MACHINE LEARNING
Science and Technology

PURPOSE-LED PUBLISHING™ Part of the MACHINE LEARNING Series

PAPER • OPEN ACCESS

Lightweight ECG signal classification via linear law-based feature extraction

To cite this article: Péter Pósfay *et al* 2025 *Mach. Learn.: Sci. Technol.* **6** 035001

View the article online for updates and enhancements.

You may also like

- Photonic-digital hybrid artificial intelligence hardware architectures: at the interface of the real and virtual worlds
Lília M S Dias, Dinis O Abranches, Ana R Bastos et al.
- ICRH modelling of DTT in full power and reduced-field plasma scenarios using full wave codes
A Cardinali, C Castaldo, F Napoli et al.
- Thermal conductivity of suspended MBE-grown $PtSe_2$
Juliette Jolivet, Arkadiusz P Gertych, Eva Desgué et al.

This content was downloaded from IP address 146.110.64.23 on 08/01/2026 at 13:40

IOP Publishing Mach. Learn.: Sci. Technol. 6 (2025) 035001 https://doi.org/10.1088/2632-2153/ade6c3

MACHINE LEARNING
Science and Technology

CrossMark

OPEN ACCESS

RECEIVED
20 December 2024

REVISED
26 May 2025

ACCEPTED FOR PUBLICATION
20 June 2025

PUBLISHED
2 July 2025

Original Content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence.

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

PAPER

Lightweight ECG signal classification via linear law-based feature extraction

Péter Pósfay[1], **Marcell T Kurbucz**[2,*], **Péter Kovács**[3] and **Antal Jakovác**[1,4]

[1] Department of Computational Sciences, Institute for Particle and Nuclear Physics, HUN-REN Wigner Research Centre for Physics, 29-33 Konkoly-Thege Miklós Street, H-1121 Budapest, Hungary
[2] Institute for Global Prosperity, The Bartlett, University College London, 149 Tottenham Court Road, W1T 7NE London, United Kingdom
[3] Department of Numerical Analysis, Eötvös Loránd University, 1/c Pázmány Péter sétány, H-1117 Budapest, Hungary
[4] Department of Statistics, Institute of Data Analytics and Information Systems, Corvinus University of Budapest, 8 Fővám Square, H-1093 Budapest, Hungary
[*] Author to whom any correspondence should be addressed.

E-mail: m.kurbucz@ucl.ac.uk, posfay.peter@wigner.hun-ren.hu, kovika@inf.elte.hu and jakovac.antal@wigner.hun-ren.hu

Keywords: ECG classification, linear law, representation learning, anomaly detection, machine learning

Abstract

This paper introduces LLT-ECG, a novel semi-supervised method for electrocardiogram (ECG) signal classification that leverages principles from theoretical physics to generate features without relying on backpropagation or hyperparameter tuning. The method identifies linear laws that capture shared patterns within a reference class, enabling compact and verifiable representations of time series data. We evaluate the method on two PhysioNet datasets, TwoLeadECG and variable projection networks (VPNet). On TwoLeadECG, a minimal configuration—using only the linear law-based transformation (LLT) and a linear decision rule—reaches 73.1% accuracy using just two features. On VPNet, LLT-ECG combined with classifiers like k-nearest neighbors and support vector machines achieves up to 96.4% accuracy, comparable to deep learning models. These results highlight LLT-ECG's promise for lightweight, interpretable, and high-performing ECG classification.

1. Introduction

The success of machine learning systems hinges heavily on how data are represented and processed before being fed into models. The optimal data representation is often task-specific and may evolve as the problem or data changes. A significant portion of machine learning research has been devoted to improving data representations through feature engineering—manually transforming raw data into formats better suited for learning algorithms. While these handcrafted techniques can boost model performance, they have inherent limitations. Feature engineering often relies on human intuition, which can lead to suboptimal or non-generalizable representations. As the scale and complexity of data increase, these methods also become labor-intensive and less effective, highlighting the need for more automated approaches to data preprocessing and feature extraction (Bengio *et al* 2013).

To address the challenges posed by increasing data complexity and variety, modern machine learning has shifted towards methods that enable algorithms to learn data representations autonomously. This paradigm shift has yielded impressive advances across various fields, including speech (Dahl *et al* 2010, 2012) and image recognition (Krizhevsky *et al* 2012), where models are now capable of dynamically learning hierarchical representations from raw data. Hierarchical data organization has also led to breakthroughs in natural language processing (Bengio *et al* 2003, Collobert *et al* 2011), demonstrating its versatility across domains. Additionally, this concept has roots in neuroscience, particularly in visual information processing (Hubel and Wiesel 1959), where hierarchies play a crucial role in the brain's ability to interpret complex visual stimuli. This idea has inspired the design of convolutional neural networks (CNNs), which mimic the

© 2025 The Author(s). Published by IOP Publishing Ltd

hierarchical structure of the human visual cortex (Lecun *et al* 1998). CNNs have become a cornerstone of computer vision, demonstrating the power of data-driven representation learning.

Despite these successes, deep learning models such as CNNs are not without significant drawbacks. They often require large volumes of labeled data and substantial computational resources to train effectively. Their performance is sensitive to hyperparameter choices and prone to overfitting when data are limited (Kiymaç and Kaya 2023). Moreover, the inner workings of such models are typically opaque, which poses a challenge in domains where transparency and trust are critical—such as in healthcare (Hanin 2018, Kemker *et al* 2018). These limitations motivate the exploration of alternative, lightweight, and more transparent approaches to learning from time series data.

Electrocardiogram (ECG) signals, a key diagnostic tool in cardiology, present a particularly compelling use case for such methods. The classification of ECG signals is a well-established research area, with traditional approaches often relying on handcrafted features derived from domain knowledge (Clifford *et al* 2006, Mar *et al* 2011). These methods are often applied alongside preprocessing steps that address issues such as incompleteness and class imbalance in medical data, both of which pose additional challenges for classifier models (Hassler *et al* 2019). Although effective, these methods struggle to scale or adapt to variations between patients. On the other hand, deep learning-based techniques offer high performance but tend to require substantial amounts of annotated data, often lack transparency, and can be computationally expensive to train and deploy. Thus, there exists a gap in the literature: a need for feature extraction techniques that combine the efficiency and transparency of traditional methods with the adaptability and robustness of modern representation learning.

In this paper, we aim to bridge this gap by proposing LLT-ECG, a novel semi-supervised method for ECG classification that generates features without the need for backpropagation or hyperparameter tuning. LLT-ECG builds on the idea of linear laws—common, conserved relationships within a class of time series data—and draws inspiration from theoretical physics, particularly from principles such as renormalization. These laws allow us to extract compact, class-specific representations from raw time series, enabling a lightweight yet effective transformation of ECG signals into a feature space that is often linearly separable. This transformation supports efficient and transparent classification using even basic models.

Our method is particularly well-suited for real-world medical applications, where data may be limited, transparency is essential, and computational efficiency is critical. Unlike prior work on the linear law-based transformation (LLT), which focused on general-purpose time series classification (Kurbucz *et al* 2022, 2024), the current study adapts and extends the method to the unique structure of ECG data. This includes accounting for the periodic and spike-like nature of heartbeats, as well as evaluating performance across benchmark datasets with varying characteristics.

The main contributions of this paper are as follows:

- We introduce LLT-ECG, a novel method that integrates ECG signal preprocessing with semi-supervised feature extraction based on linear laws—approximate, class-specific linear relationships that persist across time-embedded segments. Inspired by principles from physics, LLT-ECG captures underlying structure in ECG signals, enabling interpretable and accurate classification.
- We introduce a hyperparameter-free, forward-only transformation that avoids backpropagation, gradient-based optimization, or large-scale parameter tuning. This design results in a lightweight and interpretable representation that is especially suitable for healthcare and edge-device applications.
- Unlike black-box deep learning models, LLT-ECG generates mathematically transparent features that reflect class similarity, allowing for traceable decisions and reducing the reliance on large labeled datasets.
- We demonstrate that the LLT transformation typically maps raw ECG signals into nearly linearly separable spaces, enabling effective classification using simple linear models—thereby highlighting the method's practical efficiency without sacrificing performance.

The remainder of this paper is structured as follows. Section 2 provides an overview of related work in ECG signal classification and feature extraction. Section 3 presents the proposed LLT-ECG classification pipeline, including its mathematical foundations, feature extraction process, and its application to multiclass and anomaly detection scenarios. Section 4 describes the datasets used for evaluation. The experimental results and performance comparisons are presented in section 5, followed by a detailed discussion in section 6. Finally, section 7 concludes the paper and outlines directions for future work.

2. Related work

2.1. ECG classification

Computer-assisted ECG analysis has a long history (Luz *et al* 2016) and remains an active area of research, driven by the increasing volume of data generated by modern technologies. Advances in data acquisition tools—such as the Internet of things, wearable devices, and smart technologies (Serhani *et al* 2020)—have facilitated commercial applications in ECG monitoring. These innovations have resulted in the generation of vast datasets that necessitate efficient storage, processing, and interpretation. Key challenges within this field include data labeling for training purposes and the extraction of pertinent features that encapsulate medical information while mitigating noise interference. Data labeling is typically labor-intensive and costly, thereby constraining the availability of training datasets. To address this limitation, self-supervised learning methodologies have been developed (Rafiei *et al* 2022). Concurrently, feature extraction remains a complex endeavor, achievable either through manual feature engineering or via automated representation learning techniques (Kiranyaz *et al* 2016, Kovács *et al* 2022).

2.2. Feature extraction in clinical settings

In clinical settings, decision-making processes must be transparent, often leading to a preference for handcrafted features that combine medical expertise with signal processing knowledge. Temporal and statistical features—such as RR intervals, ECG durations, and moment-based indices—are widely studied in the literature (Mar *et al* 2011, Luz *et al* 2016). Additionally, morphological features related to the shapes of key ECG waveforms (including the QRS complex, T-wave, and P-wave) are critical, especially for detecting arrhythmias, which is the primary focus of this section (Clifford *et al* 2006).

2.3. Morphological and frequency-based feature extraction

Several methodologies exist for extracting morphological features from ECG signals. Time-domain approaches operate directly on ECG signal samples (de Chazal *et al* 2004), calculating metrics such as power, derivatives, and extreme values (Jekova *et al* 2008, Letz *et al* 2023). However, these features are often susceptible to noise, prompting a frequent transition to frequency-domain feature extraction. Spectral methods, such as linear filtering, presume weak stationarity of the signal—a condition frequently unmet in ECG data due to physiological variabilities like respiration, body movement, and arrhythmia. To accommodate the nonstationary nature of ECG signals, joint time–frequency representations have been proposed, including the short-time Fourier transform, Choi–Williams distribution, and multiwindow spectrogram (Cakrak and Loughlin 2001). Unlike these techniques, wavelet transformations employ time windows of varying widths, thereby enhancing temporal and spectral localization of nonstationary features. This flexibility has rendered wavelets a preferred tool for ECG signal processing (Addison 2005).

2.4. Adaptive data-driven transformations

Despite the availability of diverse joint time–frequency representations, the foundational components—such as window functions or mother wavelets—are typically predetermined, which can be restrictive in ECG analysis where morphological features exhibit temporal and inter-individual variability. To overcome this limitation, adaptive data-driven transformations have been introduced. These approaches represent ECG signals using a series of basis functions optimized according to specific criteria. Techniques like variable projections with parameterized basis functions (e.g. Hermite, spline, and rational functions) offer optimal representations in a least-squares sense (Bognár and Fridli 2020, Kovács *et al* 2020). In contrast, principal component analysis (PCA) transforms data by identifying orthogonal directions that maximize variance (Castells *et al* 2007), while independent component analysis (ICA) seeks to maximize higher-order statistics, such as kurtosis, to achieve blind source separation (Chawla 2009). Linear transformations, including PCA and ICA, which map input features into well-separated spaces, can also be derived from data-driven learning. For instance, the neural dynamic classification algorithm (Rafiei and Adeli 2017) constructs a feature map that minimizes intra-class variance while maximizing inter-class variance.

3. Methodology

This section introduces the LLT-ECG methodology, which integrates time-series embedding, the construction of linear laws, and feature transformation into a unified classification framework. We begin with an overview of the classification pipeline, then introduce the mathematical formulation of the approach, and finally describe how it supports both multiclass classification and anomaly detection.

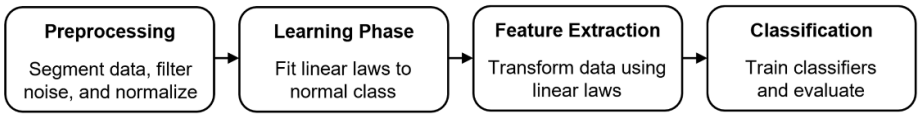


Figure 1. LLT-ECG pipeline.

3.1. Overview of the LLT-ECG pipeline

The proposed ECG classification method, LLT-ECG, consists of the following four stages, as summarized in figure 1:

1. **Preprocessing**: the dataset is segmented into various subsets based on lead differentiation and beat classification scenarios, followed by standard noise filtering and data normalization procedures.
2. **Learning phase**: linear laws are fitted to the normal class within the training set, capturing the underlying factors that characterize normal ECG patterns in the dataset.
3. **Feature extraction**: the established linear laws are utilized to transform the data into a new representation, thereby enhancing features pertinent to classification. This step builds upon previous work involving linear laws (Jakovác *et al* 2022, Kurbucz *et al* 2022) and introduces the LLT method, specifically tailored to accommodate the spike-like characteristics of ECG signals.
4. **Classification**: classifiers are trained within the transformed feature space, and their performance is subsequently evaluated using the test set.

This structured pipeline integrates the LLT algorithm with robust classifiers, enabling accurate and lightweight ECG analysis. Furthermore, this approach offers potential for developing transparent AI methods for ECG classification (see proposed future work in section 7).

3.2. Mathematical formulation of linear laws

We now present the mathematical foundations necessary for understanding linear laws (Jakovac *et al* 2020, Jakovac 2021, Jakovác *et al* 2022, Kurbucz *et al* 2022) and how they are used in the LLT-ECG transformation for ECG signal classification.

Consider a time series $y : \mathbb{R} \to \mathbb{V}$, where $\mathbb{V}$ is a finite-dimensional Hilbert space. In practice, we work with finite-dimensional representations; thus, we assume that a faithful finite-dimensional representation of the time series is provided. Consequently, we can construct a finite set of n-length samples from this time series:

$$\mathcal{Y} = \left\{ Y^{(k)} \in \mathbb{V}^{n+1} \;\middle|\; Y_i^{(k)} = y(t_k - i\Delta t)\,,\; i \in \{0,\ldots,n\}\,,\; k \in \{n,\ldots,K\} \right\}, \tag{1}$$

where $K, n \in \mathbb{N}$, Δt is the sampling interval of the time series, and $y(t_k)$ denotes the value of the time series at time t_k. This construction is known as the time delay embedding of the time series, which is sufficient to capture the dynamic state of the system (Packard *et al* 1980, Takens 1981). The selection of t_k values that serve as base points for the n-length samples can be adapted to suit specific applications. In this work, we generate maximally overlapping n-length samples for a time series of length L. To achieve this, we consider mappings of the following form:

$$\mathcal{F} : \mathbb{V}^{n+1} \to \mathbb{R}, \quad \mathcal{F}\left(Y^{(k)}\right) = 0, \quad \forall k. \tag{2}$$

In this study, we focus exclusively on linear mappings. This assumption constrains the form of $\mathcal{F}$ in equation (2). For convenience, we introduce matrix notation for the embedded time series $Y_i^{(k)}$ from equation (1):

$$Y_{ki} = Y_i^{(k)} = y(t_k - i\Delta t)\,. \tag{3}$$

Using this notation, the linear mapping $\mathcal{F}$ can be expressed as:

$$\mathcal{F}\left(Y^{(k)}\right) = \sum_{i=0}^{n} Y_{ki} w_i \equiv (Y\mathbf{w})_k = 0, \quad \forall k, \tag{4}$$

where $\mathbf{w}$ is a weight vector of length $n+1$. We refer to this construction as a 'linear law' $\mathcal{F}$, represented by the vector $\mathbf{w}$. As mentioned in the introduction, the intuition behind this nomenclature originates from physics:

$\mathcal{F}$ can be considered a 'law' on the set Y because it satisfies equation (4). This represents an ideal scenario that cannot be achieved empirically due to factors such as noise. Therefore, we utilize linear mappings that satisfy equation (4) with a random quantity ξ instead of zero, where $\langle \xi \rangle = 0$. We refer to these linear mappings as 'laws'.

3.3. Determining linear laws from data

To determine the coefficients w_i of the linear law, we express equation (4) as $\|Yw\| = 0$. Using the standard quadratic norm, this becomes:

$$\|Yw\|^2 = \frac{1}{K}(Yw)^{\mathrm{T}}(Yw) = w^{\mathrm{T}}Cw = 0, \tag{5}$$

where

$$C = \frac{1}{K}Y^{\mathrm{T}}Y, \tag{6}$$

is the correlation matrix of the dataset.

To avoid the trivial solution $w = 0$, we impose $\|w\| = 1$, transforming the problem into a constrained minimization task. Using Lagrange multipliers, we define:

$$\chi^2(\lambda) = w^{\mathrm{T}}Cw - \lambda\left(w^{\mathrm{T}}w - 1\right) \to \text{minimize}, \tag{7}$$

where λ is the Lagrange multiplier. The solution satisfies the eigenvalue equation

$$Cw^{(\lambda)} = \lambda w^{(\lambda)}, \tag{8}$$

yielding $n+1$ eigenvectors with corresponding eigenvalues λ. To select the optimal eigenvector, consider that an exact linear law satisfies $\|Yw\| = 0$. However, due to noise, the laws yield a non-zero quantity:

$$\mathcal{F}\left(Y^{(k)}\right) = \sum_{i=0}^{n} Y_{ki}w_i \equiv \xi_k, \tag{9}$$

where $\langle \xi \rangle = 0$. Substituting into equation (5) gives:

$$\|Yw\|^2 = \frac{1}{K}\sum_{k=1}^{K}\xi_k^2 = \left\langle \xi^2 \right\rangle. \tag{10}$$

Meanwhile, from equation (8), we have

$$\|Yw\|^2 = \lambda w^{\mathrm{T}}w = \lambda. \tag{11}$$

Comparing equations (10) and (11), the variance $\left\langle \xi^2 \right\rangle$ equals λ. Therefore, the optimal linear law corresponds to the eigenvector with the smallest eigenvalue, minimizing the variance and approximating $\|Yw\| = 0$. This eigenvector is guaranteed to exist as C is symmetric and positive definite.

The linear law is similar to the PCA method, but instead of the largest eigenvalue, we select the smallest. Thus, w_i is orthogonal to the dataset, indicating minimal variation in the direction of w. This identifies a common property, analogous to the normal vector of a hyperplane fitting the data.

In practice, the laws are never exact due to variations such as noise in the data. Consequently, the values of the optimal laws fall within a narrow range around zero for each respective class. When used to generate features, this results in class elements clustering near zero, while non-class elements are positioned farther from zero. This property provides a natural framework for classification algorithms.

Our formulation assumes a long time series sample, embedded into the matrix Y_{ki}. In many cases, however, the learning set consists of labeled time series samples $y_m(t)$, where $m \in \mathbb{N}$. This situation can be reduced to the previous case. By applying the embedding process to each sample, we generate matrices $Y^{(m)}_{k'i'}$, which can be concatenated along their first axis (rows) to form a compound matrix Y_{ki}. Since the rows represent time-embedded samples and their order is irrelevant for calculating the linear law, this concatenation simply increases the learning set.

In standard notation, this augmentation forms a larger block matrix with extended rows. If each sample has a uniform length L and the embedding depth is I, the matrix $Y^{(m)}_{k'i'}$ has dimensions $(L-I+1)\times I$. For M samples, concatenating them yields a matrix Y_{ki} with dimensions $((L-I+1)\cdot M)\times I$:

$$Y_{ki} = \begin{bmatrix} Y^{(1)}_{k'i'} \\ Y^{(2)}_{k'i'} \\ \vdots \\ Y^{(m)}_{k'i'} \end{bmatrix}. \tag{12}$$

This augmented matrix can be treated as Y_{ki} in the previous formulation. Thus, the linear law determination process remains consistent, independent of the dataset's structure, as all samples are embedded into a unified matrix.

3.4. Feature extraction and classification

After deriving the linear laws from the training data, we now explain how they are used to extract features and perform classification. In particular, this section describes how linear laws transform datasets and generate class-specific feature representations, supporting both multiclass and anomaly detection tasks.

We assume that linear laws encapsulate properties common to the elements of a defining set. This is formalized as a mapping from classes to linear laws:

$$\mathcal{H} : C_j \quad \rightarrow \quad w^{(j)}_i, \tag{13}$$

where C_j represents the sets corresponding to the classes in the dataset ($j \in \mathbb{N}$). The elements of a class can be transformed by linear laws as follows. Let $y_m(t)$ denote a sample from a given class, which is a time series. First, the time series is embedded in the standard manner, as described in equations (1) and (3). This maps the L-length time series sample into an $(L-n+1)\times n$ matrix $Y^{(m)}_{ki}$. Then, the linear law $w^{(j)}_i$ is applied to the sample, analogous to equation (9):

$$\sum_{i=0}^{n} Y^{(m)}_{ki} w^{(j)}_i = \xi^{(m,j)}_k, \tag{14}$$

where the vector $\xi^{(m,j)}_k$ contains the transformed features of sample m according to the linear law $w^{(j)}_i$, which corresponds to class C_j. As illustrated by equation (9), these features quantify how effectively a linear law can transform the subsamples of a given sample to zero. The closer the transformed subsamples are to zero, the better the linear law describes the sample. Intuitively, the linear law $w^{(j)}_i$ transforms elements of class C_j closer to zero than samples from other classes. Consequently, the features resulting from the transformation with $w^{(j)}_i$ act as similarity detectors for elements of class C_j.

3.4.1. Multiclass classification

The LLT features defined in equation (14) can be utilized for multiclass classification as follows. To assess a sample's similarity to each class, it must be transformed by the linear laws of all classes. Conceptually, when transforming an unknown sample, it is necessary to apply all linear laws corresponding to the possible classes to facilitate comparison. Using this transformed information, a classifier can be trained to recognize how elements of each class appear after being transformed by the linear laws of other classes. If the linear laws of different classes produce distinct transformations, the resulting feature vectors will be significantly different, especially when using the correct class's linear law. This separation in the abstract feature space enhances the performance of classifier algorithms by increasing the distance between samples from different classes.

Let the jth class (and its corresponding linear law, as per equation (13)) be indexed by j. Then, the transformed features for a sample $y_m(t)$ can be organized into a feature vector as follows:

$$\boldsymbol{\xi}^m = \left[\xi^{(m,1)}_k, \xi^{(m,2)}_k, \ldots, \xi^{(m,J)}_k\right]. \tag{15}$$

This vector comprises the feature vectors corresponding to each class[5]. Depending on the application, this feature vector can be processed differently. One approach is to flatten $\boldsymbol{\xi}^m$ by concatenating the individual

[5] The feature generation approach in the LLT-ECG method presented in this paper differs from the original LLT algorithm (Kurbucz *et al* 2022). In this method, we segment the signals to detect deviations from healthy patterns, whereas the original LLT analyzes the entire series using statistical measures such as mean and variance.

$\xi_k^{(m,j)}$ sequentially. Generally, this results in a feature vector of length $(L-n+1)\cdot J$, where J is the number of classes and n is the length of the linear law. In practice, this feature vector can be downsampled if a less detailed representation suffices for classification.

3.4.2. Application to anomaly detection

In binary classification scenarios ($J=2$), the feature vector $\boldsymbol{\xi}$ can be significantly simplified. It suffices to use the linear law corresponding to the reference class:

$$\boldsymbol{\xi}^m = \xi_k^{(m,1)}. \tag{16}$$

This simplification is particularly advantageous for anomaly detection, where a well-defined reference class is provided, and the objective is to distinguish samples from this class from others that may not belong to any predefined class. By using only the reference class's linear law, the classifier focuses on determining whether a given sample resembles the reference class. This approach can also be considered a form of outlier detection, as it does not require a linear law for the outliers, which may not be well-defined or existent. Nonetheless, some outlier samples are necessary to train the classifier after the LLT transformation.

For example, in classifying ECG signals between normal and ectopic types, numerous ectopic variations exist, but the goal is not to classify these variations individually. Instead, the task is to identify whether an ECG signal is healthy. Normal heartbeats are selected as the reference class, and ectopic signals are classified as 'not normal.' Each sample is transformed using the linear law derived from normal heartbeats, and the resulting features quantify the similarity of a sample to normal ECG signals. This constitutes a semi-supervised feature learning approach in which only a subset of the data, specifically the reference class samples, needs to be labeled.

In our experiments, the linear laws were derived exclusively from samples labeled as normal (healthy) heartbeats in the training set. Then these laws were used to transform all ECG signals, including ectopic samples, into the LLT feature space. This design enables LLT-ECG to function as a semi-supervised method, since labeled data are required only for the reference class during feature construction. Note that this is advantageous in clinical settings, where the majority class primarily consists of normal heartbeats, which are much easier to collect than rare abnormalities. This makes the proposed framework well-suited for personalized training scenarios, where just a few seconds of data can be used to adapt the model to an individual patient.

4. Employed datasets

In this study, we utilized two datasets: TwoLeadECG (Keogh 2019) and variable projection networks (VPNet) (Kovács *et al* 2022). Both datasets are derived from the PhysioNet MIT-BIH Arrhythmia Database (Moody and Mark 2001).

4.1. TwoLeadECG dataset

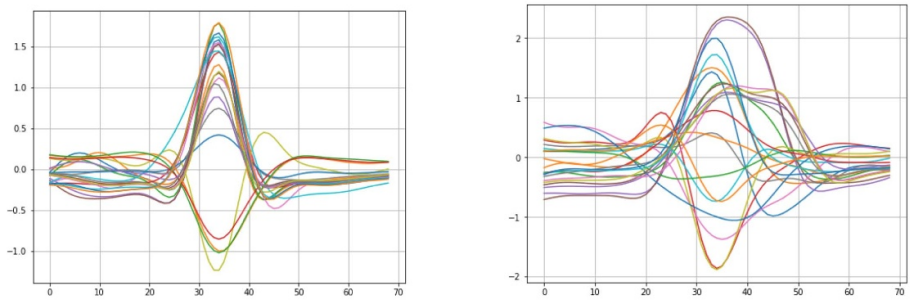
To demonstrate the transformation capabilities of the LLT algorithm in the ECG context—as applied by the proposed LLT-ECG approach—we employed the TwoLeadECG dataset, a subset of the MIT-BIH Long-Term ECG database (Moody and Mark 2001). This dataset comprises long-term recordings from the same patient using two different leads. It contains a total of 1162 samples, with 581 labeled as Class 1 and 581 labeled as Class 2. Each sample consists of 82 consecutive measurements. The classification task involves distinguishing between signals originating from each lead. Following the approach of Harada *et al* (2019), we randomly divided the dataset into training and test sets, resulting in 523 signals for training and 639 signals for testing.

4.2. VPNet dataset

For training and testing purposes, we utilized the VPNet dataset (Kovács *et al* 2022), which consists of QRS complexes. This balanced subset of the MIT-BIH database (Moody and Mark 2001) includes only healthy and ectopic beats. The training-testing split adheres to the protocol defined by de Chazal *et al* (2004), ensuring that samples from the same patients are not present in both the training and testing sets. This separation guarantees that the results generalize well to new samples. The training set comprises 8520 samples, while the test set contains 6440 samples. Examples from the dataset are illustrated in figure 2.

For this study, the VPNet dataset was further divided into three parts: 40% of the training set was allocated for training, and the remaining 60% was designated as a validation set, primarily used to tune the hyperparameters of various classifiers. The original test set remained unchanged and was solely used to evaluate classification accuracy.

Samples were processed using standard procedures (Clifford *et al* 2006, Joshi *et al* 2013, Gospodinov *et al* 2019). Initially, low-pass and high-pass filters with cutoff frequencies of 20 Hz and 0.5 Hz, respectively, were



(a) Healthy QRS Complexes (b) Ectopic QRS Complexes

Figure 2. Examples from the learning set.

applied to remove noise and baseline shifts. Subsequently, the signals were standardized to have a zero mean and normalized by their maximum value. The QRS peaks were identified, and each sample was segmented to include 30 data points centered on the QRS complex peak, corresponding to $L = 30$ in equation (12).

Although LLT-ECG constructs its features using only the annotated normal class, all classifiers (including LLT-ECG) were trained using the full labeled training set to ensure a fair comparison, since the baseline methods cannot operate under partial labeling.

5. Experimental results

5.1. Evaluating LLT transformation using the TwoLeadECG dataset

This section provides an in-depth demonstration of how LLT-ECG transforms the input feature space to support classification. We use the TwoLeadECG dataset (Keogh 2019), a binary classification task derived from the MIT-BIH Arrhythmia Database (MIT-BIH 2005). The dataset includes 523 training and 629 test signals, each comprising 82 consecutive samples. The objective is to distinguish between signals from two different ECG leads recorded from the same patient[6].

Rather than applying end-to-end training, we evaluate the discriminative power of the LLT transformation itself. Specifically, we apply LLT to the test set and assess how well the transformed features separate the two classes using a simple linear decision rule. This analysis highlights the transformation's ability to yield linearly separable representations.

5.1.1. LLT-based feature transformation and evaluation protocol

We construct linear laws using a rolling window of length 11 applied to the test signals[7]. Each 82-length signal yields 72 embedded subsequences, generating a total of 45 288 linear laws across the test set. These laws are grouped into two sets based on class labels.

To increase the classification challenge and simulate streaming data, we randomly selected a single 11-length series from each test signal. We then took the first test signal, applied a 6th-order time-delay embedding, and computed its product with both law matrices. The results were squared to ensure all values were non-negative[8]. We computed the column means of the squared matrices and stored 25 percentiles (p_c^o, where $o \in \{1, 2, \ldots, 25\}$) for each class c.[9] Finally, we evaluated the accuracy of each percentile pair as a linear separator between the two classes ($c \in \{1, 2\}$) using the following equation:

$$\begin{aligned}&(\text{Count}\,(\,p_1^o \leqslant p_2^o * \alpha \mid c = 1\,) + \\ &\text{Count}\,(\,p_1^o > p_2^o * \alpha \mid c = 2\,))\;/\;523,\end{aligned} \tag{17}$$

where $\alpha \in 0.01, 0.02, \ldots, 2.00$, and N is the total number of signals (in this case, $N = 629$). The accuracy of each percentile pair is depicted in figure 3.

As shown in figure 3, the 16th percentiles (p_1^{16} and p_2^{16}) achieved the best separation, reaching a total accuracy of 73.07% with $\alpha = 0.97$. The discriminative power of these two features is illustrated in figure 4.

[6] The complete experimental configuration for both datasets, including embedding details and classifier settings, is provided in appendix.

[7] This corresponds to a 6th-order time-delay embedding. The resulting embedded matrix is symmetric, with the latest sample at the final diagonal position.

[8] These calculations were performed using the LLT R package (Kurbucz *et al* 2024).

[9] The column means represent the 'goodness of fit' of the laws; lower percentiles correspond to better-fitting laws.

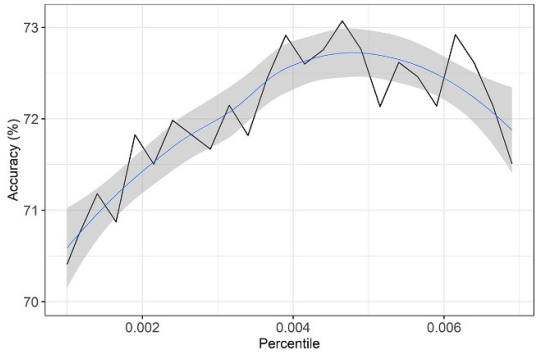


Figure 3. Accuracy of each percentile pair in linear discrimination.

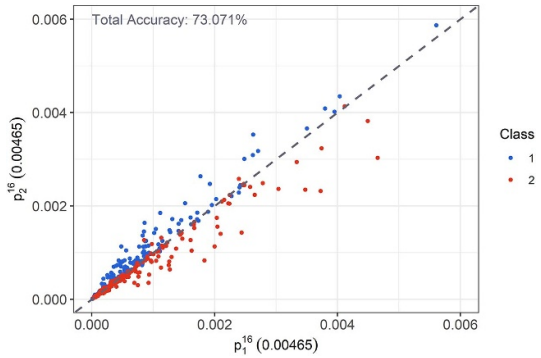


Figure 4. Discriminative power of the best percentile pair.

5.1.2. Comparison to benchmark classifiers

For benchmarking, classifiers were applied to the raw signals—optionally transformed using standard dimensionality reduction techniques—without the LLT transformation. We trained 34 variations of 11 standard classifiers, both with and without dimensionality reduction techniques such as PCA, kernel PCA (Schölkopf *et al* 1998), uniform manifold approximation and projection (UMAP) (McInnes *et al* 2018), and t-distributed stochastic neighbor embedding (t-SNE) (van der Maaten and Hinton 2008), on the test set[10]. Table 1 summarizes their average and maximum accuracies.

5.2. Classification results on the VPNet dataset

Here, we present the results of the LLT-ECG method compared to both simple learning methods—such as random forests (RFs) (Ho 1995), k-nearest neighbors (KNNs) (Altman 1992), support vector machines (SVMs) (Cortes and Vapnik 1995), and a basic NN (McCulloch and Pitts 1943, Lippmann 1987, Abadi *et al* 2015)—as well as more complex classifiers, including deep NNs (DNNs).

5.2.1. Simple learning methods

The performance of LLT-ECG-based classification methods is summarized in table 2.[11]

Standard performance metrics were used to evaluate the classifiers. Total accuracy (ACC) is defined as:

$$\mathbf{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}}, \tag{18}$$

[10] Each classifier was evaluated using 10-fold cross-validation. The KPCA, UMAP, and t-SNE transformations were performed using the `kernlab`, `umap`, and `Rtsne` R packages, respectively. t-SNE yielded 2 components, while the other methods provided 5 components.

[11] To confirm the class-specificity of LLT features, we computed statistical descriptors (e.g. mean, median, standard deviation, minimum, maximum, interquartile range) for each sample and applied the Wilcoxon test (3878 healthy vs 4250 ectopic). Most metrics showed significant differences ($p < 10^{-50}$), except for skewness and kurtosis ($p = 0.75$, $p = 0.79$).

Table 1. Benchmark classification accuracies on the TwoLeadECG test set. LLT with linear discriminator achieves 73.07% accuracy using only two features (16th percentiles).

Classifier	Num. of ver.	Versions (best is italics)	Mean acc. (%)	Max. acc. (%)
Binary GLM logistic regression	1		49.452	49.452
Binary GLM logistic regression (PCA)	1		52.895	52.895
Binary GLM logistic regression (KPCA)	1		52.895	52.895
Binary GLM logistic regression (UMAP)	1		50.704	50.704
Binary GLM logistic regression (t-SNE)	1		55.399	55.399
Discriminant	2	Linear, *quadratic*	51.487	53.678
Discriminant (PCA)	2	*Linear*, *quadratic*	52.895	52.895
Discriminant (KPCA)	2	Linear, *quadratic*	53.521	54.304
Discriminant (UMAP)	2	Linear, *quadratic*	53.834	56.964
Discriminant (t-SNE)	2	Linear, *quadratic*	55.086	55.399
Efficient linear SVM	1		48.357	48.357
Efficient linear SVM (PCA)	1		50.861	50.861
Efficient linear SVM (KPCA)	1		53.678	53.678
Efficient linear SVM (UMAP)	1		51.017	51.017
Efficient linear SVM (t-SNE)	1		53.678	53.678
Efficient logistic regression	1		48.357	48.357
Efficient logistic regression (PCA)	1		52.895	52.895
Efficient logistic regression (KPCA)	1		53.365	53.365
Efficient logistic regression (UMAP)	1		50.704	50.704
Efficient logistic regression (t-SNE)	1		55.399	55.399
Ensemble	5	Bagged-, boosted-, RUSboosted trees, *subs. KNN*, -disc.	65.509	**78.091**
Ensemble (PCA)	5	*Bagged*-, boosted-, RUSboosted trees, subs. KNN, -disc.	59.906	67.919
Ensemble (KPCA)	5	*Bagged*-, boosted-, RUSboosted trees, subs. KNN, -disc.	62.754	68.075
Ensemble (UMAP)	5	*Bagged*-, boosted-, RUSboosted trees, subs. KNN, -disc.	61.565	70.110
Ensemble (t-SNE)	5	*Bagged*-, boosted-, RUSboosted trees, subs. KNN, -disc.	61.064	69.640
Kernel	2	Logistic regression, *SVM*	69.405	71.049
Kernel (PCA)	2	Logistic regression, *SVM*	62.363	62.441
Kernel (KPCA)	2	*Logistic regression*, SVM	62.676	63.850
Kernel (UMAP)	2	Logistic regression, *SVM*	65.102	65.101
Kernel (t-SNE)	2	*Logistic regression*, SVM	54.851	56.182
KNN	6	Fine, medium, coarse, cosine, cubic, *weighted*	69.014	**78.404**
KNN (PCA)	6	Fine, medium, coarse, cosine, *cubic*, weighted	63.589	69.484
KNN (KPCA)	6	Fine, medium, coarse, cosine, cubic, *weighted*	63.250	66.510
KNN (UMAP)	6	Fine, medium, coarse, cosine, cubic, *weighted*	63.485	67.762
KNN (t-SNE)	6	*Fine*, medium, coarse, cosine, cubic, weighted	65.023	71.987
Naive Bayes	2	Gaussian, *kernel*	55.008	56.964
Naive Bayes (PCA)	2	Gaussian, *kernel*	54.617	56.495
Naive Bayes (KPCA)	2	Gaussian, *kernel*	55.947	58.842
Naive Bayes (UMAP)	2	Gaussian, *kernel*	53.443	53.521
Naive Bayes (t-SNE)	2	*Gaussian*, kernel	54.773	55.086
NN	5	Narrow, medium, *wide*, bilayered, trilayered	**74.773**	**77.465**
NN (PCA)	5	Narrow, medium, *wide*, bilayered, trilayered	71.487	**73.396**
NN (KPCA)	5	Narrow, *medium*, wide, bilayered, trilayered	71.362	**74.961**
NN (UMAP)	5	Narrow, medium, *wide*, bilayered, trilayered	68.263	71.362
NN (t-SNE)	5	Narrow, medium, *wide*, bilayered, trilayered	71.424	**75.743**

(Continued.)

Table 1. (Continued.)

Classifier	Num. of ver.	Versions (best is italics)	Mean acc. (%)	Max. acc. (%)
SVM	6	Linear, quadratic, cubic, *fine*-, medium-, coarse Gaussian	58.764	**79.030**
SVM (PCA)	6	Linear, quadratic, cubic, *fine*-, medium-, coarse Gaussian	53.365	65.102
SVM (KPCA)	6	Linear, quadratic, cubic, *fine*-, medium-, coarse Gaussian	57.512	69.640
SVM (UMAP)	6	Linear, quadratic, cubic, *fine*-, medium-, coarse Gaussian	53.469	60.720
SVM (t-SNE)	6	Linear, quadratic, cubic, *fine*-, medium-, coarse Gaussian	53.599	65.571
Tree	3	*Fine*, medium, coarse	61.189	68.232
Tree (PCA)	3	*Fine*, medium, coarse	61.189	65.884
Tree (KPCA)	3	*Fine*, medium, coarse	61.242	67.293
Tree (UMAP)	3	*Fine*, medium, coarse	61.607	69.953
Tree (t-SNE)	3	*Fine*, medium, coarse	62.493	67.293

where TP, TN, FP, and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively. Sensitivity (Se), also known as recall, is defined as:

$$\mathbf{Se} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{19}$$

Precision (+P), or positive predictive value, is defined as:

$$\mathbf{+P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}. \tag{20}$$

These performance metrics were calculated for both the validation and test sets to examine how well the LLT-generated features generalize to a new group of patients. The validation set contains ECG signals from the same patients as the training set, whereas the test set comprises signals from different patients than those in the training and validation sets. These results characterize the real-world performance of the proposed method on new, previously unseen data. As observed in table 2, the metrics are similar for both sets, falling within the range of state-of-the-art methods (Luz *et al* 2016).

5.2.2. DNNs

Table 3 summarizes the classification accuracy, specificity (Sp), and sensitivity (Se) for state-of-the-art ECG classification methods.

The first group of methods (CNN, SNN, SCNN, and VPNet) was trained on the balanced VPNet dataset, as detailed in section 4.2, and underwent an extensive hyperparameter search covering the learning rate, optimizer, number of hidden neurons, and activation function. Consequently, table 3 reports the performance of the models with their best-performing configurations.

In contrast, the state-of-the-art DNN architectures in the second group were trained on an imbalanced version of the VPNet dataset, where the majority class (normal) was not downsampled to match the number of ectopic samples. Both groups followed the same training-test split protocol as used in the balanced dataset version (de Chazal *et al* 2004), ensuring the results in tables 2 and 3 are comparable.

The bottom row of table 3 presents the corresponding inter-patient performance ranges of state-of-the-art methods, derived from the entries in table 5 of Qiao *et al* (2023), which were trained and tested on the DS1/DS2 subsets of the MIT-BIH Arrhythmia Dataset (Goldberger *et al* 2000). Based on Qiao *et al* (2023), we also report the performance metrics of specific deep learning models for which the number of parameters was either published or could be calculated. These include DNNs with three and six fully connected hidden layers, as well as variations of DenseNet, ResNet, CNN, LSTM, and an adaptation of the VGGNet architecture (Simonyan and Zisserman 2015) known as O-WCNN. In addition, we have indicated the top-performing LLT-ECG classifier model from table 2.

6. Discussion

6.1. Results on the TwoLeadECG dataset

The benchmark classifiers presented in table 1 explored a broad spectrum of machine learning paradigms—including logistic regression, discriminant analysis, SVMs, decision trees, ensemble methods,

Table 2. LLT-ECG-based classification performance of simple methods evaluated on the VPNet dataset (Kovács *et al* 2022). The general performance ranges of state-of-the-art (sota) methods, which are not necessarily trained on the same dataset, are also presented in the last two rows based on da S. Luz *et al* (2016) and Ansari *et al* (2023). The highest value in each column is highlighted.

	Validation					Test				
		Normal		Ectopic			Normal		Ectopic	
Method	Total accuracy	Se	+P	Se	+P	Total accuracy	Se	+P	Se	+P
RF	93.6%	94.3%	93.1%	93.0%	94.2%	92.1%	92.9%	91.4%	91.2%	92.8%
SVM (rbf)	95.0%	96.3%	93.8%	93.6%	96.2%	**94.3%**	94.4%	**94.2%**	**94.2%**	94.4%
SVM (linear)	89.4%	89.9%	89.0%	88.9%	89.8%	91.8%	93.2%	90.6%	90.4%	93.0%
NN	95.2%	95.7%	94.7%	94.7%	95.6%	93.1%	94.0%	92.3%	92.2%	93.9%
KNN ($k = 4$)	**96.4%**	**97.4%**	**95.5%**	**95.4%**	**97.4%**	91.5%	**95.0%**	88.8%	88.0%	**94.6%**
sota till 2016	N/A	N/A	N/A	N/A	N/A	N/A	80%–99%	85%–99%	77%–96%	63%–99%
sota 2017–2023	N/A	N/A	N/A	N/A	N/A	96%–99%	N/A	N/A	71%–100%	44%–98%

Table 3. Classification performances of DNNs on the test set.

Architecture	Number of parameters	Accuracy	Sp	Se
Proposed (LLT-ECG + SVM (rbf)	12	94.3%	94.4%	94.2%
CNN Kovács and Samiee (2022)	212 610	95.92%	96.09%	95.75%
SNN Kovács and Samiee (2022)	58 880	95.59%	93.73%	**97.45**%
SCNN Kovács and Samiee (2022)	376 704	95.42%	95.31%	95.53%
VPNet Kovács *et al* (2022)	39	96.65%	96.83%	96.61%
Raw data + DNN Xu *et al* (2019)	61 900	99.70%	99.89%	**97.68**%
6 features + DNN Jun *et al* (2016)	11 700	99.41%	N/A	96.08%
DenseNet + BiLSTM Gan *et al* (2021)	857 700	92.37%	94.51%	68.29%
Deep residual CNN Li *et al* (2022)	>2026	88.99%	94.75%	52.10%
O-WCNN Jangra *et al* (2023)	>52 mil	99.43%	99.69%	91.06%
State-of-the-art Qiao *et al* (2023)	N/A	88%–99%	88%–99%	52%–98%

KNNs, naïve Bayes, and shallow NNs—each with multiple configurations and hyperparameter settings. These configurations involved various kernel types (e.g. linear, RBF, polynomial), regularization strategies, distance metrics, and network architectures, contributing to the complexity of these models. Despite this, most benchmark configurations did not surpass the performance of the proposed approach, which applies the LLT transformation to ECG signals and classifies them using a simple linear decision rule, achieving an accuracy of 73.071% using only two features.

Only a small number of benchmark methods exceeded this accuracy: for example, the highest-performing SVM variant reached 79.030%, KNN peaked at 78.404%, and ensemble methods achieved up to 78.091%. However, these models required more complex architectures and hyperparameter tuning, often relying on all 11 original features or reduced representations obtained via dimensionality reduction (e.g. 2 components with t-SNE, 5 components for PCA, KPCA, and UMAP). In contrast, LLT-ECG achieved its results with a minimalist, transparent setup.

These findings suggest that the LLT transformation, as applied in LLT-ECG, successfully maps the original ECG signals into a nearly linearly separable feature space (see figure 4). This enables competitive classification performance using simple models, bypassing the need for deep learning architectures or complex pre-processing pipelines. Unlike approaches that require backpropagation and large-scale optimization, LLT-ECG is lightweight, transparent, and easy to audit—making it especially suitable for applications where transparency is critical.

6.2. Results on the VPNet dataset

In our second experiment (see table 2), we first applied several simple classifiers within the LLT-ECG framework and compared their performance to state-of-the-art methods. We then extended the comparison to include complex DNNs to evaluate the proposed method against advanced architectures.

6.2.1. Simple learning methods

According to the results presented in table 2, basic learning methods—such as RF, KNN, SVM, and NN—were effectively trained as part of the ECG-LLT framework. For the RF classifier, we balanced the number of estimators and tree depth to avoid overfitting. By selecting 10 estimators with trees of depth 6, we achieved comparable performance on both validation and test sets, indicating successful generalization.

The KNN classifier was optimized using the validation set, resulting in $k = 4$ with the Chebyshev metric. This configuration yielded the highest overall accuracy on the validation set, although its performance on the test set was less convincing, particularly in terms of sensitivity for ectopic beats. Since KNN relies on proximity to training samples, new patient data in the test set that are farther from training samples in the feature space led to reduced accuracy. In contrast, NN and SVM can better generalize due to their ability to learn more complex patterns.

The SVM classifier, particularly with a nonlinear kernel, is well-suited to the LLT-transformed feature space, effectively finding separating hypersurfaces between clusters. It provided high and consistent accuracy, exceeding 94% on both validation and test sets, demonstrating that the LLT-ECG successfully extracts meaningful features. Even a linear SVM achieved approximately 90% accuracy, suggesting that the proposed method maps samples into distinct regions that are linearly separable.

Furthermore, a simple NN with one hidden layer—an input layer of 24 neurons, a hidden layer of 8 neurons, and an output layer of 2 neurons—was sufficient to achieve good classification performance. The

NN generalized well, with similar accuracies on validation and test sets (table 2). This simplicity indicates that the LLT-ECG effectively captures the necessary features.

6.2.2. DNNs

The proposed LLT-ECG classification method typically achieves a test accuracy that is about 1%–2% lower than the first group of methods and approximately 5% lower than the top-performing DNN approaches. On the other hand, all of these methods require training via backpropagation, and most of them yield deep AI models with high complexity. In contrast, the proposed method operates without the need for error backpropagation, and the number of parameters, i.e. the length of the linear law (n), is equal to 11. Moreover, as presented in section 5.1, the LLT transformation extracts valuable features for classifying streaming data without the necessity for precise data alignment. This does not apply to its end-to-end DNN counterparts (Jun *et al* 2016, Xu *et al* 2019), which segment and align heartbeats using the characteristic points of ECG waveforms (such as QRS, T, P). For this reason, improper peak detection and beat alignment may lead to high variability in the feature vectors (Xu *et al* 2019), potentially degrading classification accuracy on noisy data. The proposed LLT transformation, however, provides more robust features with decent discrimination power.

6.3. Real-world applicability and limitations

The primary computational complexity in constructing linear laws is determined by the column dimension of the matrix Y_{ki} in equation (12). Specifically, the eigenvectors of the correlation matrix C in equation (6) can be computed using singular value decomposition, which requires $4mn^2 + 8n^2$ floating-point operations (flops) for an $m \times n$ matrix, where $m \geq n$—see table 8.6.1 in Golub and Van Loan (2013). It is important to note that the column dimension is typically small and fixed prior to training, as it is equal to 11 in our ECG case study. Consequently, linear laws can be efficiently updated with new data samples, unlike traditional DNNs that rely on backpropagation and necessitate retraining on the entire dataset. In practice, new time-embedded data samples only increase the row dimension of Y_{ki} in equation (12), without impacting the column dimension that primarily governs computational complexity. This characteristic renders LLT features well-suited for incremental learning scenarios, such as clinical applications, where substantial amounts of physiological data are continuously collected.

However, the LLT-ECG-based approach presents certain limitations within the context of ECG applications. Although it has demonstrated robustness against various time series distortions, including noise and amplitude scaling Kurbucz *et al* (2022), see, it remains sensitive to scaling along the time axis—a common occurrence in ECG signals (Keogh 1997). Another limitation pertains to the continuous filtration of laws obtained through streamed data classification. While employing a time-window approach can help maintain the method's speed and lightweight nature, ensuring an efficient algorithm that is free from catastrophic forgetting necessitates the inclusion of a law screening step following each classification.

Lastly, it is important to highlight that the proposed LLT features capture the similarity between the time embeddings of the input data and the extracted linear laws. In this regard, the LLT-ECG method embodies mathematical transparency, a form of interpretability (Lipton 2018). Nevertheless, the clinical explainability concerning how the linear laws correlate with medical features remains an open question. To elucidate such relationships, post-hoc explanation methods can be utilized to evaluate the relevance between input features and the model's output after training. For example, similar to the approach in Bender *et al* (2024), the average attribution per class could be computed for each linear law using the integrated gradients method (Sundararajan *et al* 2017).

7. Conclusions and future work

In this paper, we introduced a novel technique for learning features from time series data and successfully applied it to the task of binary ECG signal classification. This new method extracts features in a data-driven, forward manner, resulting in a classifier-agnostic feature space. These characteristics are achieved through the use of the principle of linear laws and the LLT method (Jakovác *et al* 2022, Kurbucz *et al* 2022). Linear laws are defined as common linear relationships among points in samples that belong to the same class. As a result, linear laws provide a concise and effective representation of classes. Moreover, the proposed LLT-ECG transformation offers a lightweight feature learning procedure that avoids the need for resource-intensive training via backpropagation, thereby eliminating the vanishing gradient problem. Additionally, the LLT-based approach demonstrated high data efficiency, achieving performance comparable to state-of-the-art methods while using less than half the training samples.

The proposed method provides the following key advantages:

- Semi-supervised feature learning that requires labeled data only from a reference class, making it highly practical in scenarios where annotated data is scarce.
- A fully forward and interpretable transformation that avoids backpropagation and hyperparameter tuning.
- Lightweight and computationally efficient processing, suitable for deployment on edge devices and real-time systems.
- Generation of nearly linearly separable feature spaces, enabling strong performance with simple classifiers.
- Broad generalizability, demonstrated on real-world ECG datasets from PhysioNet across different patients and signal types.

Future work on the LLT-ECG method could focus on two main research areas: developing techniques that build on the core ideas presented in this paper to enhance the medical interpretability of the classifications, and exploring self-selection mechanisms for the laws to ensure LLT-ECG remains up-to-date and lightweight, even in streaming data classification tasks. In addition to these research directions, we plan to release the LLT-ECG Python package to serve as a foundation for further development. Beyond ECG data, the original LLT algorithm has already shown success in various time series classification tasks, such as human activity recognition (Kurbucz *et al* 2022) and price movement prediction (Kurbucz *et al* 2023). In our future research, we plan to explore additional applications, including the classification of visually evoked potentials (Dózsa *et al* 2021) and tire sensor data (Dózsa *et al* 2022). Since these signals exhibit morphological characteristics similar to ECG data, we expect that our method will perform well in the corresponding classification and regression tasks.

Data availability statement

All the raw data generated in this study are available from the corresponding author upon reasonable request.

Acknowledgment

Project No. PD142593 was implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development, and Innovation Fund, financed under the PD_22 'OTKA' funding scheme. P K was supported by the ÚNKP-22-5 New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development and Innovation Fund. Project No. TKP2021-NVA-29 and K146721 have been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021-NVA and the K_23 'OTKA" funding schemes, respectively.

Conflict of interest

The authors declare no competing interests.

Author contributions

Péter Pósfay 0000-0002-6769-3302
Conceptualization (equal), Data curation (equal), Formal analysis (equal), Methodology (equal), Software (equal), Validation (equal), Visualization (equal), Writing – original draft (equal), Writing – review & editing (equal)

Marcell T Kurbucz 0000-0002-0121-6781
Conceptualization (equal), Data curation (equal), Formal analysis (equal), Methodology (equal), Software (equal), Validation (equal), Visualization (equal), Writing – original draft (equal), Writing – review & editing (equal)

Péter Kovács 0000-0002-0772-9721
Conceptualization (equal), Data curation (equal), Formal analysis (equal), Methodology (equal), Software (equal), Validation (equal), Visualization (equal), Writing – original draft (equal), Writing – review & editing (equal)

Antal Jakovác 0000-0002-7410-0093
Conceptualization (equal), Formal analysis (equal), Methodology (equal), Software (equal), Supervision (equal), Validation (equal), Writing – original draft (equal), Writing – review & editing (equal)

Appendix. Experimental configuration summary

This appendix summarizes the key dataset characteristics, feature extraction strategies, classification settings, and evaluation protocols of the two complementary experimental pipelines presented in this study.

Experiment 1: TwoLeadECG dataset

- **Dataset**: TwoLeadECG (a subset of the MIT-BIH Long-Term ECG Database; see section 4.1).
- **Feature extraction**: performed using the LLT R package (Kurbucz *et al* 2024). Each 82-sample signal was embedded using a 6th-order time-delay embedding (window length = 11), resulting in 72 embedded subsequences per signal.
- **Feature representation**: for each LLT-transformed signal, 25 percentiles of the squared residuals were calculated. The 16th percentile from each class was selected as the final two-feature representation for classification.
- **Benchmark classifiers**: classification was carried out using MATLAB's Classification Learner App. A total of 34 configurations across 11 classifier families were evaluated, including logistic regression, discriminant analysis (linear and quadratic), SVMs (linear, Gaussian, polynomial), decision trees, KNNs, naïve Bayes, ensemble methods (bagging, boosting, RUSBoost), and shallow NNs with various architectures.
- **Dimensionality reduction techniques**: PCA, KPCA, UMAP, and t-SNE were applied to the raw signals as baseline transformations. PCA, KPCA, and UMAP reduced each signal to 5 components; t-SNE yielded a two-dimensional representation.
- **Training procedure**: all classifiers were evaluated using 10-fold cross-validation with default hyperparameters as provided by MATLAB's Classification Learner App.

Experiment 2: VPNet dataset

- **Dataset**: VPNet (QRS complex subset from the MIT-BIH Arrhythmia Database; see section 4.2).
- **Preprocessing**: low-pass and high-pass filtering with cutoff frequencies of 20 Hz and 0.5 Hz, respectively. Zero-mean standardization, amplitude normalization, and QRS-centered beat segmentation (30 samples per beat).
- **Feature extraction**: performed using Python. A window size of 11 was used, resulting in a 20×11 embedding matrix for each sample.
- **Feature representation**: LLT features were derived using only the linear laws associated with the normal class, resulting in a feature vector for each sample that quantifies its similarity to normal ECG signals.
- **Benchmark classifiers**: four classification models were evaluated: KNNs ($k = 4$, Chebyshev distance), RF (10 estimators, maximum depth = 6), SVMs (linear and RBF kernels), and a shallow NN (one hidden layer with 8 ReLU neurons).
- **Training procedure**: the dataset was split into training (40%), validation (60%), and a held-out test set for evaluation.

References

Abadi M *et al* 2015 TensorFlow: large-scale machine learning on heterogeneous systems (available at: www.tensorflow.org/)
Addison P S 2005 Wavelet transforms and the ECG: a review *Physiol. Meas.* **26** 155–99
Altman N S 1992 An introduction to kernel and nearest-neighbor nonparametric regression *Am. Stat.* **46** 175–85
Ansari Y, Mourad O, Quarage K and Serpedin E 2023 Deep learning for ECG arrhythmia detection and classification: an overview of progress for period 2017–2023 *Front. Physiol.* **14** 1–20
Bender T, Beinecke J M, Krefting D, Müller C, Dathe H, Seidler T, Spicher N and Hauschild A-C 2024 Analysis of a deep learning model for 12-lead ECG classification reveals learned features similar to diagnostic criteria *IEEE J. Biomed. Health Inf.* **28** 1848–59
Bengio Y, Courville A and Vincent P 2013 Representation learning: a review and new perspectives *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 1798–828
Bengio Y, Ducharme R, Vincent P and Janvin C 2003 A neural probabilistic language model *J. Mach. Learn. Res.* **3** 1137–55
Bognár G and Fridli S 2020 ECG heartbeat classification by means of variable rational projection *Biomed. Signal Process. Control* **61** 102034
Cakrak F and Loughlin P J 2001 Multiwindow time-varying spectrum with instantaneous bandwidth and frequency constraints *IEEE Trans. Signal Process.* **49** 1656–66
Castells F, Laguna P, Sörnmo L, Bollmann A and Roig J M 2007 Principal component analysis in ECG signal processing *EURASIP J. Adv. Signal Process.* **2007** 1–21

Chawla M P S 2009 A comparative analysis of principal component and independent component techniques for electrocardiograms *Neural Comput. Appl.* **18** 539–56
Clifford G D, Azuaje F and McSharry P 2006 *Advanced Methods And Tools for ECG Data Analysis* (Artech House, Inc.)
Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K and Kuksa P 2011 Natural language processing (almost) from scratch *J. Mach. Learn. Res.* **12** 2493–537
Cortes C and Vapnik V 1995 Support-vector networks *Mach. Learn.* **20** 273–97
Dahl G E, Ranzato M, Mohamed A-r and Hinton G 2010 Phone recognition with the mean-covariance restricted boltzmann machine *Proc. 23rd Int. Conf. on Neural Information Processing Systems - Volume 1 NIPS'10* (Curran Associates Inc.) pp 469–77
Dahl G E, Yu D, Deng L and Acero A 2012 Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition *IEEE Trans. Audio Speech Language Process.* **20** 30–42
de Chazal P, O'Dwyer M and Reilly R 2004 Automatic classification of heartbeats using ECG morphology and heartbeat interval features *IEEE Trans. Biomed. Eng.* **51** 1196–206
Dózsa T, Böck C, Bognár G, Meier J and Kovács P 2021 Color classification of visually evoked potentials by means of Hermite functions *Proc. 55th Asilomar Conf. on Signals, Systems and Computers* pp 251–5
Dózsa T, Radó J, Volk J, Kisari A, Soumelidis A and Kovács P 2022 Road abnormality detection using piezoresistive force sensors and adaptive signal models *IEEE Trans. Instrum. Meas.* **71** 1–11
Gan Y, cheng Shi J, Ming He W and jia Sun F 2021 Parallel classification model of arrhythmia based on densenet-bilstm *Biocybern. Biomed. Eng.* **41** 1548–60
Goldberger A L, Amaral L A, Glass L, Hausdorff J M, Ivanov P C, Mark R G, Mietus J E, Moody G B, Peng C-K and Stanley H E 2000 Physiobank, physiotoolkit and physionet: components of a new research resource for complex physiologic signals *Circulation* **101** e215–20
Golub H G and Van Loan F C 2013 *Matrix Computations* 4th edn. (The Johns Hopkins University Press)
Gospodinov M, Gospodinova E and Georgieva-Tsaneva G 2019 Mathematical methods of ECG data analysis *Healthcare Data Analytics and Management* (Elsevier) pp 177–209
Hanin B 2018 Which neural net architectures give rise to exploding and vanishing gradients? *NIPS'18* (Curran Associates Inc.) pp 580–9
Harada S, Hayashi H and Uchida S 2019 Biosignal generation and latent variable analysis with recurrent generative adversarial networks *IEEE Access* **7** 144292–302
Hassler A P, Menasalvas E, José G-G F, Rodríguez-Manas L and Holzinger A 2019 Importance of medical data preprocessing in predictive modeling and risk factor discovery for the frailty syndrome *BMC Med. Inf. Decis. Making* **19** 1–17
Ho T K 1995 Random decision forests *Proc. 3rd International Conference on Document Analysis and Recognition* vol 1 (IEEE) pp 278–82
Hubel D H and Wiesel T N 1959 Receptive fields of single neurones in the cat's striate cortex *J. Physiol.* **148** 574–91
Jakovac A 2021 Time series analysis with dynamic law exploration (arXiv:2104.10970)
Jakovac A, Berenyi D and Posfay P 2020 Understanding understanding: a renormalization group inspired model of (artificial) intelligence (arXiv:2010.13482)
Jakovác A, Kurbucz M T and Pósfay P 2022 Reconstruction of observed mechanical motions with artificial intelligence tools *New J. Phys.* **24** 073021
Jangra M, Dhull S K, Singh K K, Singh A and Cheng X 2023 O-WCNN: an optimized integration of spatial and spectral feature map for arrhythmia classification *Complex Intell. Syst.* **9** 2685–98
Jekova I, Bortolan G and Christov I 2008 Assessment and comparison of different methods for heartbeat classification *Med. Eng. Phys.* **30** 248–57
Joshi S L, Vatti R A and Tornekar R V 2013 A survey on ECG signal denoising techniques *2013 Int. Conf. on Communication Systems and Network Technologies* pp 60–64
Jun T J, Park H J and Kim Y-H 2016 Premature ventricular contraction beat detection with deep neural networks *Proc. 15th IEEE Int. Conf. on Machine Learning and Applications* pp 859–64
Kemker R, McClure M, Abitino A, Hayes T and Kanan C 2018 Measuring catastrophic forgetting in neural networks *Proc. AAAI Conference on Artificial Intelligence* vol 32 (https://doi.org/10.1609/aaai.v32i1.11651)
Keogh E 1997 Fast similarity search in the presence of longitudinal scaling in time series databases *Proc. 9th IEEE Int. Conf. on Tools With Artificial Intelligence* (IEEE) pp 578–84
Keogh E 2019 TwoLeadECG dataset *Time Series Class.* (available at: https://www.timeseriesclassification.com/description.php?Dataset=TwoLeadECG)
Kiranyaz S, Ince T and Gabbouj M 2016 Real-time patient-specific ECG classification by 1-D convolutional neural networks *IEEE Trans. Biomed. Eng.* **63** 664–75
Kiymaç E and Kaya Y 2023 A novel automated CNN arrhythmia classifier with memory-enhanced artificial hummingbird algorithm *Expert Syst. Appl.* **213** 119162
Kovács P, Bognár G, Huber C and Huemer M 2022 VPNet: variable projection networks *Int. J. Neural Syst.* **32** 2150054
Kovács P, Fridli S and Schipp F 2020 Generalized rational variable projection with application in ECG compression *IEEE Trans. Signal Process.* **68** 478–92
Kovács P and Samiee K 2022 Arrhythmia detection using spiking variable projection neural networks *Proc. 49th Computing in Cardiology Conf. (CinC)* pp 1–4
Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks *Advances in Neural Information Processing Systems* vol 25, ed F Pereira, C Burges, L Bottou and K Weinberger (Curran Associates, Inc.)
Kurbucz M T, Pósfay P and Jakovác A 2022 Facilitating time series classification by linear law-based feature space transformation *Sci. Rep.* **12** 18026
Kurbucz M T, Pósfay P and Jakovác A 2023 Predicting the price movement of cryptocurrencies using linear law-based transformation (arXiv:2305.04884)
Kurbucz M T, Pósfay P and Jakovác A 2024 LLT: an R package for linear law-based feature space transformation *SoftwareX* **25** 101623
Lecun Y, Bottou L, Bengio Y and Haffner P 1998 Gradient-based learning applied to document recognition *Proc. IEEE* **86** 2278–324
Letz T *et al* 2023 Automatic ECG-based detection of left ventricular hypertrophy and its predictive value in haemodialysis patients *Physiol. Meas.* **44** 075002
Li Y, Qian R and Li K 2022 Inter-patient arrhythmia classification with improved deep residual convolutional neural network *Comput. Methods Programs Biomed.* **214** 106582
Lippmann R 1987 An introduction to computing with neural nets *IEEE ASSP Mag.* **4** 4–22

Lipton Z C 2018 The mythos of model interpretability: in machine learning, the concept of interpretability is both important and slippery *Queue* **16** 31–57

Luz E J D S, Schwartz W R, Cámara-Chávez G and Menotti D 2016 ECG-based heartbeat classification for arrhythmia detection: a survey *Comput. Methods Programs Biomed.* **127** 144–64

Mar T, Zaunseder S, Martínez J P, Llamedo M and Poll R 2011 Optimization of ECG classification by means of feature selection *IEEE Trans. Biomed. Eng.* **58** 2168–77

McCulloch W S and Pitts W 1943 A logical calculus of the ideas immanent in nervous activity *Bull. Math. Biophys.* **5** 115–33

McInnes L, Healy J and Melville J 2018 UMAP: uniform manifold approximation and projection for dimension reduction (arXiv:1802.03426)

MIT-BIH 2005 MIT-BIH arrhythmia database (available at: www.physionet.org/physiobank/database/mitdb/)

Moody G and Mark R 2001 The impact of the MIT-BIH arrhythmia database *IEEE Eng. Med. Biol. Mag.* **20** 45–50

Packard N H, Crutchfield J P, Farmer J D and Shaw R S 1980 Geometry from a time series *Phys. Rev. Lett.* **45** 712–6

Qiao X, Lee K, Mokhtar S A, Ismail I, bin Md Pauzi A L, Zhang Q and Lim P Y 2023 Deep learning-based ECG arrhythmia classification: a systematic review *Appl. Sci.* **13** 1–25

Rafiei M H and Adeli H 2017 A new neural dynamic classification algorithm *IEEE Trans. Neural Netw. Learn. Syst.* **28** 3074–83

Rafiei M H, Gauthier L V, Adeli H and Takabi D 2022 Self-supervised learning for electroencephalography *IEEE Trans. Neural Netw. Learn. Syst.* **35** 1457–71

Schölkopf B, Smola A and Müller K-R 1998 Nonlinear component analysis as a kernel eigenvalue problem *Neural Comput.* **10** 1299–319

Serhani M A, El Kassabi T, Ismail H and Nujum Navaz A 2020 ECG monitoring systems: review, architecture, processes and key challenges *Sensors* **20** 1796

Simonyan K and Zisserman A 2015 Very deep convolutional networks for large-scale image recognition *3rd Int. Conf. on Learning Representations, ICLR 2015, (San Diego, CA, USA, 7 May–9 May, 2015), Conf. Track Proc.* ed Y Bengio and Y LeCun (https://doi.org/10.48550/arXiv.1409.1556)

Sundararajan M, Taly A and Yan Q 2017 Axiomatic attribution for deep networks *Proc. 34th Int. Conf. on Machine Learning* (*Proc. of Machine Learning Research*) vol 70, ed D Precup and Y W Teh (PMLR) pp 3319–28

Takens F 1981 Detecting strange attractors in turbulence *Dynamical Systems and Turbulence, Warwick 1980* ed D Rand and L-S Young (Springer) pp 366–81

van der Maaten L and Hinton G 2008 Visualizing data using T-SNE *J. Mach. Learn. Res.* **9** 2579–605 (available at: http://jmlr.org/papers/v9/vandermaaten08a.html)

Xu S S, Mak M-W and Cheung C-C 2019 Towards end-to-end ECG classification with raw signal extraction and deep neural networks *IEEE J. Biomed. Health Inf.* **23** 1574–84