

# Exploring Manifold Learning For Automated Electrocardiography Arrhythmia Analysis

Yan Yan, *Student Member, IEEE*, and Lei Wang,

**Abstract**—Arrhythmia analysis had been a classical research topic for decades. The various and changing morphologic features in the time series brought challenges for automated analysis. A geometrical descriptor for the time series is the mapping from the original input into some manifold. In this work, we investigate several manifold learning methods for electrocardiography embeddings, using the learned manifold features for arrhythmia analysis. The manifold methods have been validated on the MIT-BIH Arrhythmia Database. The intrinsic dimension analysis and the imbalance problem of electrocardiography dataset have been experimented and discussed. The proposed experiments shows that the multi-layer autoencoder generates a better representations than the other geometrical rule based manifold learning methods.

**Index Terms**—Manifold Learning, Arrhythmia, ECG Classification, Machine Learning

## I. INTRODUCTION

THE computer aided system for pathology detection and diagnosis are based on a feature space constructed from abundant clinical attributes. The noninvasive, inexpensive and well-established technology of electrocardiographic signal in mobile health or personal health has the greatest popularity in heart function analysis. Automated arrhythmia analysis provides indispensable assist in long-term clinical monitoring, and a large number of approaches have been proposed for the task, easing the diagnosis of arrhythmic changes as well as further inspection, e.g., heart rate variability or heart turbulence analysis. Information in electrocardiography signals related to the cardiac electrical activity is therefore represented by a large dimensional space from the data analysis angle. Plenty of parameters were extracted from the high dimensional data as the indicators for diagnosis. The large dimensional space hinders a proper interpretation of the embedded symbolic physiology in the feature space [1]. However, the increasingly developing of the artificial intelligence domain and machine learning methods provides powerful tools to deal with the large dimensional electrocardiography data.

Lots of machine learning algorithms had been proposed for the classification and detection of electrocardiography signals. The electrocardiography classification or detection task had been divided into two parts: the feature extraction process and classifier. Simple classifier such as linear discriminants [2] and kNN [3], more complex classifiers like neural networks [4],

Y. Yan is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, China, (e-mail: yan.yan@siat.ac.cn).

L. Wang is with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences.

Manuscript received ; revised.

[5], [6], [7], fuzzy inference engines [7], [8], hidden Markov model [9], [10], independent component analysis [11] and support vector machine [3], [12], [13] were also adopted by lots of researchers.

Beyond the classifier, the performance of a recognition system highly depends on the determination of extracted electrocardiography features. Time domain features, frequency domain features, and statistical measures features for six fundamental waves (PQRSTU) had been used in feature extraction process [14]. Time domain features like morphological features include shapes, amplitudes, and durations were adapted primarily in [15], [16], [17], frequency domain features like wavelet transformation were widely used [18], [19] stationary features like higher-order statistics also had been developed. Principal component analysis [20] and Hermite functions [21] have been used in electrocardiography classification and related analysis technologies as well. Almost every single published paper proposes a new set of features to be used, or a new combination of the existing ones [22].

The results from these algorithms or models were not amenable to expert labelling, as well as for the identification of complex relationships between subjects and clinical conditions [23]. But for the ambulatory electrocardiography clinical application, as well as the normal application in daily healthcare monitoring for cardiac function or early warning of heart disease, an automated algorithm or model would have significant meaning. The application of artificial intelligence methods has become an important trend in electrocardiography for the recognition and classification of different arrhythmia types [23]. The data explosion puts forward the new request to the method of data processing and information mining.

Over the past decades computational techniques proliferated in the pattern recognition field, simultaneously the applications in electrocardiography recognition, detection and classification for relevant trends, patterns, and outliers. Most of the literatures in the electrocardiography classification task were focused on the supervised learning methods, as in unsupervised learning methods were infrequently used, which needs a lot of effort in labelling data. The MIT-BIH database [24] was the most widely used data in the classification and detection algorithm developments, while mass unlabelled electrocardiography data had been ignored due to the supervise learning approaches essential. Unsupervised learning methods become crucial in mining or analysing unlabelled data, as the unlabelled electrocardiography data accumulated. Unsupervised learning-based approaches and the application to electrocardiogram classification in literatures mainly include clustering-based techniques [21], [25], [26],

self-adaptive neural network-based methods [27], [28] and some hybrid unsupervised learning systems [29].

Manifold learning is a kind of method attempting to uncover the manifold structure in a data set, it can also be considered as a kind of nonlinear dimensional reduction technique. It has been widely used in multivariate data classification, data analysis, and data visualization. Manifold learning directly map the input data space into some feature space to find a better descriptor to analysis the real world problems.

Manifold learning directly map the input data space into some feature space to find a better descriptor to analysis the real world problems. Manifold learning directly map the i

## II. METHODOLOGY

The arrhythmia analysis problem can be generally considered as one time series based classification problem from the viewpoint of data analysis. Consider a set of  $n$  electrocardiography based time series data set  $\mathbf{X} = \{\mathbf{x}_i\} \subset \mathbb{R}^d$ , is given as the samples. Heartbeats in  $\mathbf{X}$  include both labelled and unlabelled samples, the task is to estimate the labels of unlabelled data with some methods based on the labelled and relative methods. Here we adopt several manifold learning based techniques to embed the high dimensional input data into some lower dimension subspace. With the techniques some data properties were preserved in the embedded subspace (feature space). Then we train a classifier in the feature space using the labelled data, and use the classifier to classify the unlabelled data. From which we can get the arrhythmia information. Here we introduce several manifold learning techniques, the detailed theory and explanation can be accessed from the related literatures.

### A. PCA

Principal Component Analysis (PCA) is a linear dimensionality reduction method, which try to find a linear subspace of lower dimension keeping the largest variance compare to the original feature space. PCA is by far the most popular unsupervised linear technique. The most important task in PCA method is to find the principal components of a data set, which can be implemented by finding a linear basis with reduced dimensionality for the input data sets, with the amount of variance in the data is maximal.

Mathematically, consider data set  $\{\mathbf{x}_i\} \subset \mathbb{R}^d$ , the embedded subspace with PCA can be denote as  $\{\mathbf{y}_i\} \subset \mathbb{R}^k$  ( $d$  and  $k$  are the dimension numbers). Then the problem of finding the “best”  $k$  principal components can be transfer to finding  $k$ -dimensional subspaces that minimize the orthogonal distances. Solve

$$\operatorname{argmin}_{ij} \sum_{ij} (d_{ij}^2 - \|\mathbf{y}_i - \mathbf{y}_j\|^2) \quad (1)$$

where  $P_s(\cdot)$  is the projection onto subspace  $\mathcal{S}$ . Solution of this problem is to find a linear mapping  $\mathbf{M}$  from the original space to subspace  $\mathcal{S}$ , which maximizes the cost function  $\text{trace}(\mathbf{M}^T \text{cov}(\mathbf{X}) \mathbf{M})$ . Then the data low-dimensional data features  $\mathbf{y}_i$  of data points  $\mathbf{x}_i$  in original space can be computed by mapping  $\mathbf{Y} = \mathbf{X} \mathbf{M}$ . A detailed theory analysis and tutorial can be found in [30] and [31] respectively.

### B. Kernel PCA

The linear mapping could not be an accurate description of data in the nonlinear case, which happens in real world applications like the arrhythmia analysis. In these cases, PCA will produce a large error measure. The geometrically nonlinear surface of data motivated different kinds of modeling approaches include kernel PCA. Kernel PCA is a reformulation of linear PCA in a high-dimensional space constructed using kernel functions. Kernel PCA computes the principal eigenvectors of the kernel matrix rather than the covariance matrix in the origin PCA method. Apparently, constructing the kernel space transfer the linear based PCA into a nonlinear mapping. The idea behind kernel PCA is to project the data into a new, higher-dimensional feature space.

Mathematically, let  $n$  data points (here are the segmented heartbeat sample)  $\mathbf{x}_i \in \mathbb{R}^d$  be given, suppose  $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^D$ , where  $D > d$ . Assume that the mapping in the feature vectors have zero mean which is  $\frac{1}{n} \sum_{i=1}^n \phi(\mathbf{x}_i) = 0$ . Use  $\Phi = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_n)]^T \in \mathbb{R}^{n \times D}$ , apply PCA to  $\Phi$ . Usually the  $\phi(\mathbf{x}_i)$  are unknown and it is not possible to work out the decomposition explicitly, then we define

$$\kappa(\mathbf{x}_i, \mathbf{x}_i) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_i) \quad (2)$$

and consider  $\Phi \Phi^T = \{\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_i)\}$ , so we have kernel matrix under the mapping of  $\phi(\cdot)$  is  $\mathbf{K} = \{\kappa(\mathbf{x}_i, \mathbf{x}_j)\}$ . The principal  $d$  eigenvectors of the centered kernel matrix can be computed, then the covariance matrix in feature space is constructed by  $\kappa$ . A similar optimization problem can be summarized like in Equation 1 while the projection involves a kernel transformation. Details of kernel PCA theory and application tutorial can be found in [32], [33], [34]. Kernel PCA is a kernel-based method, the mapping performed greatly relies on the choice of kernel function  $\kappa$ . The linear mapping PCA could be equal to a Kernel PCA when a linear kernel is chosen. Typical kernel functions include Gaussian kernel, polynomial kernel etc.

### C. Maximum Variance Unfolding (Semidefinite Embedding)

Since in the kernel PCA based method, the choice of kernel function is quite arbitrary. Sometime a poor kernel function could not lead to a good manifold embedding. Maximum Variance Unfolding (MVU) is a technique that attempts to solve such problem by learning from data so that the kernel matrix can be obtained, which was formerly known as Semidefinite Embedding [35].

The notion of local isometry was proposed in MVU. In mathematics, an isometry of a manifold is any (smooth) mapping of that manifold into itself, or into another manifold that preserves the notion of distance between points. Given  $n$  input points  $\mathbf{x}_i \in \mathbb{R}^d$  and a prescription for identifying neighborhood relations, find some mapped output  $\mathbf{y}_i \in \mathbb{R}^k$  such that both the inputs and outputs are both locally isometric (or approximation locally isometric). MVU starts with the construction of graph  $\mathcal{G}$  illustrates the neighborhood relations.  $\mathbf{x}_i$  is connected to its  $k$  nearest neighbors, MVU tries to maximize the sum of the squared Euclidean distances between data points, with the constraint that the distances inside  $\mathcal{G}$

are preserved. Mathematically, let  $\mathbf{y}_i$  denote the mapped representation of  $\mathbf{x}_i$ , and define a kernel matrix  $\mathbf{K}$  as the outer product of data presentations  $\mathbf{Y}$ . After reformulating the problem turns to:

$$\begin{aligned} \operatorname{argmax} & \sum_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \\ \text{s.t. } & \|\mathbf{y}_i - \mathbf{y}_j\|^2 = \|\mathbf{x}_i - \mathbf{x}_j\|^2 \quad \text{for } \forall(i, j) \in \mathcal{G} \end{aligned} \quad (3)$$

After we solve the semidefinite programming problem (SDP), the low-dimensional data representation  $\mathbf{Y}$  is obtained by performing an eigenvector decomposition of  $\mathbf{K}$ .

#### D. Isomap

PCA finds a low-dimensional representation of the data points that best preserves the variance as which measured in the high-dimensional input space. Later the classical multidimensional scaling (MDS) method was proposed which finds an embedding preserving the inter-point distances[36]. The MDS method equivalent to PCA when those distances are Euclidean. While in many case, the high-dimensional data lies on or near a curved manifold, PCA and MDS may lead a mistake in the some datasets contain essential nonlinear structures that are invisible to them, like in some face recognition dataset. Isomap builds on classical MDS but seeks to preserve the intrinsic geometry of the data, as captured in the geodesic manifold distances between all pairs of data points [37].

Mathematically, the geodesic distance between the data points  $\{\mathbf{x}_i\} \subset \mathbb{R}^d$  are computed so as to construct a neighborhood graph  $\mathcal{G}$ , where every data point  $\{\mathbf{x}_i\}$  is connected with its  $k$  nearest neighbors in the dataset. The shortest path between two points in the graph forms an estimate of geodesic distance, which can be computed using Dijkstra's shortest-path algorithm, therefore we can get a pairwise geodesic distance matrix  $\mathcal{D}$ . The low-dimensional representation can be achieved by MDS on  $\mathcal{D}$ .

#### E. Local Linear Embedding

Local Linear Embedding is a technique that is similar to Isomap and MVU, all try to construct a graph representation of the data points. Compare to Isomap, LLE only attempts to preserve local properties of data [38], which are constructed by writing the high-dimensional data points as a linear combination of their nearest neighbors. In the low-dimensional manifold, LLE attempts to retain the reconstruction weights in the linear combinations as good as possible.

LLE describes the local properties of the manifold around a data point  $\mathbf{x}_i$  using a linear combination of its  $k$  nearest neighbors with related reconstruction weights  $\mathbf{w}_i$ . There is an assumption that the manifold is locally linear, which means that  $\mathbf{w}_i$  of datapoints  $\mathbf{x}_i$  are invariant to translation, rotation, and rescaling. Then the reconstruction weights  $\mathbf{w}_i$  can also reconstruct datapoint  $\mathbf{y}_i$  from its neighbors in the low-dimensional data representation. Then finding the data representation  $\mathbf{Y}$  can be consider as an optimization problem

$$\begin{aligned} \operatorname{argmin} & \sum_i \|\mathbf{y}_i - \sum_{j=1}^k w_{ij} \mathbf{y}_{i,j}\|^2 \\ \text{s.t. } & \|\mathbf{y}^{(k)}\| = 1 \quad \text{for } \forall k \end{aligned} \quad (4)$$

where  $\mathbf{y}^{(k)}$  represents the  $k$ -th column of the solution matrix  $\mathbf{Y}$ .

#### F. Laplacian Eigenmaps

Similar to LLE, the Laplacian Eigenmaps (LE) methods find low-dimensional data representations by preserving manifold's local properties based on pairwise distances of near neighbors. LE compute distances between data points with its neighbors using weights based on the rule: the first nearest neighbor contributes more to the cost function than the distance between the datapoint and its second nearest neighbor. The minimization of the cost function in LE is defined as an eigen-problem in spectral graph theory.

Consider a neighborbood graph  $\mathcal{G}$  in which each data sample  $\mathbf{x}_i$  is connected to its  $k$  nearest neighbors. The weights of the edges are computed using Gaussian kernel function:

$$w_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}} \quad (5)$$

where  $\sigma$  indicates the Gaussian variance. Then we can get a sparse adjacency matrix  $\mathbf{W}$ , and the optimization problem:

$$\operatorname{argmin} \sum_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 w_{ij} \quad (6)$$

Large weights  $w_{ij}$  correspond to small distances of  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , then the difference between their manifold representation  $\mathbf{y}_i$  and  $\mathbf{y}_j$  highly contributes to the cost function. So the nearby points in the original space (high-dimensional) are put as close as possible in the low-dimensional manifold, which forms a better representation for input datasets [39].

#### G. Autoencoder

While the above methods are based on some geometrical explantations, Auto Encoders are a family of nonlinear dimensional reduction methods. Autoencoders are neural networks had been used and explained in the term deep learning where had been used to predict the input by let the output is trained to be as similar as possible as the input. The structure of autoencoders are illustrated in Figure II-G, on the end of autoencoder there were less hidden nodes than input nodes by encoding as much information as it can be for the hidden neural nodes. Superficially high-dimensional and complex phenomena can be mapped into some lower dimensional manifold, performs good representation of inputs.

Suppose the input  $\mathcal{X} = \{\hat{\mathbf{x}}_i\}$  and the learned autoencoder based manifold  $\mathcal{Y} = \{\mathbf{y}_i\}$ . The autoencoder structure is constructed with an  $a$ -layer neural network in which the output  $h_{W,b}(\mathbf{x})$  equals to the input  $\mathbf{x} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \dots, \hat{\mathbf{x}}_n)^T$ . Let an autoencoder parameters are  $\mathbf{W}$  and  $\mathbf{b}$ , and  $f, g$  were encoder and decoder respectively. The output of autoencoder is:

$$h_{W,b}(\hat{\mathbf{x}}) = g(f(\hat{\mathbf{x}})) \approx \hat{\mathbf{x}} \quad (7)$$

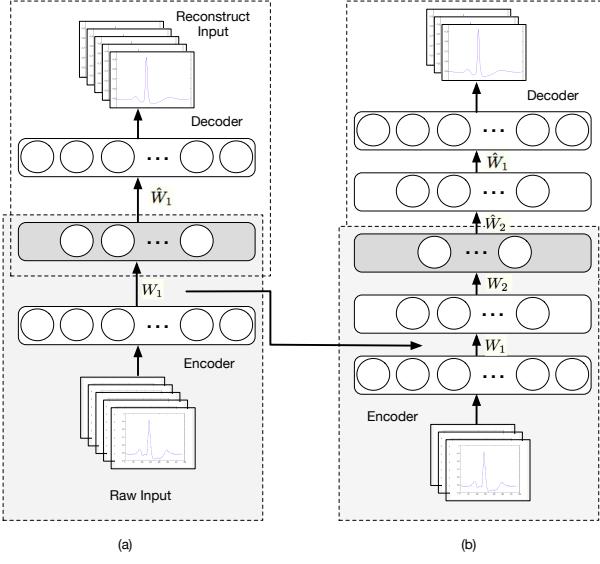


Fig. 1. A Multi-layer Autoencoder Reconstruction Structure

Then the training of the autoencoder can be consider as a problem of:

$$\operatorname{argmin}_{\mathbf{W}, \mathbf{b}} \frac{1}{2} \|h_{\mathbf{W}, \mathbf{b}}(\hat{\mathbf{x}}) - \hat{\mathbf{x}}\|^2 \quad (8)$$

This multi-layer neural network structure can be trained using the backpropagation method in a unsupervised way. In deep learning literatures a layer-wised method had been widely used as [40] illustrated.

Once the architecture parameters of  $\mathbf{W}, \mathbf{b}$  are learned, one can use the last layer of encoder as the output. From the view point of manifold learning, the autoencoder mapped the original input into a manifold feature space. One can use the manifold feature for further classification problem.

#### H. Arrhythmia Classification with Softmax Classifier

After the manifold embedding procedure for the input  $\mathcal{X} = \{\mathbf{x}_i\}$ , we get the data features  $\mathcal{Y} = \{\mathbf{y}_i\}$  on manifold, with a cascade classifier the arrhythmia classification could be accomplished.

The softmax function is a generalization of the logistic function which had been widely used as a multi-class classifier or the last layer of neural networks [41]. In probability theory, the outputs can represent categorical distribution over  $k$  possible outcomes. We use softmax regression as the classifier using the manifold representations as the training samples for the classifier, and then deal the test dataset using the same embedding methods. The whole task can be illustrated as two phases, the learning phase and the predict(classifying) phase. In the learning phase, first the ECG samples are embedded into some manifold to get correspond features with dimensional reduction methods as we illustrated in Section II.A to Section II.G. After the embedding, the achieved features are used for training a softmax based classifier in a supervised learning

method. In the prediction phase use the same strategy to get embedded features, and the class label for arrhythmia analysis can be achieved by the trained softmax classifier.

Mathematically, let  $\{c_i\}$  illustrated the class label,  $\mathcal{X} = \{\mathbf{x}_i\}$  as input data, and  $\mathcal{Y} = \{\mathbf{y}_i\}$  the manifold feature. Let  $1\{\cdot\}$ be the indicator function, so that  $1\{\text{true statement}\} = 1$ , and  $1\{\text{false statement}\} = 0$ . Then the cost function for the softmax regression classifier between the ground truth label  $c^{(i)}$  and the estimated label  $g(\mathbf{y}^{(i)}, \theta)$  is computed by loss function L:

$$J(\theta) = \frac{1}{n} \sum_{i=1}^n L(c^{(i)}, g(\mathbf{y}^{(i)}, \theta)) + \lambda r(\theta) \quad (9)$$

where the loss function of  $\mathbf{y}^{(i)}$  is defined as:

$$L(c^{(i)}, g(\mathbf{y}^{(i)}, \theta)) = - \sum_{k=1}^K \mathbf{1}\{c^{(i)} = k\} \log \frac{e^{\theta_k^T \mathbf{y}^{(i)}}}{\sum_{c=1}^K e^{\theta_c^T \mathbf{y}^{(i)}}} \quad (10)$$

while the regularization term we use is

$$r(\theta) = \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^d \theta_{kj}^2 \quad (11)$$

It is well-known that softmax regression is a convex model which can find a global optimal solution [42], then the training for the classifier is:

$$\theta = \operatorname{argmin} J(\theta) \quad (12)$$

The class label output of the softmax classifier can be achieved via:

$$c_i = \operatorname{argmax} g(\mathbf{y}^{(i)}, \theta) \quad (13)$$

## III. EXPERIMENTS AND RESULTS

We include the above mentioned methods for the ECG arrhythmia classification task for a comparative study for an illustration of manifold learning in time series based biomedical signal analysis. In this section we first make a short description about the task and data types since this problem had been quite well discussed in related literatures, after which the experiment flows are illustrated.

#### A. ECG Arrhythmia & Heartbeat Classification

After the segmentation for the ECG records, we got plenty of ECG waveform samples with variety categories. Since different physiological disorder may be reflected on the different type of abnormal heartbeat rhythms. For the task of classification, it is quite important to determine the classes would be used. In the early literatures, there were no unified class labels for an ECG classification problem. The MIT-BIH Arrhythmia Database was the first available set of standard test material for evaluation of arrhythmia detectors; it played an important role in stimulating manufacturers of arrhythmia analyzers to compete by objectively measurable performance. The annotations in the open database for the ECG categories adopted the ANSI/AAMI EC57: 1998/(R)2008 standard AAMI (2008), which recommended to group the heartbeats into five classes: on-ectopic beats (N); supraventricular ectopic

beat (S); ventricular ectopic beat (V); fusion of a V and a N (F); unknown beat type (Q). These classes or labels have been widely used in the ECG classification tasks. The normal beat, supraventricular ectopic beat and the ventricular ectopic beat categories were used much more frequently while the unknown beat type were abandoned because of its clinical valueless.

The most popular dataset used in arrhythmia analysis task is the MIT-BIH Arrhythmia Database [23] which contains 48 half-hour recordings each containing two 30-min ECG lead signals (lead I and lead II), sampled at 340Hz. However, only the lead I was used in this study. In agreement with the AAMI recommended practice, the four recordings with paced beats were removed from the analysis.

Besides, a unlabelled data set of ambulatory electrocardiography were used in this study, which includes recordings of 100 subjects with partial arrhythmia ECG samples. This dataset would be used in the autoencoder pre-training, which has been quite popular in the deep learning literatures. Even though we explore autoencoder method from a manifold angle, a pre-training procedure is powerful strategy in extracting manifold representations for arrhythmia analysis.

### B. Data Preprocessing

In the preprocessing stage, filtering algorithms were adapted to remove the artifact signals from the ECG signal. The signals include baseline wander, power line interference, and high-frequency noise. The segmentation and R wave detection algorithms had been explored in [43] and segmentation program of Laguna [44] was adapted, which also had been validated by other related work [2]. After segmentation, the data samples become 340-dimensional time series.

### C. Intrinsic Dimension Analysis

The dimension of the embedding is a key parameter for manifold feature achieving. If the dimension is too small, important data features are "collapsed", meanwhile if the dimension is too large, the projection into a manifold is too large, leading to a noisy or unstable data feature. When performing the manifold learning methods, the learning parameter settings could be quite arbitrary sometimes. The intrinsic dimension analysis could help for the parameter determination. Especially in the neural network based autoencoder, there is an insufficiency of method for determining the dimension for output layer. Intrinsic dimension analysis provide a reasonable method for parameter determination.

The widely adopted method of intrinsic dimension analysis include correlation dimension estimation, nearest neighbor dimension estimation[45], geodesic minimum spanning tree[46], packing numbers [47], maximum likelihood estimation method [48], etc. Here we use the the maximum likelihood estimation method for intrinsic dimension analysis. The dimension from the input dimension turns to 14 with the maximum likelihood estimation based intrinsic dimension analysis.

### D. Sample Imbalance Treatment

For the arrhythmia analysis, a typical problem from the viewpoint of data analysis is the unbalancedness of data. It is

due to the characters from the cardiac physiological activity. Even for some severe heart disease, the ECG achieved would include much more normal beats than the arrhythmia beats after segmentation. The skewed distribution makes manifold learning algorithms less effective, especially in the prediction of minority class samples. Usually, a confusion matrix had been build for the performance assess to prove the validity of proposed methods. Here we arbitrarily delete part of the data sample to relieve the imbalance problem, and use the confusion matrix to evaluate the classification performance. The class distributions are illustrated in Table ??, a more complicate and detailed analysis of impact for imbalance impact can be found in [49] and more about two-class situation in [50].

TABLE I  
COMPARISONS OF IMBALANCED AND BALANCED OF DS-I  
FROM MITBHI-AR

Classes	N	S	V	F	Q
Imbalanced	90431	2774	7698	802	8023
Ratio	82.4%	2.5%	7%	0.7%	7.3%
Balanced	18086	2774	7698	802	8023
Ratio	48.38%	7.5%	20%	2.2%	21.5%

### E. Manifold Learning and Classification

The first step for the arrhythmia classification problem is mapping the data into some manifold to gain related data embedding. It can be considered as some feature extraction, or representation learning in the machine learning literatures. The difference was that these manifold based methods are inspired from some geometrical concepts. For the mentioned methods, the PCA, kernel based PCA, MVU, Isomap, LLE, Laplacian Eigenmaps only involve the MIT-BIH-AR data samples for geometrical transformation, in the autoencoder based manifold learning, we use some unlabelled data to train a multilayer neural network in a stacked manner.

As Algorithm 1 illustrated, let  $\mathcal{M}$  illustrated some mapping method with rule  $\mathcal{R}$ , and the related softmax classifier as  $\mathcal{S}$ , consider the balanced dataset as DS1 (from the MIT-BHI AR database).

---

#### Algorithm 1 Manifold Learning With Rule $\mathcal{R}$

**Input:** Data Set I (DS-I) Samples:  $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$ ;

**Output:** Sample Classes:  $\mathcal{C} = \{c_1, c_2, \dots, c_3\}$

$TrainSet, TestSet \leftarrow \mathcal{X}$

$\mathcal{M} \leftarrow$  Process  $TrainSet$  with criteria  $\mathcal{R}$

$\mathcal{Y} \leftarrow$  Map  $TrainSet$  with  $\mathcal{M}$

$\mathcal{S} \leftarrow$  Train the Softmax classifier with  $\mathcal{Y}$

$\mathcal{C} \leftarrow \mathcal{S}(TestSet)$

---

As Algorithm 2 illustrated, consider the multilayer autoencoder method, the random initialization of the neural network parameter might fail with the limited data samples in DS-I. Then we first use the unlabelled data set DS-II with the same preprocessing method to pre-train the neural network, it had been widely used in deep learning method [51], [52]. After the pre-training state, we use samples to do further training

for the representation, and the cascade classification. A 5-fold cross validation had been applied for the whole training and predict, refer [53] for detail.

#### Algorithm 2 Manifold Learning With Autoencoder

**Input:** Data Set I (DS-I) Samples:  $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ ;  
Data Set II (DS-II) Samples:  $\hat{\mathcal{X}} = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \dots, \hat{\mathbf{x}}_n\}$

**Output:** Sample Classes:  $\mathcal{C} = \{c_1, c_2, \dots, c_n\}$

$$\theta \leftarrow \text{self-reconstruction of } \hat{\mathcal{X}}$$

$$TrainSet, TestSet \leftarrow \mathcal{X}$$

$$\mathcal{Y} \leftarrow \text{Map } TrainSet \text{ with encoder parameters } \theta$$

$$S \leftarrow \text{Train the Softmax classifier with } \mathcal{Y}$$

$$C \leftarrow S, \theta, TestSet$$

#### F. Performance Analysis

Correctly detected episodes are termed true positive (TP) episodes, while the missed as false negatives (FNs). When a negative episode in some class is predicted as true is called true negative, and a false positive (FP) if it is considered as positive. And the sensitivity(Se), specificity(Sp), Precision, and F score are defined as following with respectively:

$$\begin{aligned} Se &= \frac{TP}{TP + FN} & Sp &= \frac{TN}{TN + FP} \\ Precision &= \frac{TP}{TP + FP} & F1 - Score &= \frac{2TP}{2TP + FP + FN} \end{aligned} \quad (14)$$

Refer [22] for an further knowledge for parameters for algorithm evaluation in ECG data analysis.

## IV. RESULTS AND DISCUSSION

Using the projectors from Section II, we use the time series based manifold feature to perform classification task. Since the working

### A. Experimental Results

Lore ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

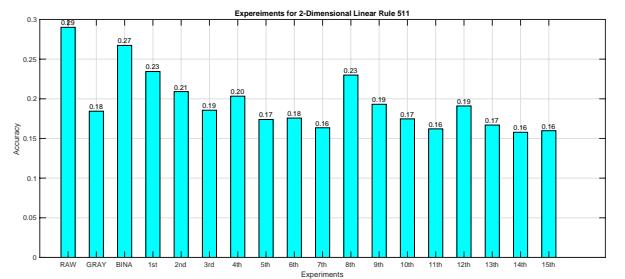
Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae

TABLE II  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label	Algorithm label					T
	N	S	V	F	Q	
N	41,778	38	48	17	5	41,886
S	93	460	3	1	2	559
V	52	1	4,067	11	6	4,137
F	15	0	13	214	2	244
Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

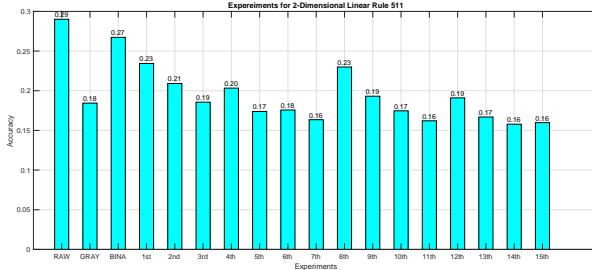


where an .eps filename suffix will be assumed under latex, and a .pdf suffix will be assumed for pdflatex; or what has been declared via

Fig. 2. Simulation results for the network.

Lore ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Lore ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at,



where an .eps filename suffix will be assumed under latex, and a .pdf suffix will be assumed for pdflatex; or what has been declared via

Fig. 3. Simulation results for the network.

mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

TABLE III  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label		Algorithm label					T
		N	S	V	F	Q	
Reference	N	41,778	38	48	17	5	41,886
label	S	93	460	3	1	2	559
	V	52	1	4,067	11	6	4,137
	F	15	0	13	214	2	244
	Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

TABLE IV  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label		Algorithm label					T
		N	S	V	F	Q	
Reference	N	41,778	38	48	17	5	41,886
label	S	93	460	3	1	2	559
	V	52	1	4,067	11	6	4,137
	F	15	0	13	214	2	244
	Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

TABLE V  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label		Algorithm label					T
		N	S	V	F	Q	
Reference	N	41,778	38	48	17	5	41,886
label	S	93	460	3	1	2	559
	V	52	1	4,067	11	6	4,137
	F	15	0	13	214	2	244
	Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

TABLE VI  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label		Algorithm label					T
		N	S	V	F	Q	
Reference	N	41,778	38	48	17	5	41,886
label	S	93	460	3	1	2	559
	V	52	1	4,067	11	6	4,137
	F	15	0	13	214	2	244
	Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

### B. Comparisons

1) *Manifold Methods:* Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis

egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

2) *Comparison With Other Works:* Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

TABLE VII  
TEST RESULT FOR 4-HIDDEN-LAYER AUTOENCODER NETWORK

Reference label	Algorithm label					
	N	S	V	F	Q	T
N	41,778	38	48	17	5	41,886
S	93	460	3	1	2	559
V	52	1	4,067	11	6	4,137
F	15	0	13	214	2	244
Q	1	0	1	1	1	5

The test accuracy is about 99.34%.

TABLE VIII  
PERFORMANCE COMPARISON FOR MANIFOLD LEARNING METHODS

Items	N-Sp	S-Sp	V-Sp	F-Sp	Q-Sp	F-Score
PCA	90%)	20%	2233	122%	1222	1
KPCA	2	3	d	d	111	1
MVU	2	3	d	d	111	1
Isomopha	2	3	d	d	111	1
LLE	2	3	d	d	111	1
LE	2	3	d	d	111	1
SAE	2	3	d	d	111	1

## V. CONCLUSION AND FUTURE WORK

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis,

viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## REFERENCES

- [1] E. Delgado-Trejos, A. Perera-Lluna, M. Vallverdú-Ferrer, P. Caminal-Magrangs, and G. Castellanos-Domínguez, "Dimensionality reduction oriented toward the feature visualization for ischemia detection," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 4, pp. 590–598, 2009.
- [2] P. de Chazal, M. O'Dwyer, and R. Reilly, "Automatic classification of heartbeats using ecg morphology and heartbeat interval features," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.
- [3] F. Melgani and Y. Bazi, "Classification of electrocardiogram signals with support vector machines and particle swarm optimization," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 12, no. 5, pp. 667–677, Sept 2008.
- [4] W. Jiang and S. Kong, "Block-based neural networks for personalized ecg signal classification," *Neural Networks, IEEE Transactions on*, vol. 18, no. 6, pp. 1750–1761, 2007.
- [5] T. Olmez, "Classification of ecg waveforms by using rce neural network and genetic algorithms," *Electronics Letters*, vol. 33, no. 18, pp. 1561–1562, Aug 1997.
- [6] C.-W. Lin, Y.-T. Yang, J.-S. Wang, and Y.-C. Yang, "A wearable sensor module with a neural-network-based activity classification algorithm for daily energy expenditure estimation," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 16, no. 5, pp. 991–998, Sept 2012.
- [7] S. Osowski and T. H. Linh, "Ecg beat recognition using fuzzy hybrid neural network," *Biomedical Engineering, IEEE Transactions on*, vol. 48, no. 11, pp. 1265–1271, Nov 2001.
- [8] M. Kundu, M. Nasipuri, and D. Basu, "A knowledge-based approach to ecg interpretation using fuzzy logic," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 28, no. 2, pp. 237–243, Apr 1998.
- [9] R. Andreao, B. Dorizzi, and J. Boudy, "Ecg signal analysis through hidden markov models," *Biomedical Engineering, IEEE Transactions on*, vol. 53, no. 8, pp. 1541–1549, 2006.
- [10] D. Coast, R. Stern, G. Cano, and S. Briller, "An approach to cardiac arrhythmia analysis using hidden markov models," *Biomedical Engineering, IEEE Transactions on*, vol. 37, no. 9, pp. 826–836, 1990.
- [11] Y. Zhu, A. Shayan, W. Zhang, T. L. Chen, T.-P. Jung, J.-R. Duann, S. Makeig, and C.-K. Cheng, "Analyzing high-density ecg signals using ica," *Biomedical Engineering, IEEE Transactions on*, vol. 55, no. 11, pp. 2528–2537, Nov 2008.
- [12] A. Kampouraki, G. Manis, and C. Nikou, "Heartbeat time series classification with support vector machines," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 13, no. 4, pp. 512–518, 2009.
- [13] A. Khandoker, M. Palaniswami, and C. Karmakar, "Support vector machines for automated recognition of obstructive sleep apnea syndrome from ecg recordings," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 13, no. 1, pp. 37–48, Jan 2009.
- [14] C.-P. Shen, W.-C. Kao, Y.-Y. Yang, M.-C. Hsu, Y.-T. Wu, and F. Lai, "Detection of cardiac arrhythmia in electrocardiograms using adaptive feature extraction and modified support vector machines," *Expert Systems with Applications*, vol. 39, no. 9, pp. 7845 – 7852, 2012.

- [15] I. Jekova, G. Bortolan, and I. Christov, "Assessment and comparison of different methods for heartbeat classification," *Medical Engineering & Physics*, vol. 30, no. 2, pp. 248–257, 2008.
- [16] I. Christov, G. Gomez-Herrero, I. Krasteva, I. Jekova, A. Gotchev, and K. Egiazarian, "Comparative study of morphological and time frequency ecg descriptors for heartbeat classification," *Medical Engineering & Physics*, vol. 28, no. 9, pp. 876–887, 2006.
- [17] C. Ye, B. Kumar, and M. Coimbra, "Heartbeat classification using morphological and dynamic features of ecg signals," *Biomedical Engineering, IEEE Transactions on*, vol. 59, no. 10, pp. 2930–2941, Oct 2012.
- [18] O. T. Inan, L. Giovangrandi, and G. T. A. Kovacs, "Robust neural-network-based classification of premature ventricular contractions using wavelet transform and timing interval features," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 12, pp. 2507–2515, 2006.
- [19] S. Banerjee and M. Mitra, "Application of cross wavelet transform for ecg pattern analysis and classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 2, pp. 326–333, 2014.
- [20] T. Stamkopoulos, K. Diamantaras, N. Maglaveras, and M. Strintzis, "Ecg analysis using nonlinear pca neural networks for ischemia detection," *Signal Processing, IEEE Transactions on*, vol. 46, no. 11, pp. 3058–3067, Nov 1998.
- [21] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo, "Clustering ecg complexes using hermite functions and self-organizing maps," *Biomedical Engineering, IEEE Transactions on*, vol. 47, no. 7, pp. 838–848, Jul 2000.
- [22] T. Mar, S. Zaunseder, J. Martinez, M. Llamedo, and R. Poll, "Optimization of ecg classification by means of feature selection," *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 8, pp. 2168–2177, Aug 2011.
- [23] G. D. Clifford, F. Azuaje, and P. McSharry, *Advanced Methods And Tools for ECG Data Analysis*. Norwood, MA, USA: Artech House, Inc., 2006.
- [24] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000 (June 13).
- [25] H. Nishizawa, T. Obi, M. Yamaguchi, and N. Ohyama, "Hierarchical clustering method for extraction of knowledge from a large amount of data," *Optical Review*, vol. 6, no. 4, pp. 302–307, 1999.
- [26] C. Maier, H. Dickhaus, and J. Gittinger, "Unsupervised morphological classification of qrs complexes," in *Computers in Cardiology, 1999*, 1999, pp. 683–686.
- [27] S. Palreddy, W. J. Tompkins, and Y. H. Hu, "Customization of ecg beat classifiers developed using som and lvq," in *Engineering in Medicine and Biology Society, 1995., IEEE 17th Annual Conference*, vol. 1, Sep 1995, pp. 813–814 vol.1.
- [28] M. Risk, J. Sobh, and J. Saul, "Beat detection and classification of ecg using self organizing maps," in *Engineering in Medicine and Biology Society, 1997. Proceedings of the 19th Annual International Conference of the IEEE*, vol. 1, Oct 1997, pp. 89–91 vol.1.
- [29] P. Tadejko and W. Rakowski, "Hybrid wavelet-mathematical morphology feature extraction for heartbeat classification," in *EUROCON, 2007. The International Conference on Computer as a Tool*, Sept 2007, pp. 127–132.
- [30] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.
- [31] J. Shlens, "A tutorial on principal component analysis," *arXiv preprint arXiv:1404.1100*, 2014.
- [32] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*. Cambridge university press, 2004.
- [33] C. Chatfield, *Introduction to multivariate analysis*. Routledge, 2018.
- [34] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: a comparative," *J Mach Learn Res*, vol. 10, pp. 66–71, 2009.
- [35] K. Q. Weinberger and L. K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *International journal of computer vision*, vol. 70, no. 1, pp. 77–90, 2006.
- [36] J. B. Kruskal and M. Wish, *Multidimensional Scaling*. BOOK ON DEMAND POD, 1978.
- [37] J. B. Tenenbaum, S. V. De, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–23, 2000.
- [38] L. K. Saul and S. T. Roweis, "An introduction to locally linear embedding," *unpublished. Available at: http://www.cs.toronto.edu/~roweis/lle/publications.html*, 2000.
- [39] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [40] Y. Bengio, "Learning deep architectures for ai," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, Jan. 2009. [Online]. Available: <http://dx.doi.org/10.1561/2200000006>
- [41] M. B. Christopher, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.
- [42] Y. Ren, P. Zhao, Y. Sheng, D. Yao, and Z. Xu, "Robust softmax regression for multi-class classification with self-paced learning," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. AAAI Press, 2017, pp. 2641–2647.
- [43] V. Afonso, W. J. Tompkins, T. Nguyen, and S. Luo, "Ecg beat detection using filter banks," *Biomedical Engineering, IEEE Transactions on*, vol. 46, no. 2, pp. 192–202, Feb 1999.
- [44] L. Sörnmo and P. Laguna, "Electrocardiogram (ecg) signal processing," *Wiley Encyclopedia of Biomedical Engineering*, 2006.
- [45] J. A. Costa, A. Girotra, and A. Hero, "Estimating local intrinsic dimension with k-nearest neighbor graphs," in *Statistical Signal Processing, 2005 IEEE/SP 13th Workshop on*. IEEE, 2005, pp. 417–422.
- [46] J. A. Costa and A. O. Hero, "Geodesic entropic graphs for dimension and entropy estimation in manifold learning," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2210–2221, 2004.
- [47] B. Kégl, "Intrinsic dimension estimation using packing numbers," in *Advances in neural information processing systems*, 2003, pp. 697–704.
- [48] E. Levina and P. J. Bickel, "Maximum likelihood estimation of intrinsic dimension," in *Advances in neural information processing systems*, 2005, pp. 777–784.
- [49] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1119–1130, 2012.
- [50] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.
- [51] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.
- [52] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.
- [53] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.



**Yan Yan** received the B.Eng. degree and the MSc in Instrument Engineering at the Harbin Institute of Technology in 2010 and 2012 respectively. He worked as research assistance at the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences from 2012 to 2014. He is working on his PhD in Computer Science started from 2014. His research interests are digital signal processing, machine learning, and control system.



**Lei Wang** received the B.Eng. degree in information and control engineering, and the Ph.D. degree in biomedical engineering from Xian Jiaotong University, Xian, China, in 1995 and 2000, respectively. He was with the University of Glasgow, Glasgow, U.K., and Imperial College London, London, U.K., from 2000 to 2008. He is currently a full professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. He has published over 200 scientific papers, authored four book chapters, and holds 60 patents. His current research interests include body sensor network, digital signal processing, and biomedical engineering.