by analyzing eye movement data such as pupil dilation, fixation points, and blink rates. IntelEye specifically uses the K Nearest Neighbor (KNN) algorithm to classify eye movement patterns into stress levels. The system processes this data by first segmenting the video-watching experience into different scenes, identifying the moments when stress indicators (e.g., dilation of pupils, rapid blinking) are detected. These stress-related signals are then correlated with the specific scenes, allowing IntelEye to detect when stress occurs and to identify the exact content that triggered it. This dual capability, detecting stress and linking it to specific triggers, makes IntelEye a powerful tool for monitoring mental health.

3.1. Study Selection

This section discusses three typical studies conducted over the past decade that explore the advancements in machine learning algorithms for mental health assessment. The focus is on research published from 2015 to 2023, although the review is not systematic and does not cover every possible study meeting the broader criteria. Relevant research was identified using PubMed, ScienceDirect, IEEE, and Google Scholar, with a focus on clinical studies applying machine learning to mental health. Studies involving untested or theoretical ML applications were excluded. The selected studies represent key original research efforts, as outlined in Table 1.

3.2. Review of Selected Studies

Researchers explored a method for automatically assessing depression severity by analyzing facial landmarks

Table 1: Typical ML and Mental Health Studies

Authors	Sample Size	Method	Performance
Anis Kacem et al. [24]	49	SL	84%
Subhagata Chattopadhyay et al. [25]	302	DL	95.5%
Fabian Wahle et al. [26]	28	SL	62%

and 3D head motion using barycentric coordinates and Lie-algebra rotation matrices [24]. Key features were extracted, processed, and encoded using Gaussian Mixture Models (GMM) and Fisher vector encoding. The study, which involved adults with chronic depression, achieved classification accuracy comparable to state-of-the-art deep learning models, while providing interpretable clinical insights.

Building on these computational approaches, a mathematical model was developed to reflect how psychiatrists evaluate depression symptoms [25]. Fourteen symptoms of adult depression were considered, in line with the Diagnostic and Statistical Manual (DSM-IV-TR). Principal Component Analysis (PCA) was used to reduce the number of symptoms to seven key features, which were then input into a hybrid system that combined Mamdani's fuzzy logic controller with a feed-forward multilayer neural network (FFMNN). The system was further refined using a backpropagation neural network (BPNN). This model, validated on 302 real-world depression cases and 50 controls, achieved an average diagnostic accuracy of 95.50

As technological interventions in mental health continue to evolve, [26] investigated the potential of smartphone-based interventions to support individuals with depressive symptoms by collecting data through the Mobile Sensing and Support (MOSS) app. The app gathered context-sensitive sensor data from participants and provided real-time, personalized interventions based on cognitive behavior therapy. Over an eight-week period, participants regularly completed self-reported depression surveys (PHQ-9). For those with clinical depression and high adherence, significant reductions in PHQ-9 scores were observed, indicating the effectiveness of the system in reducing depressive symptoms.

3.3. Results and Future Directions

After reviewing various works in the field, it is clear that eye gaze measures have been employed by multiple models to detect emotions, stress, cognitive load, mental fatigue, and other mental states [27, 28, 29]. This study builds upon these methodologies by developing a system to collect and analyze students' eye movements and physiological responses during remote learning activities, such as attending online lectures. Using machine learning (ML) algorithms, the data will be analyzed to identify patterns and anomalies that indicate changes in mental health, such as stress, anxiety, depression, or distraction. The model will be trained using a combination of eye-tracking data and mental health survey results. The system will then provide real-time feedback to both students and instructors, offering personalized support strategies, such as relaxation exercises or mindfulness training.

The system will be developed and tested in two phases.

In the first phase, participants will watch videos designed to elicit both positive and negative emotions while their eye movements are tracked. The gathered data will then be used to train a reinforcement learning (RL) model, which will learn to identify emotional states based on eye movements. In the second phase, the trained model will be integrated into an online platform that provides personalized mental health interventions based on the detected emotional states. This could include relaxation exercises, motivational messages, or other interventions aimed at improving student well-being.

To evaluate the system's performance, participants will complete mental health assessment questionnaires before and after using the platform, focusing on anxiety, depression, and stress. The accuracy, sensitivity, and specificity of the RL model in identifying emotional states will be assessed, along with changes in the participants' mental health scores. The study's overall effectiveness will be evaluated through a series of experiments and surveys, measuring changes in self-reported mental health and academic performance before and after the intervention. The results are expected to improve mental health support for engineering students by offering personalized, real-time interventions tailored to individual needs.

The bar chart in figure 1 summarizes the count of different emotions across all participants, providing an overview of the most to least frequently recorded emotions. Anger: Most frequently observed with 24 instances. Fear: Notably present, accounted for 17 instances. Happiness: Identified 14 times. Surprise and Disgust: Both emotions were less frequently detected, with 6 and 7 instances each. Sadness: Least frequent, with only 3 occurrences. These insights are based on the emotion label mapping which categorizes the emotions as Anger (0), Disgust (1), Fear (2), Happy (3), Sad (4), and Surprise (5).

Figure 2 Heatmap depicting the correlation between Emotion Day Mean and Overall Mean for various physiological markers (e.g., actigraphy counts, EDA, MET). Warmer colors indicate positive correlations, while cooler colors reflect negative correlations.

3.3.1. Key Findings

The analysis of the relationship between emotional states and physiological markers was conducted by comparing the differences between the emotion day mean and the overall mean across various biometric measures. The results are visualized in the heatmap (Fig.2.), which illustrates the correlations between emotional states (Anger, Fear, Happiness, Disgust, Surprise, and Sadness) and physiological variables, including actigraphy counts, electrodermal activity (EDA), respiratory rate, and other measures.
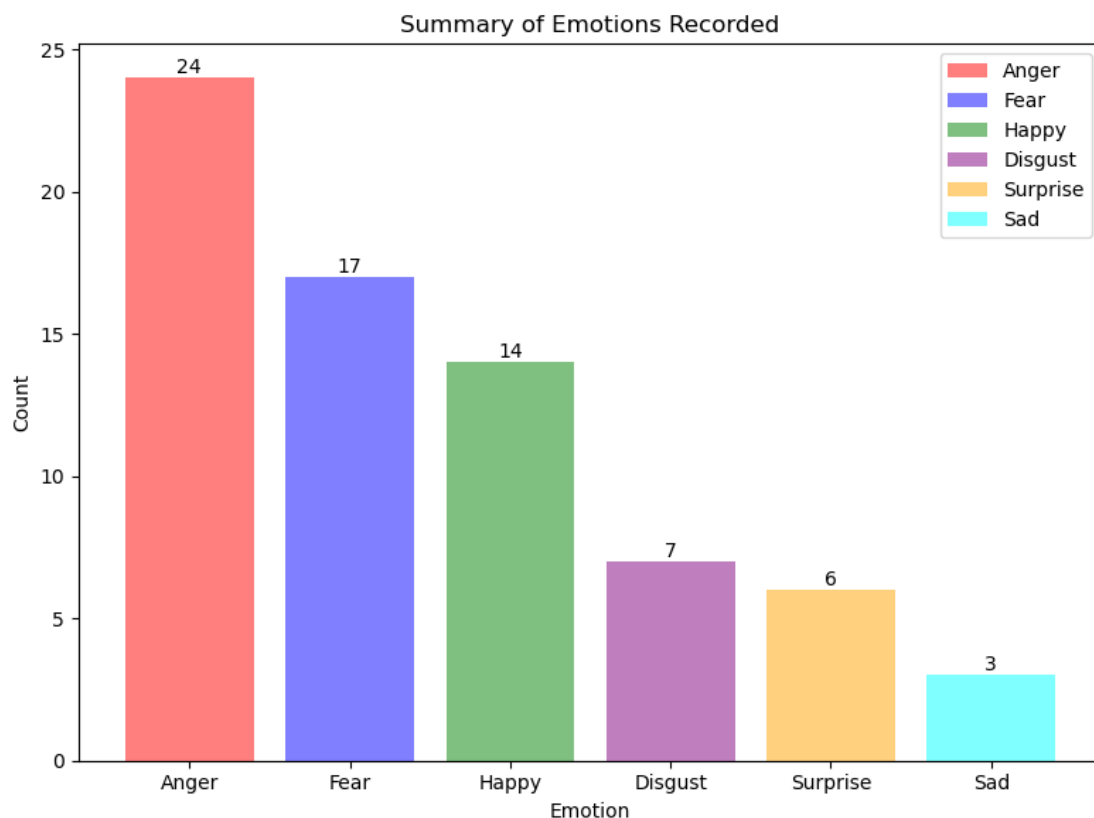
Figure 1: Summary of Emotions Recorded

1. **Electrodermal Activity (EDA)**: A strong positive correlation was observed between EDA and Anger ($r = 0.30$, $p = 0.013$), suggesting increased EDA levels during emotional episodes characterized by anger. Conversely, a significant negative correlation was found between EDA and Fear ($r = -0.26$, $p = 0.031$), indicating lower EDA levels during fearful states.

2. **Metabolic Equivalent (MET)**: Sadness showed a strong positive correlation with MET ($r = 0.32$, $p = 0.008$), suggesting higher energy expenditure on days when participants reported feeling sad.

3. **Activity Counts**: Activity counts demonstrated moderate positive correlations with Anger ($r = 0.24$, $p = 0.046$) and Surprise ($r = 0.24$, $p = 0.051$), implying increased physical activity during moments of these emotions.

4. **Respiratory Rate**: Sadness also correlated positively with respiratory rate ($r = 0.30$, $p = 0.015$), indicating that respiratory rates tended to be higher on days marked by sadness.

5. **Actigraphy Counts**: While most correlations involving actigraphy counts were non-significant, moderate correlations were observed for Sadness across different actigraphy measures, particularly on the Z-axis ($r = 0.25$, $p = 0.038$) and the vector magnitude ($r = 0.24$, $p = 0.051$), suggesting that emotional states may influence overall body movement.

Overall, the heatmap indicates that physiological changes are indeed associated with distinct emotional states.

Anger and Sadness, in particular, exhibit stronger relationships with biometric markers such as EDA, MET, and respiratory rate. These findings demonstrate the potential of using physiological signals to track and identify emotional states in real-time, thus informing targeted mental health interventions for students.

3.4. Limitations of ML and Mental Health

While ML and eye-tracking technologies show significant potential in monitoring mental health, there are some notable limitations. One challenge is the lack of clinical validation, which could hinder its readiness for real-world decision-making in clinical settings. The quality and size of the dataset are also critical factors affecting the performance of ML algorithms. Small sample sizes could lead to overfitting, and if models are only tested within a specific dataset, their generalizability may be limited. Furthermore, ML models often rely heavily on specific input features, meaning their predictions may only be accurate under certain conditions. Studies using binary classifiers also tend to oversimplify conditions and overlook their severity. Additionally, imbalanced datasets often lead to models that predict the majority class while missing rare events.

Future research must focus on recruiting larger, high-quality, and diverse datasets to address the challenges of participant retention and engagement. In this study, although initially recruited 50 participants, the complexity of maintaining engagement throughout the study resulted in 18 active participants by the end. While larger sample sizes might enhance the generalizability of findings, human factors, particularly in mental health studies, introduce complexities that make such generalizations difficult. The current findings should be viewed as a starting point, and
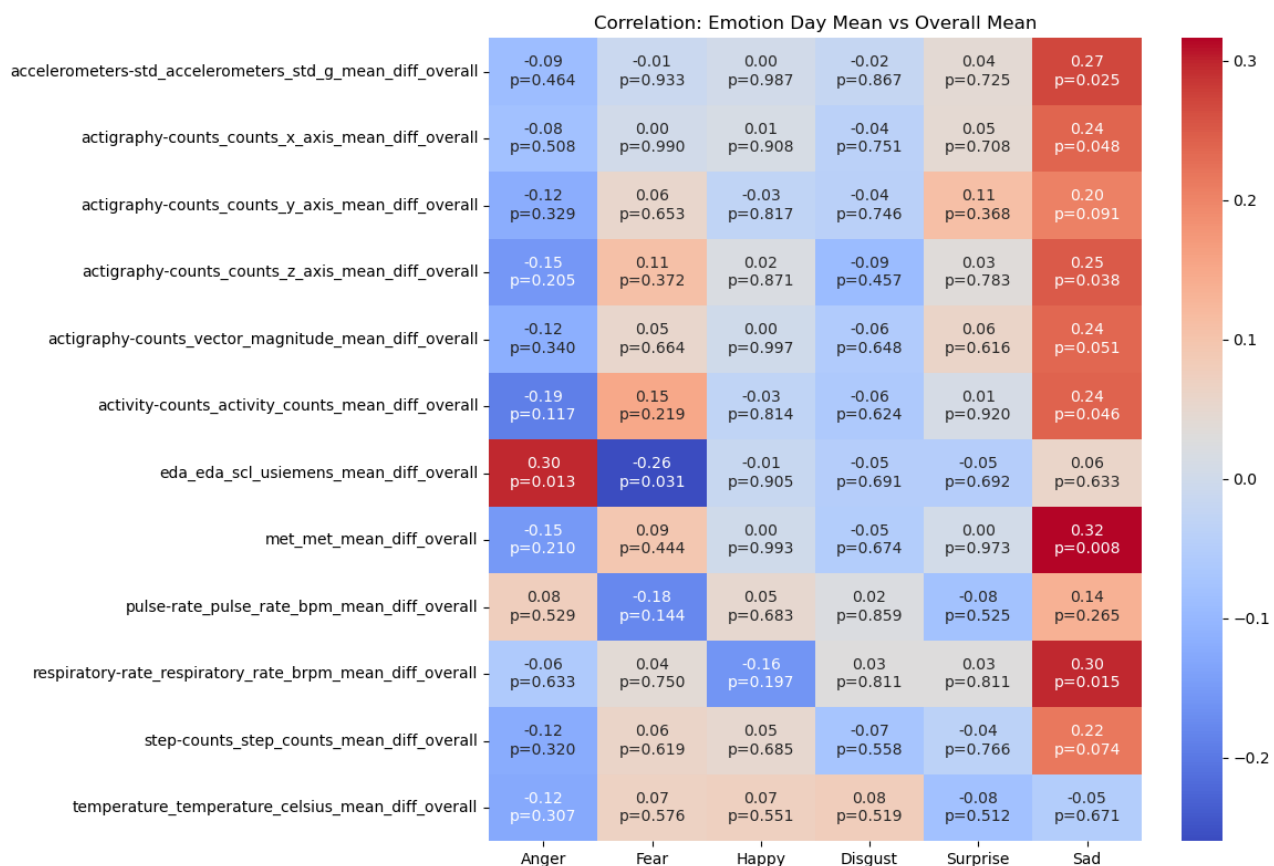
Figure 2: Emotion Day Mean vs. Overall Mean

future research should explore more diverse and larger populations to validate and extend these insights.

Collaboration among institutions for robust data sharing is essential to achieving larger, more representative datasets. Deep learning (DL) methods will become increasingly important in handling complex data, but ensuring that models remain interpretable is crucial. Transfer learning may also be beneficial in improving model performance, especially when working with diverse datasets. Researchers must consider the heterogeneity in input data and develop machine learning models capable of continuous learning to prevent "catastrophic forgetting." Interdisciplinary collaborations between data scientists and clinicians will likely yield the best results for mental health applications, helping to ensure the scalability and reliability of the findings.

3.5. Future Research Directions

Further research should explore a larger sample size and a longer study duration to enhance the generalizability of findings. Larger datasets will allow for more rigorous statistical analysis and reveal insights into the long-term impact of interventions. Personalized feedback and adaptive interventions based on individual stress levels would also improve the system's effectiveness. Expanding the range of physiological measures to include heart rate variability and electrodermal activity could provide a more comprehensive mental health assessment. Finally, researchers should investigate the scalability of the system for broader educational environments, identifying any practical challenges that may arise in larger-scale implementations.

Machine learning is increasingly becoming a cornerstone of digital medicine, with promising applications for mental health. However, to fully unlock its potential, collaboration across disciplines is critical. Clinicians, scientists, and data experts must work together to ensure that ML models are valid, reliable, and free from bias. Moreover, ethical considerations must be addressed, especially when deploying ML technologies in mental health care.

3.6. Recommendations

In conclusion, ML and eye-tracking technologies hold great promise in enhancing mental health support in engineering education. By delivering real-time feedback and personalized interventions, these technologies can help students better manage stress, anxiety, and other emotional challenges, leading to improved academic outcomes and overall well-being. However, further research is necessary to broaden the application of these techniques to other fields and student populations. Ethical concerns, particularly around the use of sensitive personal data in ML applications, must also be addressed. By fostering collaboration between engineers, data scientists, and mental health professionals, the potential benefits of ML in improving student mental health can be fully realized.

Conflict of Interest The authors declare no conflict of interest.

References

[1] R. Baltà-Salvador, N. Olmedo-Torre, M. Peña, A.-I. Renta-Davids, "Academic and emotional effects of online learning during the covid-19 pandemic on engineering students", *Education and information*

technologies, vol. 26, no. 6, pp. 7407–7434, 2021, doi:10.1007/s10639-021-10593-1.

[2] S. Behera, S. S. S. L. Paluri, A. Mishra, "Mental health status of students pursuing professional training: A questionnaire-based study", *Journal of education and health promotion*, vol. 10, no. 1, p. 399, 2021, doi:10.4103/jehp.jehp_1340_20.

[3] J. L. Hopton, S. M. Hunt, C. Shiels, C. Smith, "Measuring psychological well-being. the adapted general well-being index in a primary care setting: a test of validity", *Family Practice*, vol. 12, no. 4, pp. 452–460, 1995, doi:10.1093/fampra/12.4.452.

[4] M. K. Kovich, V. L. Simpson, K. J. Foli, Z. Hass, R. G. Phillips, "Application of the perma model of well-being in undergraduate students", *International journal of community well-being*, vol. 6, no. 1, pp. 1–20, 2023, doi:10.1007/s42413-022-00184-4.

[5] K. J. Hsu, M. E. McNamara, J. Shumake, R. A. Stewart, J. Labrada, A. Alario, G. D. Gonzalez, D. M. Schnyer, C. G. Beevers, "Neurocognitive predictors of self-reported reward responsivity and approach motivation in depression: A data-driven approach", *Depression and anxiety*, vol. 37, no. 7, pp. 682–697, 2020, doi:10.1002/da.23042.

[6] S. Moazemi, S. Vahdati, J. Li, S. Kalkhoff, L. J. Castano, B. Dewitz, R. Bibo, P. Sabouniaghdam, M. S. Tootooni, R. A. Bundschuh, *et al.*, "Artificial intelligence for clinical decision support for monitoring patients in cardiovascular icus: a systematic review", *Frontiers in Medicine*, vol. 10, p. 1109411, 2023, doi:10.3389/fmed.2023.1109411.

[7] B. Mahesh, "Machine learning algorithms -a review", *International Journal of Science and Research (IJSR)*, vol. 9, 2019, doi:10.21275/ART20203995.

[8] T. M. Mitchell, "Machine learning and data mining", *Commun. ACM*, vol. 42, no. 11, pp. 30–36, 1999, doi:10.1145/319382.319388.

[9] R. Sutton, A. Barto, "Reinforcement learning: An introduction", *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 1054–1054, 1998, doi:10.1109/TNN.1998.712192.

[10] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, J. Dean, "A guide to deep learning in healthcare", *Nature medicine*, vol. 25, no. 1, pp. 24–29, 2019, doi:https://doi.org/10.1038/s41591-018-0316-z.

[11] Y. Liu, Z. Zou, A. C. H. Tsang, O. S. Pak, Y.-N. Young, "Mechanical rotation at low reynolds number via reinforcement learning", *Physics of Fluids*, vol. 33, p. 062007, 2021, doi:10.1063/5.0053563.

[12] Z. Zou, Y. Liu, Y.-N. Young, O. S. Pak, A. C. H. Tsang, "Gait switching and targeted navigation of microswimmers via deep reinforcement learning", *Communications Physics*, vol. 5, p. 158, 2022, doi:10.1038/s42005-022-00935-x.

[13] Z. Zou, Y. Liu, A. C. Tsang, Y.-N. Young, O. S. Pak, "Adaptive micro-locomotion in a dynamically changing environment via context detection", *Communications in Nonlinear Science and Numerical Simulation*, vol. 128, p. 107666, 2024, doi:https://doi.org/10.1016/j.cnsns.2023.107666.

[14] Y. Liu, B. Zoghi, "Enhancing stem education using machine learning and reinforcement learning techniques for educational software and serious games", pp. 7148–7152, 2023, doi:10.21125/edulearn.2023.1871.

[15] Y. Liu, B. Zoghi, "Emerging technologies in education: Enhancing distance learning with technology-enhanced learning", pp. 7153–7157, 2023, doi:10.21125/edulearn.2023.1872.

[16] Y. Liu, H. Jiang, Z. Ben, "Employing artificial intelligence and machine learning to enhance student learning and outcomes with a focus on building trust and interaction", "EDULEARN24 Proceedings", 16th International Conference on Education and New Learning Technologies, pp. 3069–3074, IATED, 2024, doi:10.21125/edulearn.2024.0814.

[17] Y. Liu, Z. Ben, "Ai-powered strategies for alleviating graduate student burnout through emotional intelligence and wearable technology", "EDULEARN24 Proceedings", 16th International Conference on Education and New Learning Technologies, pp. 3041–3049, IATED, 2024, doi:10.21125/edulearn.2024.0809.

[18] Y. Liu, W. Lu, A. T. Zavareh, M. Rigsby, B. Zoghi, "Improving mental health support in engineering education using machine learning and eye-tracking", "2023 IEEE Frontiers in Education Conference (FIE)", pp. 1–5, IEEE Computer Society, Los Alamitos, CA, USA, 2023, doi:10.1109/FIE58773.2023.10343428.

[19] L. Fierro, Y. Liu, M. Rigsby, B. Zoghi, "Stress and happiness: Investigating stress tolerance and happiness in technical professionals", "INTED2023 Proceedings", 17th International Technology, Education and Development Conference, pp. 3964–3968, IATED, 2023, doi:10.21125/inted.2023.1053.

[20] A. Rehman, S. Abbas, M. Khan, T. M. Ghazal, K. M. Adnan, A. Mosavi, "A secure healthcare 5.0 system based on blockchain technology entangled with federated learning technique", *Computers in Biology and Medicine*, vol. 150, p. 106019, 2022, doi:https://doi.org/10.1016/j.compbiomed.2022.106019.

[21] Z. F. Khan, S. R. Alotaibi, "Applications of artificial intelligence and big data analytics in m-health: A healthcare system perspective", *Journal of healthcare engineering*, vol. 2020, no. 1, p. 8894694, 2020, doi:10.1155/2020/8894694.

[22] C. Jyotsna, J. Amudha, "Eye gaze as an indicator for stress level analysis in students", "2018 International conference on advances in computing, communications and informatics (ICACCI)", pp. 1588–1593, IEEE, 2018, doi:10.1109/ICACCI.2018.8554715.

[23] C. Jyotsna, J. Amudha, A. Ram, G. Nollo, "Inteleye: An intelligent tool for the detection of stressful state based on eye gaze data while watching video", *Procedia Computer Science*, vol. 218, pp. 1270–1279, 2023, doi:https://doi.org/10.1016/j.procs.2023.01.105.

[24] A. Kacem, Z. Hammal, M. Daoudi, J. Cohn, "Detecting depression severity by interpretable representations of motion dynamics", "2018 13th ieee international conference on automatic face & gesture recognition (fg 2018)", pp. 739–745, IEEE, 2018, doi:10.1109/FG.2018.00116.

[25] S. Chattopadhyay, "A neuro-fuzzy approach for the diagnosis of depression", *Applied computing and informatics*, vol. 13, no. 1, pp. 10–18, 2017, doi:https://doi.org/10.1016/j.aci.2014.01.001.

[26] F. Wahle, T. Kowatsch, E. Fleisch, M. Rufer, S. Weidt, *et al.*, "Mobile sensing and support for people with depression: a pilot trial in the wild", *JMIR mHealth and uHealth*, vol. 4, no. 3, p. e5960, 2016, doi:10.2196/mhealth.5960.

[27] L. Avila-Carrasco, D. L. Díaz-Avila, A. Reyes-López, J. Monarrez-Espino, I. Garza-Veloz, P. Velasco-Elizondo, S. Vázquez-Reyes, A. Mauricio-González, J. A. Solís-Galván, M. L. Martinez-Fierro, "Anxiety, depression, and academic stress among medical students during the covid-19 pandemic", *Frontiers in Psychology*, vol. 13, p. 1066673, 2023, doi:10.3389/fpsyg.2022.1066673.

[28] S. P. Behere, R. Yadav, P. B. Behere, "A comparative study of stress among students of medicine, engineering, and nursing", *Indian journal of psychological medicine*, vol. 33, no. 2, pp. 145–148, 2011, doi:10.4103/0253-7176.92064.

[29] S. Pourmohammadi, A. Maleki, "Stress detection using ecg and emg signals: A comprehensive study", *Computer methods and programs in biomedicine*, vol. 193, p. 105482, 2020, doi:10.1016/j.cmpb.2020.105482.

Yuexin Liu has completed her PhD degree in Mathematical Sciences from New Jersey Institute of Technology.

She is currently an instructional assistant professor at Department of Engineering Technology and Industrial Distribution, Texas A&M University, focusing her research on AI and well-being, RL and locomotory design.

Amir Zavareh has completed his PhD degree in Electrical and Electronics Engineering from Texas A&M University.

His research interests are Analog and mixed-signal bioelectronic design, Signal processing, Biomedical imaging and Biomedical instruments.

Ben Zoghi has completed his PhD degree in Biomedical and Electrical Engineering from Texas A&M University.

His research interests are RFID/Sensor technology and how to utilize wearable technology and artificial intelligence (AI)/ machine learning (ML) to develop technical leaders. During his 37 years tenure with Texas A&M College of Engineering, Ben secured over $15M in grants, 9 books and book-chapters and has over 125 peer-reviewed conference and journal papers.