IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 3, June 2025, pp. 2055~2071
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i3.pp2055-2071

Electrocardiogram sequences data analytics and classification using unsupervised and supervised machine learning algorithms

Sami Ghnimi, Pratapa Raju Moola, Jamaludeen Abdul Shariff
Department of Engineering, College of Engineering and Technology, University of Technology and applied Sciences-Ibra, Ibra, Oman

Article Info

Article history:

Received Aug 11, 2024
Revised Mar 1, 2025
Accepted Mar 15, 2025

Keywords:

Cardiovascular decease
ECG signal classification
Electrocardiograph
Feature selection algorithms
Machine learning algorithms
Support vector machines

ABSTRACT

This paper explores the prediction of cardiovascular disease (CVD) through the classification of electrocardiogram (ECG) sequences using both supervised and unsupervised machine learning (ML) algorithms. ECG 5000 dataset is considered to perform essential data analytics, clustering, and classification, effectively categorizing ECG heartbeats into optimal groups to forecast CVD. The Elbow and Silhouette methods are applied to estimate optimal number of clusters within the dataset. Using K-means and hierarchical clustering algorithms, the data is grouped into two and five distinguishable clusters, with performance metrics indicating that two clusters are more viable. Subsequently, multiple supervised ML classifiers—including kernel classifiers, support vector machine (SVM), naïve Bayes (NB), decision trees (DT), k-nearest neighbor (KNN) and neural networks (NN)—are trained on the labeled and clustered datasets to ensure accurate classification of ECG sequences and anomaly detection. A novel modified ML classifier, kernel-SVM with Chi-Square (χ^2) feature selection, is introduced and demonstrates exceptional performance, achieving an impressive accuracy of 0.9848, recall of 0.9973, and a training time of 1.6944 seconds, surpassing benchmarks from prior research. The results and discussion section includes a comparison of various algorithm performances, affirming that the proposed approach is an alternative to the complex deep learning (DL) and transformer-based models.

Corresponding Author:

Pratapa Raju Moola
Department of Engineering, College of Engineering and Technology
University of Technology and Applied Sciences-Ibra
Ibra, Oman
Email: pratapa.moola@utas.edu.om

1. INTRODUCTION

In accordance with the 2019-world health organization (WHO) report, it is evident that roughly 17.9 million people deceased, where 85% of these attributed to cardiovascular diseases (CVDs), underscoring CVD as a significant root of mortality [1]. The project of cardiovascular lifetime risk pooling, involving over 30,000 members from seven US cohort studies focusing on individuals aged sixty and older with low heart health, found that thirty-five years old have the highest risk of CVD, with the maximum among white males being 65.5%, followed by white females, black females, and black males being 57.1%, 51.9% and 48.4% respectively. Estimating of this risk accounted for the influence of competing risks of death from non-CVD causes. Examination of 14 studies investigating the effect of coronavirus disease 2019 (COVID-19) on individuals with preexisting CVD revealed that these individuals face a 2.25% relative rate of mortality due to COVID-19. This finding underscores the significantly heightened risk for this vulnerable demographic [1]. In 2020, a concerning statistic emerged: nearly 19 million deaths worldwide were attributed to CVD,

marking an 18.7% increase from the statistics recorded in 2010, indicating a significant rise in CVD-related mortality globally [2]. The 2022 report from the American heart association (AHA) states that CVD accounted for around 19.1 million deaths globally. Projections indicate a potential rise to 24.2 million deaths by the year 2030 [3]. An electrocardiogram (ECG) stands as a diagnostic modality employed to gauge the intricate electrical dynamics of the heart. Functioning as a visual representation, it delineates the microvolts electrical patterns manifested within the cardiac activity. The paramount objective of an ECG lies in interpreting the rhythm and tempo of the heart's contractions, alongside scrutinizing the intensity and sequential sequence of electrical signals coursing through distinct cardiac regions. It has the 'P' wave, 'QRS' complex, and 'T' wave. Detailed measurements of an ideal ECG 5000 sequences are provided in Figure 1.

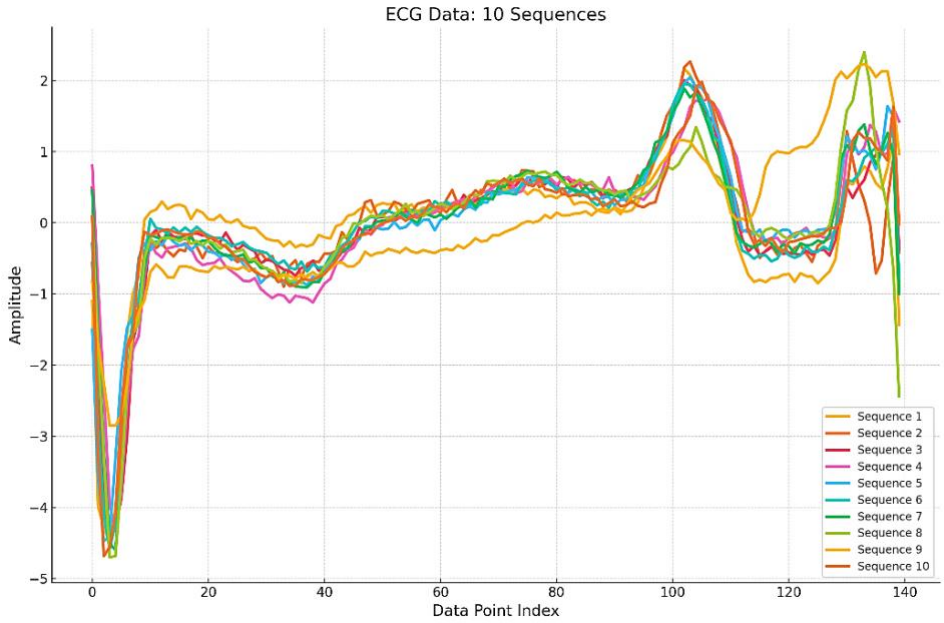


Figure 1. ECG 5000 data set: 10 sequences plot

The ECG 5000 dataset, widely studied and publicly available, has been a cornerstone in research focused on the prognosis of CVDs for over a decade. Numerous machine learning (ML) and deep learning (DL) algorithms recorded in the research, which are trained and tested for effective classification of ECG sequences. However, improving the accuracy and reliability of anomaly classification remains a critical area of research. Notably, comprehensive exploratory data analysis (EDA), clustering, and the integration of unsupervised and supervised algorithms are underexplored in literature. Eventually, a widespread examination is built of the existing research on ECG datasets and aims to address these gaps. This paper presents a holistic study that combines EDA with both unsupervised and supervised ML algorithms. The algorithms are trained and tested with and without feature selection techniques to enhance the classification of ECG signals and overcome limitations identified in previous studies.

To provide context for this work, the next section offers a detailed review of key research articles, emphasizing significant findings and identifying gaps in current approaches. The paper is structured as follows: section 2 examines associated works, focusing on the existing methodologies in detecting anomalous ECG signals. Section 3 outlines the methodology which includes EDA and approach of training unsupervised and supervised algorithms to address the classification challenge. Section 4 discusses the results, highlighting comparisons with existing approaches and drawing insights from the analysis. Section 5 concludes the study, summarizing the findings and presenting broader implications of the research.

2. RELATED WORK

Recent scholarly discourse on anomaly detection within sequential data, exemplified by ECGs, has increasingly leveraged sophisticated recurrent neural networks (RNNs) and convolutional neural networks (CNNs) renowned for their adeptness in capturing intricate sequential dependencies. Revolutionary

endeavors by Chauhan and Vig [4] underscore the efficacy of long short-term memory (LSTM) network for detecting anomalies within sequential data which are extended to ECG datasets. A plethora of methodologies has been advanced in the ECG sequences classificastion, ranging from support vector machine (SVM) methods proposed by Raj *et al.* [5] to time-domain analyses presented by Öztürk *et al.* [6]. Deep belief network (DBN) based methodologies proposed by Lu *et al.* [7] have garnered substantial traction in cardiac disease prognostication endeavors. Malhotra *et al.* [8] introduced a pre-trained deep recurrent neural network (DRNN) model tailored to extract salient sequential features conducive to supervised classification tasks. Hybrid architectures amalgamating RNNs and CNNs are demonstrated by [9], [10] have presented innovative nine-layer CNN architecture to address CVD prediction issues. The authors of [9], [10] have exemplified the evolving landscape of ECG anomaly detection methodologies. Innovative paradigms continue to emerge, as elucidated in [11] pioneering proposition of the local deep field (LDF) approach, which harnesses DL models to discern underlying class information amidst local variations. Yildirim *et al.* [12] introduced a 1D-CNN model tailored to detect arrhythmia. Furthermore, Yildirim [13] introduced deep bidirectional long short-term memory networks (BLSTMs) specifically optimized for classifying ECG signals. Oh *et al.* [14] have proposed composite network architectures fusing CNNs and LSTMs for enhanced ECG arrhythmia diagnosis.

While supervised learning methodologies have garnered significant attention, unsupervised anomaly detection techniques have been comparatively understudied. The emergence of advanced models like variational autoencoders (VAEs) and generative adversarial networks (GANs), known for their self-generative capabilities, has renewed academic interest in unsupervised anomaly detection techniques. Li *et al.* [15] utilized GANs to identify anomalies in time series data. Likewise, Hannun *et al.* [16] adapted deep neural networks (DNNs) to classify a diverse range of ECG rhythm patterns, highlighting the continuous advancements in anomaly detection methodologies. Yildirim *et al.* [17] achieved remarkable accuracy employing LSTM classifiers coupled with convolutional autoencoder (CAE) networks, boasting an accuracy of 99% on ECG arrhythmia dataset. Furthermore, Hou *et al.* [18] proposed an innovative fusion of LSTM and SVM to classify ECG arrhythmia effectively, and Pereira and Silveira [19] have also worked upon unsupervised learning of ECG sequences via VAEs, both results underscore the continuous evolution and diversification of anomaly detection methodologies within the ECG domain.

Ebrahimi *et al.* [20] have meticulously recorded a comprehensive review concerning the deployment and efficacy of a spectrum of DL methodologies, encompassing CNNs, DBNs, RNNs, LSTM networks, and gated recurrent units (GRUs). Their results emphasize the exceptional performance of DL methodologies in accurately classifying cardiac arrhythmias, with atrial fibrillation (AF) achieving a remarkable 100% accuracy, supraventricular ectopic beats (SVEB) attaining 0.998, and ventricular ectopic beats (VEB) achieving 0.997 accuracy. Notably, the utilization of GRU/LSTM architectures, CNNs, and LSTMs demonstrated expertise in achieving these remarkable classification accuracies, however extended training time and usage of graphics processing unit (GPU) for training pose real challenge in deployment. Khandual *et al.* [21] have recorded LSTM autoencoder-based approach which boasts an accuracy of 0.9793 in ECG anomaly detection, and Roy *et al.* [22] LSTM autoencoder model achieving an accuracy exceeding 0.98 on the ECG5000 dataset, alongside commendable precision, recall, and F1-scores. Matias *et al.* [23] utilized the VAE technique, incorporating a local similarity score, on the ECG5000 and MIT-BIH arrhythmia datasets analyzed in this study. Their approach yielded an area under the curve (AUC) comparable to previously reported results in the literature, achieving an impressive accuracy of 0.968.

Oluwasanmi *et al.* [24] introduced three distinct models for classifying ECG sequences: VAE, concat attention autoencoder (CAT-AE), and AE-without-attention. Ismail *et al.* [25] utilized a noise-free, augmented version of the ECG 5000 dataset to implement a temporal convolutional network (TCN). Using an 80/20 training/testing % split, they reported an accuracy of 0.9612, underscoring the effectiveness of temporal networks in capturing time-series patterns. Khalid *et al.* [26] worked with preprocessed and augmented ECG 5000 signals, applying CNN and GAN models. The CNN model achieved a strong accuracy of 0.98, while GAN models showed slightly lower performance at 0.9505. This study further confirmed the reliability of CNN in handling preprocessed datasets. Gladis *et al.* [27] focused on real-time signal clusters within the ECG 5000 dataset and proposed the use of a stripped NAS-network (SID-NASNet). Their approach demonstrated a high accuracy of 0.9822 using a 75/25 training/testing % split, highlighting the potential of novel neural architecture search methods for ECG analysis [27]. Ameen *et al.* [28] used augmented ECG 5000 signals to compare the performance of CNN, RNN, and random forest models, achieving a 10–15% improvement over traditional ML methods. Their work utilized a 70/30 training/testing % split to validate results, showing the benefits of DL over classical approaches. A comprehensive comparison of the previous work put up in classifying ECG 5000 data set is described in Table 1.

Following a thorough review, it has been confirmed that LSTM, VAE, RNN, DNN, CNN and LSTM are among the most effective algorithms for categorizing and identifying ECG time sequences. This

paper makes a significant contribution by performing a comprehensive data analysis of the ECG 5000 dataset through the following steps:

- EDA: generating elbow and silhouette curves to determine the optimal number of clusters.
- Unsupervised learning: training, testing, and evaluating clustering algorithms to segment the unlabelled data into two and five clusters effectively.
- Supervised learning: training, testing, and evaluating multiple supervised ML algorithms for the classification of anomalous ECG time sequences.
- The study culminates in proposing a kernel-SVM classifier integrated with a chi-square (χ^2) algorithm, which demonstrates an improvement over previously reported performance metrics in classifying labelled ECG 5000-time sequences.

The article highlights efforts to enhance classification accuracy reported in prior studies by [19], [22]–[28]. Detailed comparisons and advancements over these works are described under results section, underscoring the study's success in surpassing prior benchmarks on the ECG 5000 dataset.

Table 1. Recent signinficant research findings in the years 2021-2024: comparison

Authors/year	Dataset details	Algorithm	Training/ testing split	Accuracy
Ameen *et al.* [28]	ECG 5000, augmented signals	CNN, RNN, random forest	70% training / 30% testing	+10–15% over ML
Gladis *et al.* [27]	ECG 5000, real-time signal clusters	SID-NASNet	75% training / 25% testing	0.9822
Khalid *et al.* [26]	ECG 5000 (preprocessed), augmented signals	CNN, GAN	70% training / 30% testing	CNN: 0.98, GAN: 0.9505
Ismail *et al.* [25]	ECG 5000 (noise-free, augmented)	TCN	80% training / 20% testing	0.9612
Roy *et al.* [22]	ECG 5000, augmented data	Deep LSTM autoencoder	80% training / 20% testing	0.98
Matias *et al.* [23]	ECG 5000 (normal and abnormal samples)	VAE with 1D	Training: 80%; Validation: 20%	0.9711
Oluwasanmi *et al.* [24]	ECG 5000 dataset	VAE, AE-without-attention, CAT-AE, LSTM	70% training / 30% testing	VAE: 0.948, AE-without-attention: 0.956, CAT-AE: 0.958, LSTM: 0.984
Pereira and Silveira [19]	ECG 5000 dataset	VAE	20% training/ 80% of data	0.9843

3. METHODOLOGY

As depicted in Figure 2, the initial phase of the methodology involves cleaning and arranging ECG sequences dataset. This step is followed by the plotting of Elbow and Silhouette curves aimed at predicting the optimal number of clusters within the dataset. Subsequently, K-means and hierarchical unsupervised clustering algorithms are deployed to segment and label the data into five and two distinct classes. Based on performance metrics, we proceeded with supervised ML classifier algorithms for the two-class classification task. Furthermore, feature selection algorithm is employed and modified ECG sequences are used again to train the models. Averaged test results are then presented in the results section, providing a detailed account of our methodology and findings. Among the various available ECG datasets, MIT-BIH database of arrhythmia, physikalisch-technische bundesanstalt (PTB) diagnostic database of ECG, and databased of PhysioNet are widely explored by professionals and researchers in cardiology and biomedical engineering.

The dataset considered for this work is referred to as "ECG5000", it originates from an ECG recording of a 48-year-old patient afflicted with adverse congestive heart failure [29], and it is recorded for a duration of 20 hours from the patient. Automated annotation was utilized to assign class values to the data. The BIDMC congestive heart failure database (CHFD) data set which has 17,998,834 data points containing 92,584 heartbeats. Interpolated data of 5000 heartbeats are considered for the implementation with each sample representing a time sequence of Table 2, detailed descriptions of five and two classes and corresponding their labels are depicted.

3.1. EDA: ECG 5000 data set

The first step in building the classification model is conducting EDA, focusing on estimating the tangible number of groups in the dataset. Key metrics: are the distortion score and silhouette score (SS), which help determine the appropriate number of clusters. A detailed theoretical explanation of these metrics for the ECG5000 dataset is furnished in the following sections.

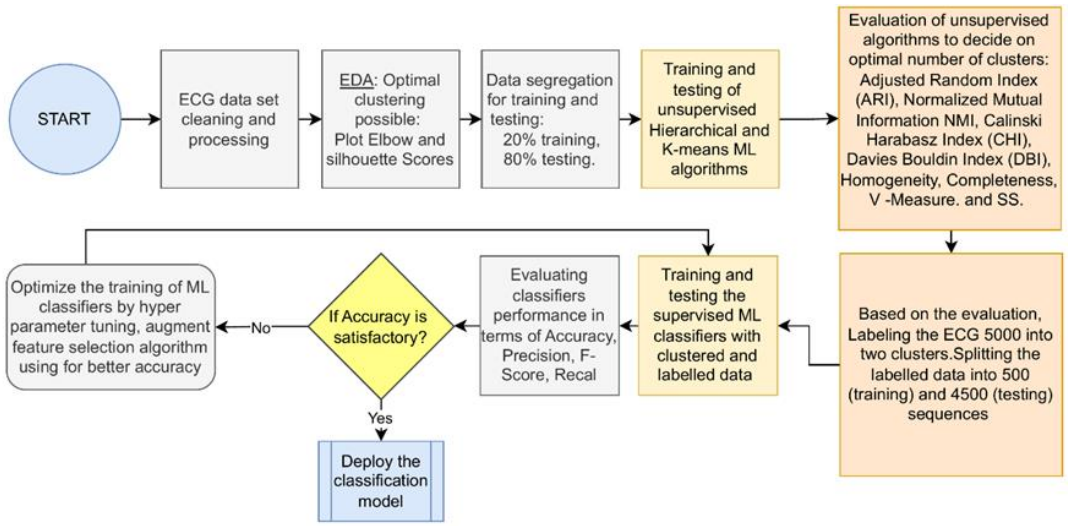


Figure 2. Flow chart: supervised and unsupervised ML classifier implementation

Table 2. ECG data set: two and five classes

S. No	Name of cluster label	Abbreviation (five classes)	Clusters	Abbreviation/description (two class)
1	Normal	N	1	0 (Normal)
2	R-on-T premature ventricular contraction	R	2	1(Anomaly)
3	Supraventricular premature or ectopic beat	S	3	1(Anomaly)
4	Premature ventricular contraction	V	4	1(Anomaly)
5	Unclassified beat	Q	5	1(Anomaly)

3.1.1. Elbow method

Elbow method is used to evaluate cluster cohesion by grouping similar data points. However, with increaseing number of clusters (K), the cohesion value will eventually approach '0', which alone does not indicate cluster quality. The method involves plotting the within-cluster sum of squares (WSS) on a graph, with the Y-axis showing the WSS value (also known as the distortion score) and the X-axis showing K where WSS gauges the compactness of the clusters [30]. WSS values can be calculated as per (1).

$$\text{WSS} = \sum_i \sum_{x \in Ci} (x - m_i)^2 \quad (1)$$

3.1.2. Silhouette method

The silhouette method employs a coefficient of silhouette, and it integrates separation and cohesion characteristics.

$$\text{If cohesion} < \text{separation: } S = 1 - \left(\frac{Cohesian\ measure}{Seperation\ measure}\right) \quad (2)$$

$$\text{If cohesion} > \text{separation: } S = \left(\frac{Cohesian\ measure}{Seperation\ measure}\right) - 1 \quad (3)$$

The silhouette coefficient/score is computed as per (2) and (3). If the separation measure surpasses the cohesion measure, the equation for SS will be as given in (2). Conversely, if the cohesion measure is greater than the separation measure, the SS is altered, and it will be as given in (3). A higher silhouette coefficient indicates a superior quality of clustering [30]. As per [29], the ECG 5000 data set is labeled with five and two classes. The same is affirmed with distortion and SS presented in results section.

3.2. Unsupervised algorithms for clusterring or grouping

After the assessment of elbow curve and silhouette curves with scores, K-means and hierarchical unsupervised clustering algorithms are considered to evaluate ECG 5000 data set. Both algorithms are trained and tested to ensure the possible clusters/classes exist in this data set.

3.2.1. K-means algorithm for clustering

This algorithm is one of the much deployed unsupervised ML algorithm in the clustering of datasets. This algorithm aims is to devide the data into 'K' clusters with each point assigned to the nearest cluster centroid. The algorithm starts by randomly initializing 'K' centroids and assigns data points to the closest

centroid based on a distance metric, usually Euclidean distance, forming the clusters. The centroids are then recalculated based on the mean of all points in their clusters. This process repeats until the centroids stabilize or a maximum number of iterations is reached, aiming to minimize within-cluster variance for compact and consistent clusters. Choosing the optimal value of 'K' can be estimated using methods like the elbow method, silhouette method, or gap statistic [31].

3.2.2. Hierarchical algorithm for clustering

Hierarchical clustering is a sophisticated technique for organizing data into a hierarchy of clusters, operating in two modes: agglomerative and divisive. Agglomerative clustering starts with each data point in its own cluster and merges them based on predefined criteria until all points form a single cluster. Divisive clustering begins with one large cluster and recursively splits it until each point is in its own cluster. A key aspect of this method is creating a distance matrix based on pairwise distances, which guides the clustering process. Linkage criteria, such as single, complete, average linkage, and ward's method, determine how clusters are merged or split. The method has been applied to ECG data clustering using hierarchical algorithms [32].

3.3. Supervised algorithms

Based on the performance analysis of unsupervised algorithms, several supervised ML based classifiers were developed for the ECG 5000 dataset. The dataset comprises 5,000 ECG sequences, each containing 140 instances. Of these, 80% of the sequences were used to train the classifier models, while the remaining 20% were reserved for testing. To this end, various kernel-based classifiers, including SVM, naïve Bayes (NB), decision tree (DT), k-nearest neighbor (KNN) and neural network (NN)-were trained and evaluated. A detailed discussion of the performance of these supervised classifiers is provided in the following section.

3.3.1. Decision tree classifiers

The foundational research on DT is first introduced in [33], focusing on three levels of DTs: 'fine,' 'medium,' and 'coarse,' used during training. Fine Trees are highly pruned, leading to fewer branches and a more streamlined decision-making process. Medium trees offer a balance between precision and complexity, moderately pruned to avoid overfitting while still identifying patterns. Coarse trees, with minimal pruning and numerous branches, excel in detecting intricate data patterns. As illustrated in Figure 3, the DT classifier implementation begins at the root node, involves node splitting, and concludes at the leaf nodes, where data is classified.

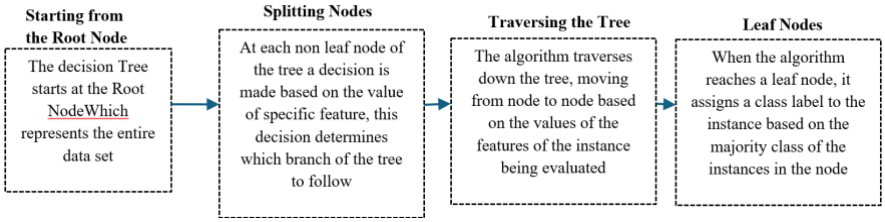


Figure 3. Decision tree workflow

3.3.2. Naïve Bayes classifiers

NB classifiers, grounded in Bayes' theorem, adopt a "naïve" assumption that all features are independent given the class label. Available in diverse variants such as Gaussian, multinomial, and Bernoulli, they are tailored to suit different data types. These classifiers estimate class priors and class-conditional probabilities from training data and subsequently make predictions by computing the posterior probability of each class given the features of a new instance. Renowned for their simplicity, computational efficiency, and efficacy, NB classifiers find widespread application in diverse domains including text classification, spam filtering, and medical diagnosis. Particularly lauded for their effectiveness in handling categorical attributes, the NB classifier is recognized as a straightforward and efficient tool for classification tasks [34]. NB classifiers basically functions based on the following equation of probability of class 'y' for features x_1, x_n. x_n of an instance is as defined in (4):

$$P(y|x_1, x_2, .. x_n) = \frac{P(x_1, x_2, .. x_n|y).P(y)}{P(x_1, x_2, .. x_n)} \quad (4)$$

Where, $P(y)$ = prior probability of class 'y', $P(x_1, x_2, .. x_n|y)$ = is the likelihood, representing the probability of observing the features given class 'y', $P(x_1, x_1, .. x_n)$ = the evidence probability, which serves as a

normalization factor and can be ignored for classification purposes since it's constant across all classes, $P(y|x_1, x_2, .. x_n)$ = probability of 'y' class for features x_1, x_2,.. x_n of an instance.

3.3.3. Support vector machine classifiers

SVM introduced by Vapnik in 1998, were initially designed to address two-class classification problems. SVM operates by identifying an optimal hyperplane to achieve maximum margin separation between classes. The fundamental concept underlying SVM is to determine the hyperplane that effectively separates the data into distinct classes while maximizing the margin between these classes [35]. The fundamental version of the decision function for the LSVM classifier is defined in (5):

$$f(x) = \{\sum_{i=1}^{N} \alpha_i + y_i \, K(x_i, x) + b\} \quad (5)$$

where, $f(x)$ = decision function, $\propto_i$ = language multiplier, y_i= class labels, $K(x_i, x)$ = kernel function, b = bias term or intercept, N = no. of support vectors. In accordance with the sign of the sum, $f(x)$ assigns specific class is assigned to the input $'x'$. If $f(x)$ is positive, 'x' is assigned to one class; if it's negative, 'x' is assigned to the other class.

3.3.4. K-nearest neighbor classifiers

This algorithm is in use since the 1970s, classifies new cases based on similarity measures by storing all cases. KNN is a nonparametric classification method, broadly categorized into two types: structure-less NN techniques and structure-based NN techniques. In structure-less techniques, the dataset is divided into training and test samples, and the nearest neighbor is identified based on the shortest distance between points. In structure-based techniques, methods such as orthogonal structure trees (OST), ball trees, k-d trees, axis trees, nearest future lines, and central lines are used to determine the structure of the data [36].

3.3.5. Neural network classifiers

The artificial neural network (ANN) classification algorithm uses ANN, modeled after the human brain, to classify input data into predefined categories. The network is made up of an input layer, one or more hidden layers, and an output layer. Each layer is composed of interconnected neurons that process and refine data by passing it through weighted connections, allowing the model to learn and make predictions. Training involves labeled data and backpropagation, where errors are propagated backward through the network to adjust weights and biases. Activation functions like sigmoid, rectified linear unit (ReLU), and SoftMax introduce non-linearity, enabling complex mappings between inputs and outputs. Various NN classifiers are trained and tested with different hyperparameters, and the results are discussed in [37].

3.3.6. Kernel classifiers

Kernel classifiers are powerful for classification tasks, especially with non-linearly separable data. Unlike linear classifiers such as logistic regression or linear SVMs, which can only draw linear decision boundaries, kernel classifiers address this limitation by projecting input features into a higher-dimensional space where linear separation is possible. This projection is achieved through the kernel trick, which allows computations in this higher-dimensional space without explicitly transforming the data. Instead of computing the dot product between vectors in the higher-dimensional space, kernel functions directly compute it in the original feature space, and they are as per (6) to (8):

$$K(x, x') = x.x' \quad (6)$$

$$K(x, x') = (\gamma x.x' + r)^{d} \quad (7)$$

$$K(x, x') = exp.(-\gamma||x - x'||^2) \quad (8)$$

Where, γ = control element of the influence of the dot product; r = coefficient; d = degree of the polynomial, and γ = control element of the spread of the kernel.

Kernel functions measure similarity or distance between data points in the original feature space and are crucial for kernel-based algorithms in ML and statistics. The choice of kernel function and its parameters can significantly impact classifier performance. In kernel-SVM, three key kernel functions are used: i) linear kernel: computes the dot product of input features, ii) polynomial kernel: calculates dot products of input vectors raised to a specific power, and iii) radial basis function (RBF) kernel: measures similarity using the Gaussian function Kernel-LR, an advanced form of traditional logistic regression, utilizes kernel functions to project data into a higher-dimensional space, improving its capability to capture non-linear relationships. This method improves performance on complex datasets like ECG5000 [38].

3.4. Chi-square feature selection algorithm

Feature selection involves choosing the essential attributes necessary to accurately represent data. By focusing on relevant features, classification algorithms typically enhance their classification accuracy, reduce the learning time, and simplify the underlying concepts. There is a wide array of feature selection algorithms available [39]. Chi- Square ($\chi 2$) feature selection algorithm proposed herewith relies on the chi-square ($\chi 2$) statistic and comprises two distinct phases. Initially, it sets a high significance level (sigLevel), like 0.5, for all numeric attributes to initiate discretization. The importance score is calculated as per (9), based on which most influencing signal features are extracted for training the ML algorithm:

$$\chi^2 = \sum_{m=1}^{2} \sum_{n=1}^{k} (B_{mn} - D_{mn})^2 \Big/ C_{mn} \quad (9)$$

where, K = number of classes; Bmn = number of patterns in mth interval, nth class; $\sum_{n=1}^{k} B_{mn}$: number of patterns in mth interval = Ri; $\sum_{m=1}^{2} A_{mn}$: number of patterns in mth interval = Cj; N: total no. of patterns = $\sum_{m=1}^{2} R_i$; Dmn: expected frequency of Bmn = Ri * Cj / N. If Ri or Cj is 0, Dmn is set to 0.1.

The degree of freedom of the $\chi 2$ statistic is one less the number of classes. Each attribute undergoes sorting based on its values, followed by the following steps: i) calculation of the $\chi 2$ values, as per (9), for every pair of adjacent intervals, ii) initially, each pattern resides in its interval containing only one attribute value, iii) merging of adjacent intervals possessing the lowest $\chi 2$ value. This merging continues until all interval pairs exhibit $\chi 2$ values surpassing the parameter determined by sigLevel, iv) initially set at 0.5, its corresponding $\chi 2$ value is 0.455 when the degree of freedom is 1, as explained further below, and v) this process repeats with a reduced sigLevel until an inconsistency rate exceeds the discretized data. Phase 1 essentially embodies a generalized version of Kerber's ChiMerge technique. Instead of setting a $\chi 2$ threshold, $\chi 2$ employs a loop that automatically adjusts the $\chi 2$ threshold (by decrementing sigLevel). It introduces consistency checking as a stopping criterion to ensure the discretized dataset accurately represents the original one. With these additions, $\chi 2$ autonomously determines an appropriate $\chi 2$ threshold that preserves the fidelity of the original data.

Phase 2 refines the process from Phase 1. Beginning with sigLevel 0 determined in Phase 1, each attribute ‘i’ is assigned a sigLevel [i] and takes turns in merging. Consistency checking occurs after each attribute's merging. If the inconsistency rate doesn't exceed, sigLevel [i] is reduced for the next round of merging; otherwise, attribute ‘i’ isn't further involved. This continues until no attribute's values can be merged. If an attribute is merged to only one value by the end of Phase 2, it signifies that the attribute isn't relevant in representing the original dataset. Consequently, when discretization concludes, feature selection is achieved [40].

4. RESULTS AND DISCUSSION

In this section, the results related to three key aspects of the ECG 5000 dataset are presented. First, the findings from EDA are discussed, with a focus on determining the optimal number of clusters. Following this, the performance metrics and outcomes of unsupervised ML algorithms applied for clustering—considering both two-class and five-class scenarios—are detailed. Finally, the classification results obtained using various supervised ML algorithms are reported, highlighting their comparative effectiveness in identifying ECG patterns.

4.1. Exploratory data analysis: elbow curve and SS

According to the elbow curve method, the optimal value of ‘K’ is identified when the graph exhibits a significant bend which denotes the optimal number of clusters. We have plotted an elbow curve for ECG 5000 data set to visualize the distortion score vs ‘K’. The "elbow" point in the plot indicates the optimal number of clusters where the distortion score starts to level off. As indicated in Figure 4, rate of change (decrement) in distortion is slowed down at ‘K’ =5 with the score of distortion score maintained consistently in the around 121042.463. As per the elbow curve visualization, distortion score tends to decrease with increasing ‘K’ value because the smaller clusters lead to lower inner-cluster distances. However, beyond a certain point, adding more clusters may not significantly reduce distortion, resulting in a less pronounced decrease in the distortion score. A lower distortion score indicates that data points are positioned closer to their respective cluster centroids, suggesting more compact and well-defined clusters. Thus, ECG 5000 data heartbeats can be possibly categorized into five types as they are illustrated in Table 2. Silhouette curve is also plotted for ECG 5000 data set to visualize the SS with respect to ‘K’ (no of clusters). As depicted in Figure 5, SS range from 0.45 to 0.20 for the ECG5000 data set. It is witnessed that the data set can be efficiently clustered into 2 classes with a higher SS of 0.45.

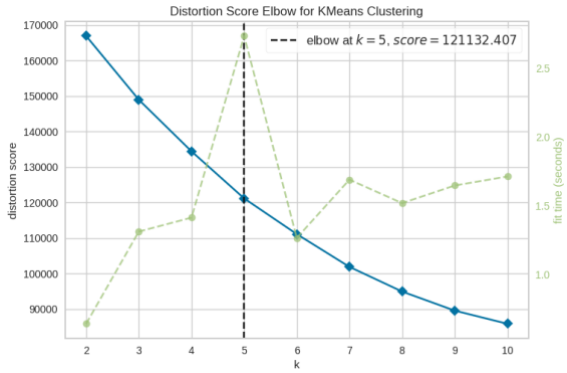


Figure 4. Elbow method: distortion score for ECG 500 data set

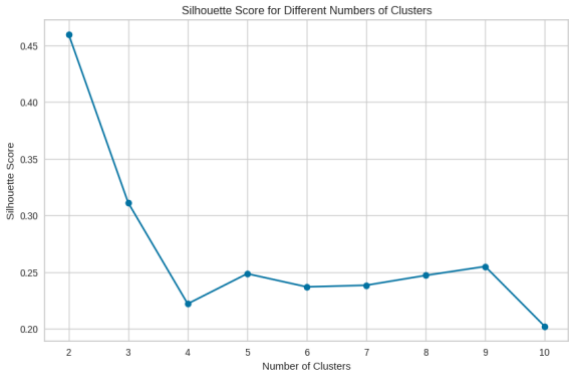


Figure 5. Silhouette method: SS for ECG 500 data set

4.2. Unsupervised algorithms: electrocardiogram 5000 data set clustering

The preceding analysis of the elbow graph and SS indicates that the ECG 5000 dataset can be effectively partitioned into either two or five clusters. The results in this section confirm the ideal set of clusters to be used for further analysis of the ECG 5000 dataset. Unsupervised K-Means and hierarchical ML clustering algorithms were trained and tested to classify the data into five and two distinct categories. The assessment of both clustering algorithms carried out by estimating the following eight performance parameters: adjusted rand index (ARI), normalized mutual information (NMI), calinski-harabasz index (CHI), davies-bouldin index (DBI), homogeneity, completeness, v-measure (Vm), and SS. Table 3 presents the scores for all performance metrics, with the highest scores highlighted for clarity, providing valuable insights into the effectiveness and efficiency of the clustering process.

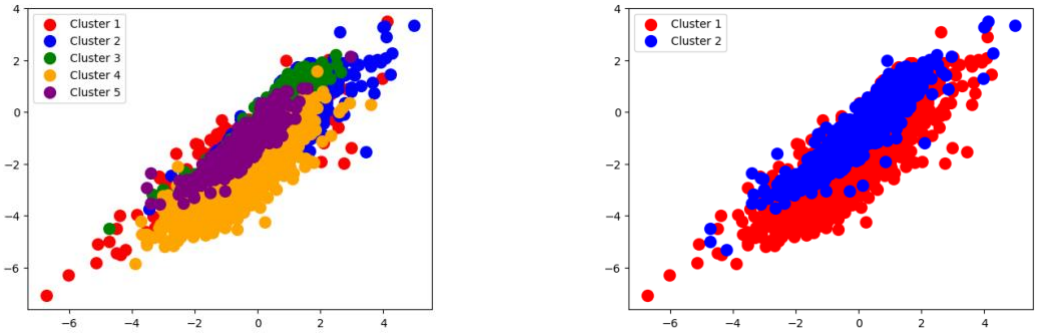
Table 3. Performance indexes of unsupervised clustering algorithms: k-means and hierarchical

Performance indexes	Hierarchical_2 clusters	Hierarchical_5 clusters	K-means_2 clusters	K-means_5 clusters
ARI	0.8255	0.5551	0.7853	0.4230
NMI	0.7669	0.5494	0.6877	0.4993
CHI	4449.1977	1918.3003	4730.194	2082.707
DBI	0.9184	1.6359	0.9173	1.6218
Homogeneity	0.7560	0.8228	0.6822	0.8033
Completeness	0.7782	0.4123	0.6934	0.3622
Vm	0.7669	0.5494	0.6877	0.4993
SS	0.4509	0.2601	0.4596	0.2506

ARI, typically ranging from -1 to 1, serves as a metric for evaluating clustering agreement, where a score of 1 indicates perfect clustering alignment, ‘0’ suggests clustering performance akin to random chance, and negative values indicate clustering discordance. As demonstrated in Table 3, ARI values for both hierarchical and K-means clustering algorithms notably excel for two clusters over five clusters. In contrast, the NMI values span from 0 to 1, with a score of 1 denoting flawless concordance between clustering outcomes, and 0 implying minimal shared information between clusters. Remarkably, NMI scores for hierarchical and K-Means clustering algorithms closely approach '1' for two clusters, unlike their performance for five clusters. The CHI endeavors to ascertain the optimal cluster count by maximizing the separation between clusters. The cluster count corresponding to the peak CHI is deemed optimal for the dataset. CHI values for hierarchical and K-means clustering algorithms peak at two clusters, while they remain relatively subdued for five clusters. On the other hand, the DBI lacks a singular optimal value. Lower DBI values are generally preferred, particularly when comparing diverse clustering solutions or determining the ideal cluster count. Notably, DBI values for hierarchical and K-means clustering for two clusters are lower compared to those for five clusters, underscoring their superior performance in the former scenario. Homogeneity serves as a measure of whether each cluster exclusively consists of members from a single class, with a perfect score of 1.0 indicating absolute homogeneity. An elevated homogeneity score suggests clusters populated solely by data points belonging to a singular class. Notably, homogeneity scores for both hierarchical and K-means clustering algorithms demonstrate superiority when employing five clusters compared to two clusters, implying that the former configuration optimally fulfills this criterion. Conversely, Completeness assesses whether all members of a particular class are assigned to the same cluster. A higher completeness score indicates greater consistency in assigning all data points associated with the same class to a single cluster, with a perfect score of 1.0 indicating

complet assignment coherence. Notably, both K-means and hierarchical algorithms exhibit completeness scores close to 1 for two clusters, implying their superior performance in this regard.

The Vm quantifies the harmonic mean of homogeneity and completeness, offering a balanced metric that considers both aspects. An ideal Vm score of 1.0 signifies optimal equilibrium between homogeneity and completeness. Mirroring other metrics, the Vm favors the two-cluster configuration with higher scores compared to five clusters for both K-Means and hierarchical algorithms, highlighting the superior balance achieved by the former. Lastly, the SS, spanning from -1 to 1, provides insight into the quality of clustering. Elevated scores indicate well-clustered data points, while negative scores suggest potential misassignments. Notably, SS for both K-Means and hierarchical clustering algorithms perform better for two clusters compared to five clusters, underscoring superior clustering quality for the former arrangement. Out of eight parameters observed, only one parameter is suggested for five optimal clusters, the rest are proposing two clusters. Detailed descriptions of performance indexes and results related to the unsupervised clustering algorithms are provided in Tables 3 and 4. Possible clusters in ECG 5000 dataset are presented in Figure 6, where Figure 6(a) indicates five clusters and Figure 6(b) indicates two clusters.



(a) (b)

Figure 6. 3D plots of ECG 5000 data set (a) 3D scatter plot with 5 clusters and (b) 3D scatter plot with 2 clusters

The effectiveness of K-means and hierarchical algorithms is assessed through the examination of recall, precision, F1-score, and accuracy which are computed as per (10) to (13) using false positives (FP), true negatives (TN), true positives (TP), and false negatives (FN) quantities derived from testing:

$$\text{Accuracy (A)} = \frac{TP+TN}{TP+TN+FN+FP} \tag{10}$$

$$\text{Precision (P)} = \frac{TP}{TP+FP} \tag{11}$$

$$\text{Recall (R)} = \frac{TP}{TP+FN} \tag{12}$$

$$\text{F} - \text{Measure} = \frac{(\beta^2+1).P.R}{\beta^2.P+R} \tag{13}$$

where, β = parameter adjusts the balance between precision and recall (β greater than or less than 1).

Results associated with unsupervised clustering algorithms are presented in Table 4, both hierarchical and K-Means algorithms witness higher accuracy, F1-score, recall, precision values for two class clustering over five class clustering. Amongst the two algorithms, hierarchical clustering algorithm is superior to K-means algorithm with higher accuracy, recall, F1-score, and precision scores. Indexes mentioned in Table 3 and results revealed in Table 4 concerning algorithms affirm that optimal clustering of ECG 5000 is achieved with 2 classes. Therefore, ECG 5000 dataset classified into two classes such as “normal” and “anomalous”, which are labeled with '0' and '1' respectively, and the same is used for further investigation of supervised classification algorithms to identify the anomalous ECG signal.

Table 4. Performance of unsupervised clustering algorithms: K-means and hierarchical

Unsupervised algorithm	% Accuracy		F1-Score		Recall		Precision	
	2-Clusters	5- Clusters	2-Clusters	5-Clusters	2-Clusters	5-Clusters	2-Clusters	5-Clusters
K-Means	0.9432	0.6182	0.94	0.72	0.94	0.62	0.94	0.91
Hierarchical	0.9544	0.486	0.95	0.53	0.95	0.49	0.96	0.61

4.3. Supervised machine learning for classification

A comprehensive training of multiple supervised ML classification algorithms is meticulously conducted using MATLAB 2022a. Classifiers such as DT, NB, SVM, KNN, NN, and Kernel classification algorithms with different learners and preset specifications are trained and tested with 2 class ECG 5000 labeled dataset. Amongst 5000 samples, 500 are considered for training and 4500 are considered for testing the performance of algorithms which is like the work presented in [19]. Average of accuracy scores of all the classifier algorithms trained and tested for three times are presented in Table 5.

Table 5. Performance of supervised DT based classifiers

Classifier type	Classifier with preset	Accuracy
DT	FT	0.9560
	MT	0.9560
	CT	0.9590
NB	GNB	0.9670
	KNB	0.9790
SVM	LSVM	0.9790
	QSVM	0.9813
	FGSVM	0.8990
	MGSVM	0.9770
	CGSVM	0.9810
KNN	FKNN	0.9780
	MKNN	0.9780
	CKNN	0.9580
	CosKNN	0.9790
	WKNN	0.9800
NN	MNN	0.9820
	WNN	0.9810
	BNN	0.9793
	TNN	0.9793
Kernel Classifiers	Kernel-SVM learner	0.9827
	Kernel-LR learner	0.9760

4.3.1. Decision tree classifier

Fine tree (FT), medium tree (MT), and coarse tree (CT) are classifiers categorized under DT classifiers. These algorithms were executed with varying preset conditions and a maximum number of splits. Details of the training specifications are given as: "fine tree classifier: Preset: coarse tree; maximum number of splits in the tree: 4, splits criterion: gini's diversity criterion". "medium tree: Preset: medium tree, maximum number of splits in the tree: 4, splits criterion: gini's diversity criterion". "Coarse tree: preset: coarse tree, maximum number of splits in the Tree: 4, splits criterion: gini's diversity criterion". As per the test results described in Table 5, CT algorithm exhibits superior performance with higher accuracy of 0.9590 over other classifiers.

4.3.2. Naïve Bayes classifier

Two NB classifiers by choosing different preset, distribution and kernel type are trained and tested and the results associated are presented in Table 5. Training specifications concerning two NB classifiers are given as: "Gaussian naïve Bayes (GNB): Preset=Gaussian NB; Numeric predictions distribution = Gaussian". "Kernel naïve Bayes (KNB): Preset=Kernel NB; Numeric predictions distribution = Kernel, kernel type= Gaussian". As per the results present in Table 5, amongst all the NB classifiers, KNB classifier exhibits superior performance with higher accuracy of 0.9790.

4.3.3. Support vector machine classifier

Five SVM classifiers with different presets, kernel functions, kernel scales, Box constraint levels and multi class methods are trained and tested, and the results associated are mentioned in Table 5. Training specifications pertaining to all the SVM classifiers are given: "Linear SVM (LSVM): Preset = Linear SVM; Kernel function = Linear; Kernel scale: Automatic; Box constraint level =1; Multi class method: one (vs) one". "Quadratic SVM (QSVM): Preset = Quadratic SVM; Kernel function = Quadratic; Kernel scale = Automatic; Box constraint level =1; Multi class method: one (vs) one". "Fine Gaussian SVM (FGSVM): Preset = Fine Gaussian SVM; Kernel function = Gaussian; Kernel scale = Automatic; Box constraint level =1; Multi class method: one (vs) one". "Medium Gaussian (MGSVM): Preset = Medium Gaussian SVM; Kernel function = Gaussian; Kernel scale =12; Box constraint level =1; Multi class method: one (vs) one". "Coarse Gaussian SVM (CGSVM): Preset = Coarse Gaussian SVM; Kernel function = Gaussian; Kernel scale =47; Box constraint level =1; Multi class method: one (vs) one". As per the results present in Table 5, amongst all SVM classifiers, QSVM classifier exhibits superior performance with higher accuracy of 0.9813.

4.3.4. K-nearest neighbor classifiers

Six KNN classifiers with different presets, number of neighbors, distance metrics, eclidean, distance weights and standardized data are trained and tested, and the results associated are mentioned in Table 5. Training specifications concerning all the KNN classifiers are as given as- "fine KNN (FKNN): preset = fine KNN; number of neighbors =1, distance metric = euclidean; distance weight: equal". "medium KNN (MKNN): preset = medium KNN; number of neighbors =10, distance metric = euclidean; distance weight: equal". "Coarse KNN (CKNN): preset = coarse KNN; number of neighbors =100, distance metric = euclidean; distance weight: equal". "Cosine KNN (CosKNN): preset = cosine KNN; number of neighbors =10, distance metric = cosine; distance weight: equal". "weighted KNN (WKNN): preset = weighted KNN; number of neighbors =10, distance metric = euclidean; distance weight: squared inverse". As per the results described in Table 5, amongst all KNN classifiers with different learners, WKNN classifier exhibits superior performance with higher accuracy of 0.9800.

4.3.5. Artificial neural network

Multiple NN classifiers with different preset, number of connected layers, first layer size, activation function, iteration limit and regularization strength (lamba) are trained and tested, and the results associated are mentioned in Table 5. Training specifications concering all the ANN classifiers are given as- "medium neural networks (MNN): preset = medium neural networks; number of connected layers =1, first layer size =25, activation function = ReLU, Iteration limit =1,000; regularization strength (lamba) =0". "wide neural networks (WNN): Preset = wide neural networks; number of connected layers =1, first layer size =100, activation function = ReLU, iteration limit =1,000; regularization strength (lamba) =0". "Bi-lateral neural networks (BNN): Preset = bi-layered neural networks; number of connected layers =2, first layer size =10, second layer size =10; activation function = ReLU, Iteration limit = 1000; regularization strength (lamba) =0". "Tri-Lateral neural networks (TNN): Preset = tri-layered neural networks; number of connected layers =3, first layer size =10, second layer size =10; third layer size =10, activation function = ReLU, iteration limit =1,000; regularization strength (lamba) =0". As per the results, MNN classifier exhibits superior performance with higher accuracy of 98.20% which is very close to kernel classifier trained with SVM learner.

4.3.6. Kernel classifiers

Two kernel classifiers with SVM and logistic regression learners are considered for the evaluation. In this regard, different presets, learners, number of expansion dimensions, regularization strength (Lamba), Kernel scale, multi class method and iteration limit are considered for training and testing, and the results associated are presented in Table 5. Training specifications considered for Kernel-SVM and kernel with logistic regression (Kernel-LR) are given as: "kernel-SVM: preset = SVM kernel, learner = SVM, number of expansion dimensions = auto, regularization strength (Lamba) = auto, Kernel scale = auto, multi class method = one (vs) one and iteration limit =1,000"."Kernel-LR: Preset = LR Kernel, learner = logistic regression, number of expansion dimensions = auto, regularization strength (Lamba) = auto, kernel scale = auto, multi class method = one (vs) one and iteration limit =1,000". Amongst the two Kernal classifiers, the kernel-SVM learner classifier exhibits superior performance with an accuracy of 98.27%. Kernel-SVM is evident to be the best fit algorithm amongst all the classifier algorithms implemented in this work, thus it is considered for further deployement of proposed classification apporach.

4.3.7. Proposed novel approach: kernel-SVM classifier with $\chi 2$ feature selection algorithm

The best performing kernel-SVM algorithm is further explored by training with the modified ECG sequences. Consequently, a novel approach 'kernel-SVM with $\chi 2$ algorithm' is proposed, the process flow for the same is depicted in Figure 7. Firstly, $\chi 2$ feature selection algorithm is run, and the modified ECG sequences are considered for further implementation. Modified signal depicted in Figure 8(b) has only 125 features/data points per sequence/heartbeat. $\chi 2$ Algorithm has importance score in the range of 2.9101< Importance score <184.7235; features with importance scores less than 10 are excluded. Thus, only 125 features out of 140 are considered to train kernel-SVM. Features (Instants) 55, 81, 66, 63, 58, 1, 84, 57, 60, 61, 65, 64, 83, 15 and 82 are excluded from the ECG sequence. It is then deployed with kernel function to arrange the data from 2-dimensional space to 3-dimensional space. Subsequently SVM function is called to separate the data using support vector hyperplane which leads to effective learning under the supervision of input and outputs. For improved reader comprehension, an ECG sequence representing one heartbeat before and after applying the χ^2 (Chi-squared) feature selection algorithm is presented in Figure 8. Figure 8(a) illustrates the features considered prior to applying the χ^2 algorithm, while Figure 8(b) depicts the features retained after feature selection. MATLAB 2023b is used to train and test ML based classifier algorithms.

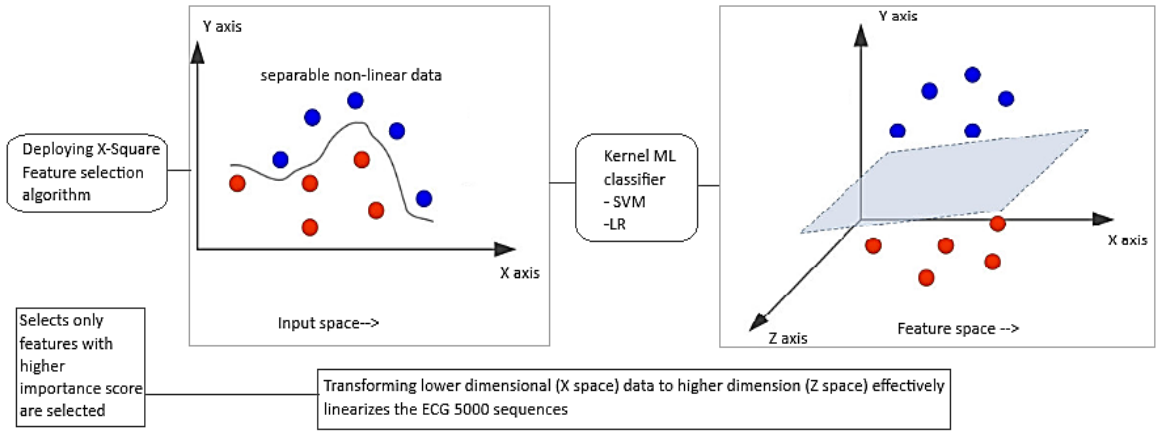


Figure 7. Proposed approach: Kernel-SVM + χ2 algorithm

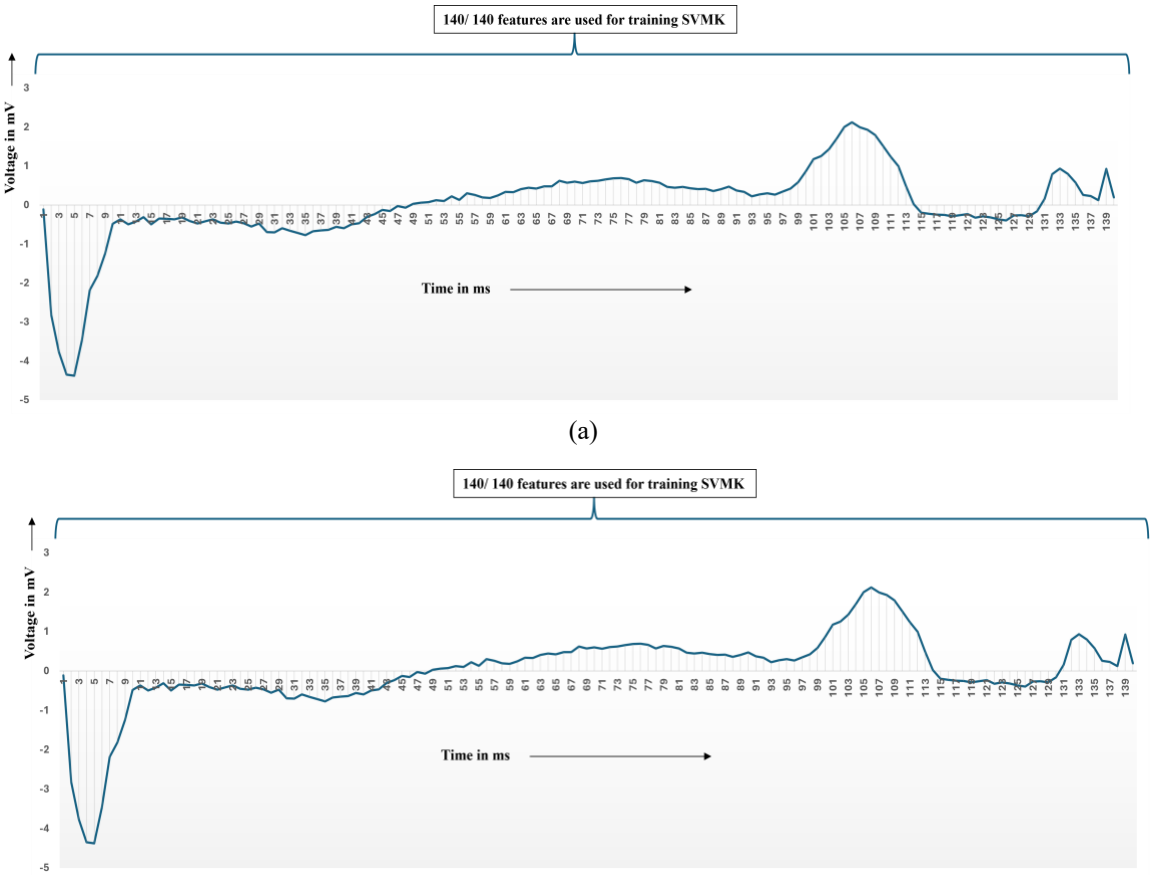


Figure 8. ECG 5000 data set: (a) One heartbeat without χ2 algorithm; (b) One heartbeat with χ2 algorithm

Confusion matrices resulted from proposed implementation is presented in Figure 9, specially, Figure 9(a) illustrates the confusion matrics for the Kenel + SVM model with χ2 algorithm and Figure 9(b) illustrates confusion matrics for Kenel + SVM model with χ2 algorithm. Kernel-SVM without χ2 algorithm has 60 FPs, 1813 TPs, 4 FNs, 2622 TNs, whereas Kernel -SVM with χ2 algorithm has 78 FPs, 1795 TPs, 5 FNs, 2621 TNs over all the 4500 samples considered for testing while 500 samples considered for training. Proposed algorithm is tested for 10 times and averaged performance metrics: accuracy, precision, recall, F1-score, training time, and prediction speed are recoorded. A comprehensive analysis is established by comparing the performance of proposed Kernel-SVM + χ2 with F-tALSTM-FCN from [9], VRAE+SVM

from [19], Deep LSTM Autoencoder from [22], VAE from [23] and [24], AE-Without-Attention from [24], CAT-AE from [24], TCN from [25], CNN, GAN from [26] SID-NASNet from [27] and the same is presented in Table 6. Proposed approach of Kernel-SVM + χ2 algorithm outperformed other algorithms, it is quick in training and prediction aspects compared to heavy LSTM and DL algorithm notified in the research records. And also, proposed ML based kernel-SVM + χ^2 algorithm is the best alternative to complex DL and transformer algorithms to run on the edge devices. Unlike DL algorithms, ML based classification algorithms do not need graphical processing units (GPUs) to perform computation-intensive ECG anomaly classification tasks on the edge devices.

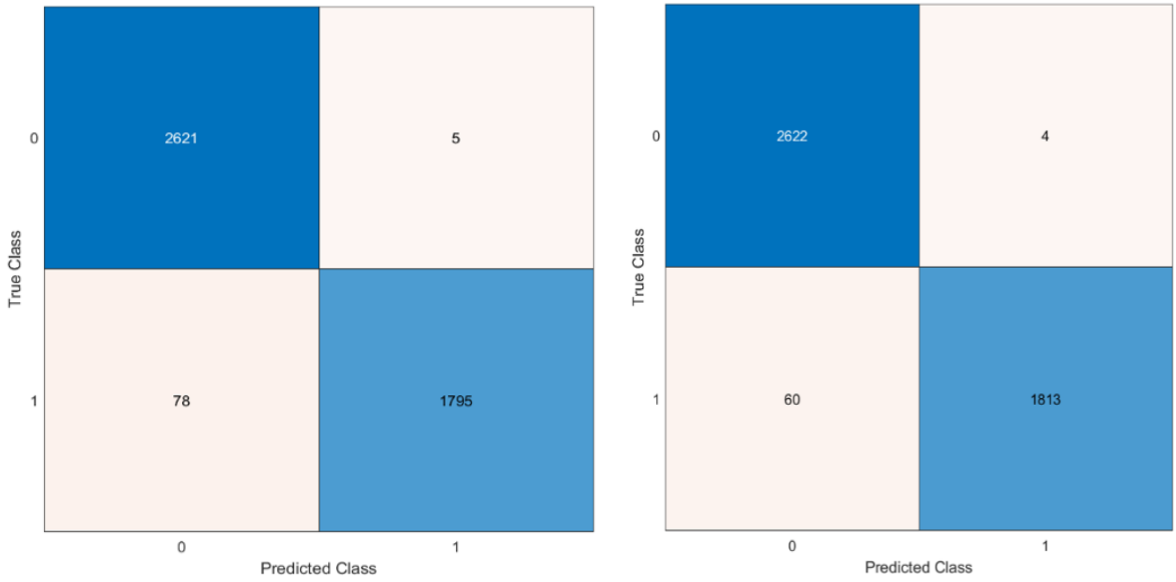


Figure 9. Confusion matrix: (a) Kenel-SVM with χ2 algorithm (b) Kenel-SVM without χ2 algorithm

Table 6. Comparative analysis of different supervised ML classifiers

Classifier models	Performance Parameters					
	Accuracy	Precision	Recall	F1- Score	Training time in seconds	Prediction in obs/sec
F-tALSTM- FCN [9]	0.9496	-	-	-	-	-
VRAE + SVM [19]	0.9843	-	-	0.9844	-	-
VAE [23]	0.968	0.984		0.957	-	-
VAE [24]	0.952		0.925	0.954	-	-
AE-Without-Attention [24]	0.97	0.988	0.955	0.971	-	-
CAT-AE [24]	0.972	0.992	0.956	0.974	-	-
LSTM-AE [21]	0.9793	-	-	-	-	-
TCN [25]	0.9612	-	-	-	-	-
CNN, GAN [26]	CNN: 0.98, GAN: 0.9505					
SID-NASNet [27]	0.9822	-	-	-	-	-
Proposed: kernel-SVM + χ2	0.9848	0.9640	0.9973	0.9804	1.6944	9040

5. CONCLUSION

This study demonstrates significant advancements in the prediction of CVD through the classification of ECG sequences using both supervised and unsupervised ML techniques. By leveraging the ECG 5000 dataset, the research effectively identified optimal clustering and classification strategies, utilizing the Elbow and Silhouette methods to determine cluster viability. Among unsupervised approaches, K-means and hierarchical clustering were tested, with the two-cluster solution emerging as the most practical for stratifying ECG data. The introduction of the kernel-SVM with Chi-Square (χ^2) feature selection stands out as a key innovation, delivering exceptional performance metrics, including a high accuracy of 98.48%, recall of 99.73%, and minimal training time of 1.6944 seconds. These results not only surpass benchmarks reported in prior studies but also underscore the efficiency and reliability of the proposed methodology for anomaly

detection in ECG classification. This work highlights the potential of combining robust clustering techniques with innovative classifiers to enhance predictive performance while optimizing computational efficiency. Looking forward, integrating these ML classifiers into edge devices presents a promising area of research for developing cost-effective solutions in CVD prognostics. Collectively, these findings furnish compelling evidence that substantiates the effectiveness of our proposed approach in the realm of ECG sequences classification.

ACKNOWLEDGMENTS

We extend a deep sense of thanks to the Research and Consultation department and Finance Department at University of Technology and Applied Sciences-Ibra for their unwavering support.

FUNDING INFORMATION

This research was funded by The Research Council, Oman under: Research Grants-2021, grant number: BHP/RG/HSS/21/206.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Sami Ghnimi	✓	✓	✓		✓	✓		✓	✓	✓			✓	
Pratapa Raju Moola	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓		
Jamaludeen Abdul Shariff		✓		✓		✓	✓			✓	✓		✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors affirm that the funding source had no influence on the design, execution, analysis, or interpretation of the research findings presented herein. It's crucial to emphasize that the authors have implemented measures to uphold the integrity and impartiality of both the research procedure and its outcomes. Any conceivable bias originating from the funding source has been mitigated through strict adherence to scientific methodology and ethical principles.

DATA AVAILABILITY

The following resources used in this study are publicly accessible:

- The data that support the findings of this study are openly available in https://www.kaggle.com/datasets/salsabilahmid/ecg50000/data.
- Python code pertaining to the elbow curve and silhouette score estimation towards clustering the ECG5000 alongside the performance metrics of unsupervised machine learning algorithms can be bound in this link- https://colab.research.google.com/drive/1v81CNR_yINfAW6nB88YYbpVBg-eYgHiR?usp=drive_link.

REFERENCES

[1] World Health Organization, "Cardiovascular diseases (CVDs)," *World Health Organization*, May 2024. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).
[2] C. W. Tsao *et al.*, "Heart disease and stroke statistics-2022 update: a report from the American Heart Association," *Circulation*, vol. 145, no. 8, Feb. 2022, doi: 10.1161/CIR.0000000000001052.
[3] American Heart Association, *Annual Report 2023–2024: 100 Years of Impact*. Dallas, TX: American Heart Association, 2024. [Online]. Available: www.heart.org.

[4] S. Chauhan and L. Vig, "Anomaly detection in ECG time signals via deep long short-term memory networks," in *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–7, Oct. 2015, doi: 10.1109/DSAA.2015.7344872.

[5] S. Raj, K. C. Ray, and O. Shankar, "Cardiac arrhythmia beat classification using DOST and PSO tuned SVM," *Computer Methods and Programs in Biomedicine*, vol. 136, pp. 163–177, Nov. 2016, doi: 10.1016/j.cmpb.2016.08.016.

[6] D. K. -Öztürk, H. A. Güvenir, U. Ravens, and N. Baykal, "A window-based time series feature extraction method," *Computers in Biology and Medicine*, vol. 89, pp. 466–486, Oct. 2017, doi: 10.1016/j.compbiomed.2017.08.011.

[7] P. Lu *et al.*, "Research on improved depth belief network-based prediction of cardiovascular diseases," *Journal of Healthcare Engineering*, vol. 2018, pp. 1–9, 2018, doi: 10.1155/2018/8954878.

[8] P. Malhotra, V. TV, L. Vig, P. Agarwal, and G. Shroff, "Time net: pre-trained deep recurrent neural network for time series classification," *arXiv-Computer Science*, 2017.

[9] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018, doi: 10.1109/ACCESS.2017.2779939.

[10] U. R. Acharya *et al.*, "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, vol. 89, pp. 389–396, Oct. 2017, doi: 10.1016/j.compbiomed.2017.08.022.

[11] W. Li and J. Li, "Local deep field for electrocardiogram beat classification," *IEEE Sensors Journal*, vol. 18, no. 4, pp. 1656–1664, 2018, doi: 10.1109/JSEN.2017.2772031.

[12] Ö. Yildirim, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Computers in Biology and Medicine*, vol. 102, pp. 411–420, Nov. 2018, doi: 10.1016/j.compbiomed.2018.09.009.

[13] Ö. Yildirim, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification," *Computers in Biology and Medicine*, vol. 96, pp. 189–202, May 2018, doi: 10.1016/j.compbiomed.2018.03.016.

[14] S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya, "Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats," *Computers in Biology and Medicine*, vol. 102, pp. 278–287, Nov. 2018, doi: 10.1016/j.compbiomed.2018.06.002.

[15] D. Li, D. Chen, J. Goh, and S. Ng, "Anomaly detection with generative adversarial networks for multivariate time series," *arXiv-Computer Science*, 2018.

[16] A. Y. Hannun *et al.*, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature Medicine*, vol. 25, no. 1, pp. 65–69, Jan. 2019, doi: 10.1038/s41591-018-0268-3.

[17] O. Yildirim, U. B. Baloglu, R. S. Tan, E. J. Ciaccio, and U. R. Acharya, "A new approach for arrhythmia classification using deep coded features and LSTM networks," *Computer Methods and Programs in Biomedicine*, vol. 176, pp. 121–133, 2019, doi: 10.1016/j.cmpb.2019.05.004.

[18] B. Hou, J. Yang, P. Wang, and R. Yan, "LSTM-based auto-encoder model for ECG arrhythmias classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 4, pp. 1232–1240, Apr. 2020, doi: 10.1109/TIM.2019.2910342.

[19] J. Pereira and M. Silveira, "Unsupervised representation learning and anomaly detection in ECG sequences," *International Journal of Data Mining and Bioinformatics*, vol. 22, no. 4, 2019, doi: 10.1504/IJDMB.2019.101395.

[20] Z. Ebrahimi, M. Loni, M. Daneshtalab, and A. Gharehbaghi, "A review on deep learning methods for ECG arrhythmia classification," *Expert Systems with Applications: X*, vol. 7, Sep. 2020, doi: 10.1016/j.eswax.2020.100033.

[21] A. Khandual, K. Dutta, R. Lenka, S. R. Nayak, and A. K. Bhoi, "MED-NET: a novel approach to ECG anomaly detection using LSTM auto-encoders," *International Journal of Computer Applications in Technology*, vol. 65, no. 4, 2021, doi: 10.1504/IJCAT.2021.10040403.

[22] M. Roy, S. Majumder, A. Halder, and U. Biswas, "ECG-NET: a deep LSTM autoencoder for detecting anomalous ECG," *Engineering Applications of Artificial Intelligence*, vol. 124, Sep. 2023, doi: 10.1016/j.engappai.2023.106484.

[23] P. Matias, D. Folgado, H. Gamboa, and A. Carreiro, "Robust anomaly detection in time series through variational autoencoders and a local similarity score," in *Proceedings of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies*, pp. 91–102, 2021, doi: 10.5220/0010320500910102.

[24] A. Oluwasanmi, M. U. Aftab, E. Baagyere, Z. Qin, M. Ahmad, and M. Mazzara, "Attention autoencoder for generative latent representational learning in anomaly detection," *Sensors*, vol. 22, no. 1, Dec. 2021, doi: 10.3390/s22010123.

[25] A. R. Ismail, S. Jovanovic, N. Ramzan, and H. Rabah, "ECG classification using an optimal temporal convolutional network for remote health monitoring," *Sensors*, vol. 23, no. 3, Feb. 2023, doi: 10.3390/s23031697.

[26] M. Khalid, C. Pluempitiwiriyawej, S. Wangsiripitak, G. Murtaza, and A. A. Abdulkadhem, "The applications of deep learning in ECG classification for disease diagnosis: a systematic review and meta-data analysis," *Engineering Journal*, vol. 28, no. 8, pp. 45–77, Aug. 2024, doi: 10.4186/ej.2024.28.8.45.

[27] K. P A. Gladis, A. Ahilan, N. Muthukumaran, and L. Jenifer, "ECG arrhythmia measurement and classification for portable monitoring," *Measurement Science Review*, vol. 24, no. 4, pp. 118–128, Aug. 2024, doi: 10.2478/msr-2024-0017.

[28] A. Ameen, I. E. Fattoh, T. Abd El-Hafeez, and K. Ahmed, "Advances in ECG and PCG-based cardiovascular disease classification: a review of deep learning and machine learning methods," *Journal of Big Data*, vol. 11, no. 1, Nov. 2024, doi: 10.1186/s40537-024-01011-7.

[29] A. L. Goldberger *et al.*, "PhysioBank, physioToolkit, and physioNet," *Circulation*, vol. 101, no. 23, Jun. 2000, doi: 10.1161/01.CIR.101.23.e215.

[30] D. M. Saputra, D. Saputra, and L. D. Oswari, "Effect of distance metrics in determining k-value in k-means clustering using elbow and silhouette method," in *Proceedings of the Sriwijaya International Conference on Information Technology and Its Applications (SICONIAN 2019)*, 2020, doi: 10.2991/aisr.k.200424.051.

[31] Ö. Yakut, E. D. Bolat, and H. Efe, "K-means clustering algorithm based arrhythmic heart beat detection in ECG signal," *Balkan Journal of Electrical and Computer Engineering*, vol. 9, no. 1, pp. 53–58, Jan. 2021, doi: 10.17694/bajece.814473.

[32] R. Ahmed and S. Arafat, "Cardiac arrhythmia classification using hierarchical classification model," in *2014 6th International Conference on Computer Science and Information Technology (CSIT)*, pp. 203–207, Mar. 2014, doi: 10.1109/CSIT.2014.6806001.

[33] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991, doi: 10.1109/21.97458.

[34] Irina Rish, "An empirical study of the naïve Bayes classifier," *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, 2001.

[35] V. N. Vapnik, *Statistical learning theory*. Wiley-Inter science, 1998.

[36] N. Bhatia and Vandana, "Survey of nearest neighbor techniques," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 80, no. 2, pp. 302–305, 2010.

[37] G. Bortolan, R. Degani, and J. L. Willems, "Neural networks for ECG classification," in *[1990] Proceedings Computers in Cardiology*, pp. 269–272, doi: 10.1109/CIC.1990.144212.
[38] C. M. Bishop, *Pattern recognition and machine learning*, vol. 16, no. 4. Springer, 2006, doi: 10.1117/1.2819119.
[39] K. Kira and L. A. Rendell, "The feature selection problem: traditional methods and a new algorithm," in *Proceedings Ninth National Conference on Artificial Intelligence*, pp. 129–134, 1992.
[40] Huan Liu and R. Setiono, "Chi2: feature selection and discretization of numeric attributes," in *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, pp. 388–391, doi: 10.1109/TAI.1995.479783.

BIOGRAPHIES OF AUTHORS

Sami Ghnimi is a senior faculty in engineering education and multidisciplinary research for more than 19 years. He is a Master and Ph.D. holder in electronics & telecommunications from the Faculty of Science of Tunis in Collaboration with Imperial College of London, UK and Higher Institute of Telecommunications of Paris, France. He has published several research papers in local, regional, and international refereed academic journals and has participated in several international conferences worldwide. His research interests include MIMO technology for Radar 4G/5G wireless communications, artificial intelligence & 4ir related technologies, pedagogy and innovation in engineering education. He can be contacted at email: sami.ghnimi@utas.edu.om.

Pratapa Raju Moola earned Ph.D. in electrical engineering from Jawaharlal Nehru Technological University, Hyderabad, India, in 2021. He is currently serving as a senior teaching faculty in the engineering department at the University of Technology and Applied Sciences (UTAS), Ibra, Oman. He has authored and published several research papers in national and international conferences and journals. His areas of specialization include AI applications in electrical and electronics engineering, data analytics in renewable energy, the internet of things (IoT), cloud and edge computing for dynamic prediction, and advancements in home energy management systems incorporating demand side management (DSM) and demand response (DR) concepts. He can be contacted at email: pratapa.moola@utas.edu.om.

Jamaludeen Abdul Shariff received Ph.D. degree specialization in optoelectronics-biomedical instrumentation from Bharathidasan University, Tamilnadu, India, in 2013. From 2002 to till now working as Asst. Professor at various institutions and continuing his research activity. His research interest includes the development of electronics instrument, audio and image processing techniques, AI and IoT embedded systems. He received research grants under various research programs such as URG (undergraduate research grant-2018) and RG (research grant-2021). He can be contacted at email: Jamaludeen.Shariff@utas.edu.om.