Chapter 1

Controlled Spatial Transformation With Deterministic Mapping Function

1.1 Introduction

In Chapter 2, we elaborated on the details of the patient-adaptive ECG classification framework. An important part of this methodology is to design a *personalized classifier* with a *deviation analysis module*. The performance of the *deviation analysis module* depends on the geometry of different classes of data samples in the feature space. In Chapter 3, we proposed a novel spatial transformation method based on MOPSO to achieve a desired symmetry in the transformed feature space to boost the performance of the deviation analysis module. This methodology enables us to further process the normal samples and identify fuzzy states between the normal and abnormal states. However, the proposed method uses an iterative optimization which may have high computational complexity. In this chapter, we introduce another deterministic spatial transformation method to model the fuzzy state of samples with a more tractable and analytical solution.

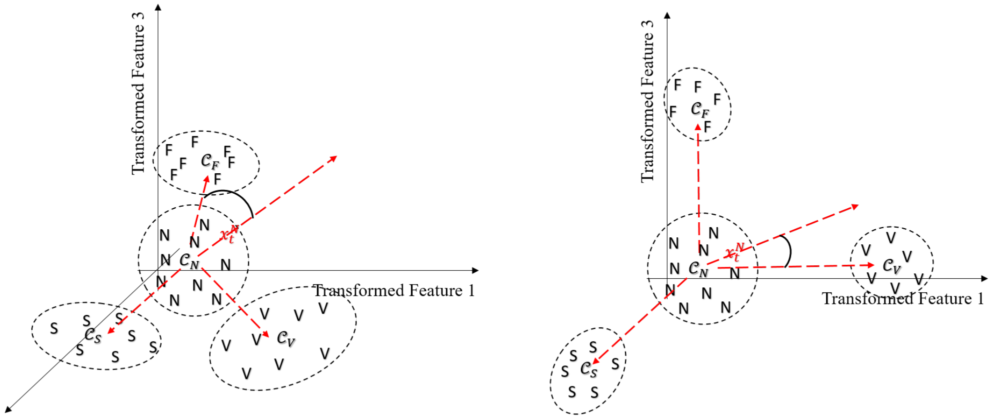

Figure 1.1: Left: illustration of the clustering topology in the original feature space; Right: illustration of the clustering topology in the feature space transformed with an optimized mapping function. Razi: it is not clear what you mean here by simple, is it unoptimized, why not just saying the original space before transmission !

While the aforementioned method in Chapter 3 yields the desired capacity of predicting upcoming abnormalities, the interpretation of the mechanisms of the system is not straightforward and easily tractable, thus it is not easily generalizable to a broader range of applications in the biomedical signal processing. The main objective of this chapter is to develop a deterministic transformation with a stronger prediction power based on a controlled spatial topology studied in Chapter 3. In this method, we further optimize the topology and spatial geometry of the clusters in the feature space by reducing the within-cluster variance after the spatial transformation. As shown in Fig. 1.1, through this improvement, the performance of the personalized classifier in terms of prediction accuracy can be further enhanced with the proposed method.

1.2 Hyper-Spherical Coordinates

In the previous chapter, the spatial transformation module is implemented with a polynomial kernel function and a heuristic optimization method. The system performance has been proven to be promising. In addition to the general drawbacks of heuristic method, it is not straightforward to select an appropriate base kernel as the core of spatial mapping function due to the high variety of kernel functions.

In order to address this issue, a novel deterministic spatial mapping function is proposed in this chapter based on *hyper-spherical* coordinates [REF: XXX (it is ok to repeat references, ans is ok to have web pages as REFs)]. Since hyper-spherical coordinates consist of angles and radii, these parameters in both original feature space and the desired target space are used to determine the mapping function.

The hyper-spherical coordinate system (n-dimensional spherical coordinate system) and its mapping to the *Cartesian* coordinate system are first introduced in [52]. If $\mathbf{x}$ is a sample vector in a n-dimensional feature space, with its Cartesian coordinates $(\xi_1,\ \xi_2\ ...\xi_n)$, then its corresponding hyper-spherical coordinates can be obtained through Eq.1.1, which is originally derived through its reverse mapping (Eq.1.2) using equation: $\sin(\arccos(x)) = \sqrt{1-x^2}$.

$$\begin{aligned}
r &= \sqrt{{\xi_n}^2 + {\xi_{n-1}}^2 + \cdots + {\xi_2}^2 + {\xi_1}^2} \\
\theta_1 &= \arccos \frac{\xi_1}{\sqrt{{\xi_n}^2 + {\xi_{n-1}}^2 + \cdots + {\xi_1}^2}} \\
\theta_2 &= \arccos \frac{\xi_2}{\sqrt{{\xi_n}^2 + {\xi_{n-1}}^2 + \cdots + {\xi_2}^2}} \\
&\vdots \\
\theta_{n-2} &= \arccos \frac{\xi_{n-2}}{\sqrt{{\xi_n}^2 + {\xi_{n-1}}^2 + {\xi_{n-2}}^2}} \\
\theta_{n-1} &= \begin{cases} \arccos \frac{\xi_{n-1}}{\sqrt{{\xi_n}^2+{\xi_{n-1}}^2}} & \xi_n \geq 0 \\ -\arccos \frac{\xi_{n-1}}{\sqrt{{\xi_n}^2+{\xi_{n-1}}^2}} & \xi_n < 0 \end{cases}
\end{aligned} \tag{1.1}$$

$$
\begin{aligned}
\xi_1 &= r\cos(\theta_1)\\
\xi_2 &= r\sin(\theta_1)\cos(\theta_2)\\
\xi_3 &= r\sin(\theta_1)\sin(\theta_2)\cos(\theta_3)\\
&\vdots\\
\xi_{n-1} &= r\sin(\theta_1)\cdots\sin(\theta_{n-2})\cos(\theta_{n-1})\\
\xi_n &= r\sin(\theta_1)\cdots\sin(\theta_{n-2})\sin(\theta_{n-1}).
\end{aligned}
\tag{1.2}
$$

where $0 \le \theta_j \le \pi,\ j = 1, \ldots, n-2;\ 0 \le \theta_{n-1} \le 2\pi;\ 0 \le r < \infty$

1.3 Orthogonalization

To simplify the algorithm, in this chapter, the topology of clusters in the feature space is approximated by their centroid locations, denoted by $\mathbf{c}_N^k, \mathbf{c}_{\mathcal{V}}, \mathbf{c}_{\mathcal{S}}, \mathbf{c}_{\mathcal{F}}$, respectively for the normal cluster and the three abnormality clusters of type V, S, and F. Furthermore, as we assume that the samples with fuzzy states deviate from the normal cluster to a specific abnormality cluster, the spatial topology stays unchanged if the normal centroid is simply translated to the origin of the Cartesian coordinate system. The clustering topology in the original feature space can be equivalently represented by the following matrix with three row vectors:

$$\mathcal{C} = \begin{bmatrix} \mathbf{c}_{\mathcal{V}} - \mathbf{c}_N^k \\ \mathbf{c}_{\mathcal{S}} - \mathbf{c}_N^k \\ \mathbf{c}_{\mathcal{F}} - \mathbf{c}_N^k \end{bmatrix} = \begin{bmatrix} \mathbf{v}_{\mathcal{VN}} \\ \mathbf{v}_{\mathcal{SN}} \\ \mathbf{v}_{\mathcal{FN}} \end{bmatrix} \tag{1.3}$$

RZ: I used lower case v for vectors, which is more consistent with other definitions. Capital letter are more commonly used for matrices.

As shown in Fig. 1.1, in order to improve the performance of the personalized classifier and to avoid ambiguity in quantifying the deviations to different abnormality classes, a topology with a maximum separation between the three vectors of $\mathcal{C}$ is preferred. To achieve a lower computational complexity in higher order dimensions, the algorithm aims at transforming vectors in $\mathcal{C}$ to orthogonal vectors using a deterministic function. This approach not only simplifies the transformation derivations, it also ensures a full symmetry among abnormality clusters. As such, the problem reduces to finding a transformation that maps a set of vectors to a set of orthogonal vectors in the same space. Therefore, we can use the popular method of *Gram-Schmidt* orthogonalization method RZ: introduced or explained, choose the right term in [53, 54]. Hence, in the first step of this stage, the three row vectors of $\mathcal{C}$ representing the centroids of the three abnormality clusters are fed to the orthogonalization process as follows:

$$\mathcal{C}^{\perp} = \text{Gram-Schmidt}(\mathcal{C}) = \begin{bmatrix} \mathbf{v}^{\perp}_{\mathcal{VN}} \\ \mathbf{v}^{\perp}_{\mathcal{SN}} \\ \mathbf{v}^{\perp}_{\mathcal{FN}}, \end{bmatrix} \tag{1.4}$$

where $\mathcal{C}^{\perp}$ is the matrix of orthogonalized vectors in the Cartesian coordinate system. The hyper-spherical coordinates $\mathcal{C}^{\perp}_{*}$ of these orthogonalized vectors are calculated subsequently using Eq.1.1. After this step, the orthogonalized vectors in the hyper-spherical coordinate system are obtained as formulated in Eq. 1.5. The j^{th} angular dimension denoted as $\theta_j^{\perp}$ and the radius is noted as $r^{\perp}$.

$$\mathcal{C}_*^{\perp} = \begin{bmatrix} r_{\mathcal{VN}}^{\perp} & \theta_{1_{\mathcal{VN}}}^{\perp} & \dots & \theta_{n-1_{\mathcal{VN}}}^{\perp} \\ r_{\mathcal{SN}}^{\perp} & \theta_{1_{\mathcal{SN}}}^{\perp} & \dots & \theta_{n-1_{\mathcal{SN}}}^{\perp} \\ r_{\mathcal{FN}}^{\perp} & \theta_{1_{\mathcal{FN}}}^{\perp} & \dots & \theta_{n-1_{\mathcal{FN}}}^{\perp} \end{bmatrix} \tag{1.5}$$

1.4 Spatial Mapping Function

After obtaining the original hyper-spherical coordinates of $[\mathbf{v}_{\mathcal{VN}},\ \mathbf{v}_{\mathcal{SN}},\ \mathbf{v}_{\mathcal{FN}}]^T$ and the orthogonalized hyper-spherical coordinates $[\mathbf{v}_{\mathcal{VN}}^{\perp},\ \mathbf{v}_{\mathcal{SN}}^{\perp},\ \mathbf{v}_{\mathcal{FN}}^{\perp}]^T$, the goal is to design a mapping function $\mathbf{F} : \mathbf{R}^n \rightarrow \mathbf{R}^n$ from the original coordinates to the orthogonal ones which exhibit the desired clustering topology, such that Equi. 1.6 holds.

In the Gram-Schmidt algorithm, the very first vector serves as a reference vector and remains unchanged in the orthogonalization process, namely $\mathbf{v}_{\mathcal{VN}} = \mathbf{v}_{\mathcal{VN}}^{\perp}$. Therefore the mapping function $\mathbf{F}$ shall satisfy the following equations:

$$\begin{aligned} \mathbf{F}(\mathbf{v}_{\mathcal{SN}} - \mathbf{v}_{\mathcal{VN}}) = \mathbf{v}_{\mathcal{SN}}^{\perp} - \mathbf{v}_{\mathcal{VN}}^{\perp} \quad &= \mathbf{v}_{\mathcal{SN}}^{\perp} - \mathbf{v}_{\mathcal{VN}} \\ \mathbf{F}(\mathbf{v}_{\mathcal{FN}} - \mathbf{v}_{\mathcal{VN}}) = \mathbf{v}_{\mathcal{FN}}^{\perp} - \mathbf{v}_{\mathcal{VN}}^{\perp} \quad &= \mathbf{v}_{\mathcal{FN}}^{\perp} - \mathbf{v}_{\mathcal{VN}} \end{aligned} \tag{1.6}$$

Furthermore, since the orthogonality of vectors is independent to their radii r, we only need to design $\mathbf{F}$ for the $n-1$ angular dimensions $(\theta_1, \dots, \theta_{n-1})$, and the coordinate r remains unchanged after the mapping. Consequently, the function $\mathbf{F}$ can be decomposed into $n-1$ functions: $f_i : \mathbf{R} \rightarrow \mathbf{R},\ i = 1 \ldots n-1$ with constraints in Eq.1.6 as well as two additional extreme points defining the range of input and output values. We use the notation $\mathbf{v}_{\mathcal{SN}} - \mathbf{v}_{\mathcal{VN}} = \Delta_{\mathcal{SV}}$ and denote the the i^{th} angular dimension of $\Delta_{\mathcal{SV}}$ as $\delta_{i_{\mathcal{SV}}}$. We use similar notations for the other dimensions, (e.g. $\mathbf{v}_{\mathcal{FN}} - \mathbf{v}_{\mathcal{VN}}$). Hence for each angular dimension i, f_i is determined by ($\delta_{i_{\mathcal{SV}}}$,$\delta^{\perp}_{i_{\mathcal{SV}}}$) and ($\delta_{i_{\mathcal{FV}}}$,$\delta^{\perp}_{i_{\mathcal{FV}}}$), along with the two *extreme boundary points* defining the domain and range of the functions.

In order to maintain the simplicity and the linearity of the mapping function, the functions f_i should be continuous and monotonic. For this purpose, the valid range of angular dimensions (after considering the periodicity of these functions) is used to determine the boundary constrains; therefore the problem is transformed into a curve fitting problem.

RZ: the text was not clear, it now reads much better. For example, if linking ($\delta_{i_{\mathcal{SV}}}$,$\delta^{\perp}_{i_{\mathcal{SV}}}$) and ($\delta_{i_{\mathcal{FV}}}$,$\delta^{\perp}_{i_{\mathcal{FV}}}$) results in a monotonically increasing function, noting the fact that the range of $n-2$ first angular dimensions is the interval $[0, \pi]$, then the two extreme boundary points would be $(0, 0)$ and (π, π), as depicted in Fig. 1.2. Conversely, for a monotonically increasing function, the extreme boundary points would be $(\pi, 0)$ and $(0i, \pi)$. These rules apply for the functions defined for the $n-2$ first angular dimensions. Similar rules apply to the last angular dimension with only one consideration that the period is 2π instead of π. Since ($\delta_{i_{\mathcal{SV}}}$,$\delta^{\perp}_{i_{\mathcal{SV}}}$) and ($\delta_{i_{\mathcal{FV}}}$,$\delta^{\perp}_{i_{\mathcal{FV}}}$) represent the orthogonalization process, they are decisive for this process. So we call these two points as well as the two extreme boundary points as *target points* of the angular dimensions in the following sections.

The simplest candidate function for f_i, which connects the two boundary points and the two target points in the 2-D plane would be a linear spline as shown in Fig.1.2.

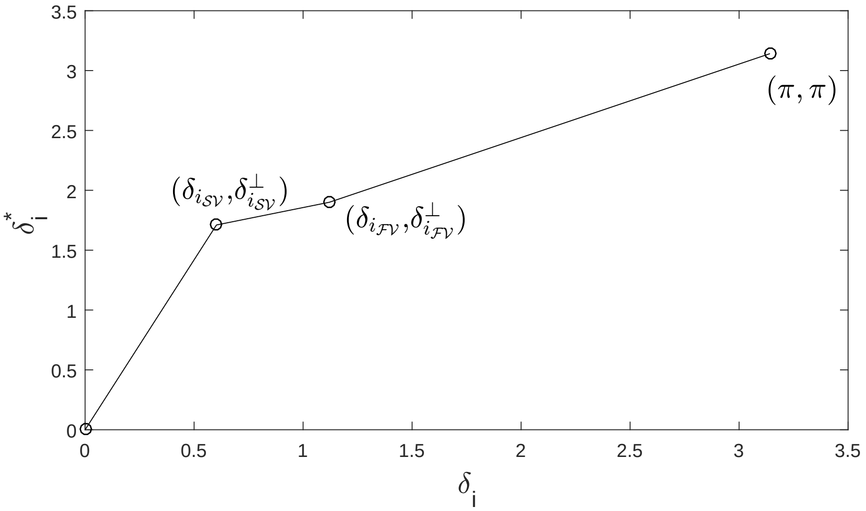

Figure 1.2: The simple mapping function for one angular dimension which maps the target points in the original space to the desired target points. Target points include the two extreme boundary points $(0,0)$, and $\pi,\pi)$ as well as (XXX,XXX) and (XXX,XXX) to yield the desired mapping. add (0,0) and the label target and extreme boundary points to the figure.

1.5 Optimized Mapping Function

The mapping function in Fig.1.2 exhibits the two desired properties of monotonicity and continuity for an ideal mapping function f_i. However, this simple peicewise linear mapping function suffers from some drawbacks. Firstlt, the function is not differentiable at the target points $(\delta_{i_{\mathcal{SV}}},\delta^{\perp}_{i_{\mathcal{SV}}})$ and $(\delta_{i_{\mathcal{FV}}},\delta^{\perp}_{i_{\mathcal{FV}}})$, which can lead to severe cluster deformation. Secondly, the mapping function is applied to the angular dimensions, the discontinuity of the derivative of the function can cause the convex-clusters to be mapped into non-convex clusters in the Cartesian coordinate system. In order to avoid deformations and to preserve maximal similarity with the original clustering geometry, it is desired to use a more linear functions with continuous derivatives. Also, in order to provide maximal separation among clusters and to concentrate clusters into disjoint non-overlapping clusters, it is beneficial to map the regions between two target points into a region with equal or even smaller range in the Cartesian coordinate system. In other words, it is desired to keep the samples close to the centroids in the mapped space as much as possible. This property is achieved by using nonlinear functions. However, there is a trade-off between the level of concentration and the linearity, which needs to be carefully addressed. In order to avoid deformation while providing maximal separation, an optimized mapping function is proposed in this section.

In order to accommodate satisfy the above-mentioned requirements, it is desired to find a function, which satisfies the following mathematical conditions:

- the function is differentiable everywhere (continuous first derivative);
- the derivative of the function is small at the target points, which correspond to the centroids of clusters in the original and transformed space, i.e. $\delta_{i_{\mathcal{S}\mathcal{V}}}$ and $\delta_{i_{\mathcal{F}\mathcal{V}}}$ in Fig. XXX;
- the derivative of the function is large at the boundaries of two regions (point $(\epsilon_{i_{sv}}, XXX)$) in Fig. XX.1.3;

Therefore, we propose to use the basis function p, which satisfies the aforementioned conditions. why you call it basis function. Basis function has a different meaning in linear analysis. I suggest you use the term basic function with adjustable parameters. Each target point is associated with a basic function. The function p is composed of two constitutive functions: $h(x)$ and $g(x)$, respectively defined in two regions: i) from the target point to the upper boundary point, ii) from the lower boundary point to the next target point. [RZ: show h(x) and g(x) in Fig. 1.3.]. Before defining the two functions $h(x)$ and $g(x)$, we need to define the boundaries between two consecutive target points ($\delta_{i_{\mathcal{S}\mathcal{V}}}$,$\delta^{\perp}_{i_{\mathcal{S}\mathcal{V}}}$) and ($\delta_{i_{\mathcal{F}\mathcal{V}}}$,$\delta^{\perp}_{i_{\mathcal{F}\mathcal{V}}}$), in which these functions are defined (i.g. $\epsilon_X XXX$). We simply choose the midpoint as the boundary points. For instance, we have $\epsilon = (\delta_1 + \delta_2)/2$ [RZ:Write the correct equation] as stated in Equi. 1.7. The lower and upper boundary points are denoted respectively as (γ,$\gamma^{\perp}$) and (ϵ,$\epsilon^{\perp}$). To ensure the continuity of the mapping function f, the boundaries and target points should satisfy:

$$(\epsilon_{i_{\mathcal{SV}}}, \epsilon^{\perp}_{i_{\mathcal{SV}}}) = (\gamma_{i_{\mathcal{FV}}}, \gamma^{\perp}_{i_{\mathcal{FV}}}) \quad = (\frac{\delta_{i_{\mathcal{SV}}} + \delta_{i_{\mathcal{FV}}}}{2}, \frac{\delta^{\perp}_{i_{\mathcal{SV}}} + \delta^{\perp}_{i_{\mathcal{FV}}}}{2}) \tag{1.7}$$

The two piece-wise functions $h(x)$ and $g(x)$ are defined as use the correct name: the inverse of logit/sigmoid function as follows:

$$\begin{aligned} K_h &= \frac{\epsilon^{\perp} - \delta^{\perp}}{e^{\alpha(\epsilon-\delta,0)^{+}} - 1} \\ h(x) &= K_h[e^{\alpha(x-\delta,0)^{+}} - 1] + \delta^{\perp} \end{aligned} \tag{1.8}$$

$$\begin{aligned} K_g &= \frac{\gamma^{\perp} - \delta^{\perp}}{e^{\alpha(-\gamma+\delta,0)^{+}} - 1} \\ g(x) &= K_g[e^{\alpha(\delta-x,0)^{+}} - 1] + \delta^{\perp} \end{aligned} \tag{1.9}$$

RZ: 1-include parameters in the function definitions: $h(x; \alpha, \epsilon, ...)$. 2-check functions, there are some typos like using different parameters in the denominator and nominator.

In order to investigate the satisfaction of the above-mentioned conditions, we first verify that these functions pass through the target points. More specifically, $g()$ should pass through points XXX and $h()$ through XXXX ($\delta_{i_{\mathcal{SV}}}$,$\delta^{\perp}_{i_{\mathcal{SV}}}$) and ($\delta_{i_{\mathcal{FV}}}$,$\delta^{\perp}_{i_{\mathcal{FV}}}$).

Once, we confirm that that the basic function $h(x)$ has the desired properties, we fit this

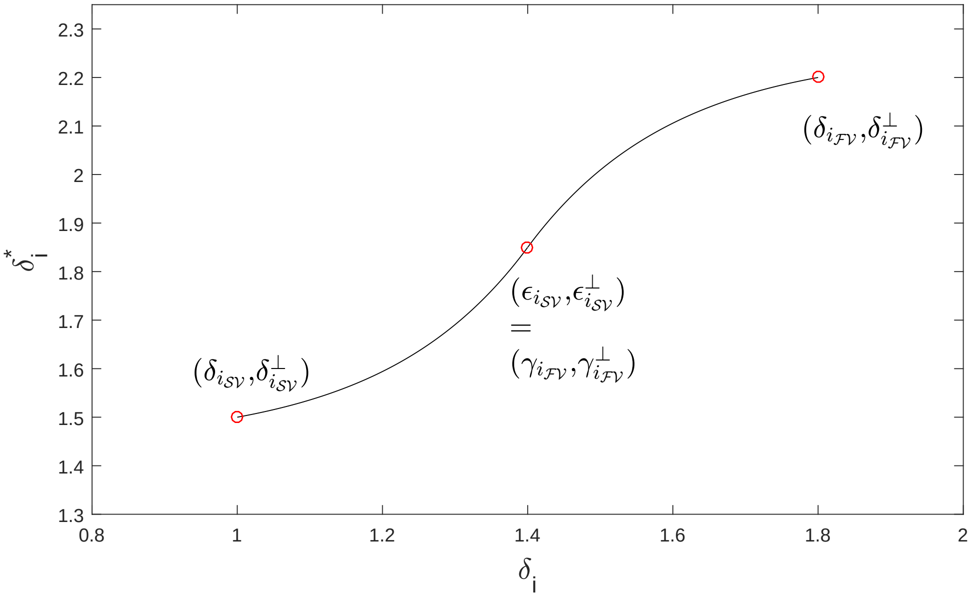

Figure 1.3: Optimized Piece-wise Interpolated Function p. I understand this figure, but is also misleading since the target points are at two ends and the boundary is in middle (the opposite is better). To solve this issue you can plot two adjacent regions and use tags and labels to show the target points δ and edge nodes/boundary nodes ϵ.

function with appropriate parameters to all regions. The final function is a smooth and continuous function, which passes through all target points $(\delta_{i_{\mathcal{SV}}},\delta^{\perp}_{i_{\mathcal{SV}}})$, $(\delta_{i_{\mathcal{FV}}},\delta^{\perp}_{i_{\mathcal{FV}}})$ as well as the two extreme boundary points as shown in Fig.1.4.

In the training process,data samples with abnormal labels belonging to all patients in DS1 and the personalized normal cluster for one patient are used to determine the mapping function for one corresponding patient. In the predicting process, the hyper-the the feature vector of new ECG sample in the spherical coordinate system is calculated and the mapping function is applied on its hyper-spherical coordinates to yield the transformed feature vector. After this step, we calculate the transformed data samples in the Cartesian coordinate system, which further fed into the personalized classification stage defined by Eq. **??** to generate the corresponding type of yellow alarm. RZ: later make sure to use a consistent notation for equations Eq. or Equi. with or without parentheses.

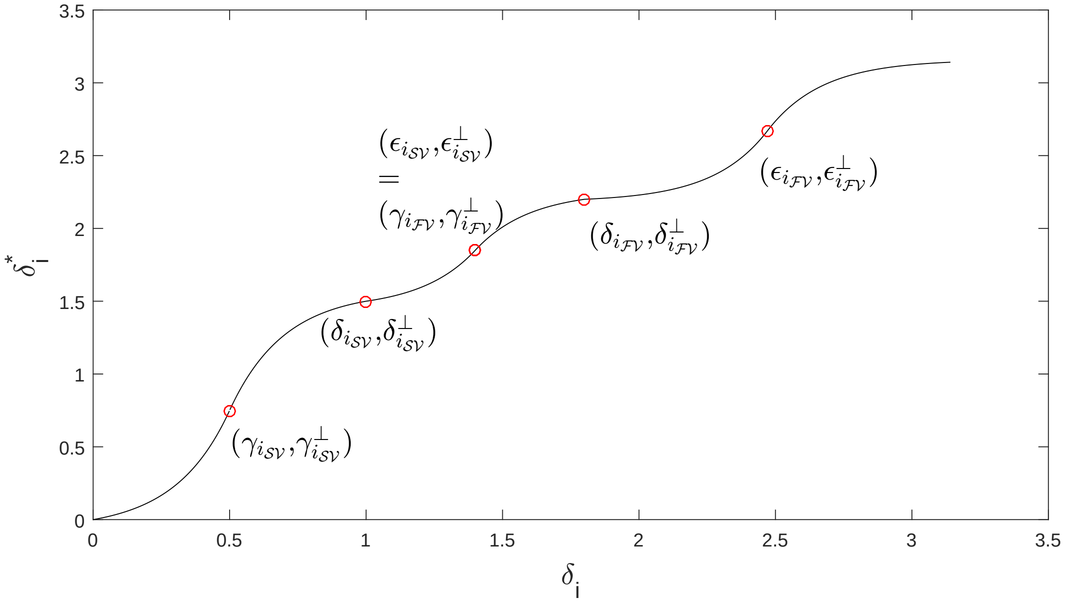

Figure 1.4: Optimized Mapping Function f.

1.6 Experimental Results

In this section, the performance of the proposed method is evaluated in terms of two aspects. We first analyze the classification performance of the system and then present the comparative results with respect to other representative ECG classifiers. Furthermore, the classification results are partitioned into two sets: red alarms generated by the global classifier and the final labels by combining the yellow and red alarms. In this way, the impact of personalized classifier on the final labels can be revealed. Finally, the prediction power of the proposed method in terms of providing precaution hints for upcoming red alarms is evaluated.

1.6.1 Classification Performance

The experimental results are assessed in terms of classification performance of 4 AAMI ECG classes using the test subset of MITBIH Arrhythmia DS2. Originally, DS2 contains 15357 samples after feature extraction. While training the personalized classifier, the first 20% of the normal samples serve as initialization set for the personalized dynamic normal cluster. not clear: Therefore, all samples before the last initialization normal sample should be excluded from the test set for each records. Consequently, the actual test set contains 12414 samples in total consisting of 10105 type-N, 1702 type-V, 508 type-S and 99 type-F samples.

To present the result, we select the weighted k-Nearest Neighbors method with $k = 10$ as our choice of global classifier because it is one of the low-complexity methods with a relatively good classification accuracy [REF].The parameter α, used in Equi .XXX in the deviation detection module is set to 1 for test purpose. if there are more parameters you can define here.

Table 1.1 summarizes the cumulated confusion matrix for all records in the test set. In order to compare the result of global classifier (red alarms) and combined results (final labels including both red and yellow alarms), the sample numbers are presented in the following format: $combined(globaled)$RZ: change to final label (primary label by global classifier). In order to measure the classification performance, we adopt three metrics: accuracy(Ac), sensitivity(Se), specificity(Sp), as proposed in [12, 13, 16]. All three metrics are calculated based on the true positive TP, false positive FP, false negative FN and true negative TN in a binary confusion matrix, where we combined all abnormality classes into a one abnormal class/ or where one class is the specific abnormality class and all other abnormality and normal classes combined into one class, whichever is correct.. Therefore all four metrics are

Table 1.1: Cumulated Confusion Matrix for All Records in DS2. The numbers are XXX ... .

		Ground Truth			
		N	V	S	F
Result	N	9255(10076)	21(38)	72(90)	1(5)
	V	657(22)	1678(1663)	8(2)	9(7)
	S	71(6)	3(1)	417(416)	0(0)
	F	122(1)	0(0)	11(0)	89(87)

calculated for each class by converting the 4x4 matrix to a 2x2 matrix.

While cumulative classification results are demonstrated in Table 1.1, the robustness of the proposed method should be evaluated based on the performance variation over 22 test records in DS2. Further, medians and IQRs (interquartile range) for each metric and each class are included in Table 1.2 to represent the robustness of proposed methods. RZ: Is this what you mean: The robustness of the system is asses in terms of the variation between the performance of different types. The system is more robust if this variation is lower. In Table 1.2, we observe that among all abnormal classes It is ok to use abnormal class, but since we used abnormality class, better to be consistent throughout the write-up., the proposed method demonstrates a stable performance on class V but a lower stability for classes S and F.

As MITDB is widely used to verify ECG classifier performance, we compared the proposed system with five significant methods proposed in the literature. According to AAMI standards, the performance of ECG classification should be evaluated over the binary classifiers applied to *Ventricular (V)* versus *non-V* types and *Supraventricular (S)* versus *non-S* types.

Table 1.2: Classification Performance and Within-Set Variation of the Proposed System.

statistics	N			V			S			F		
	Ac	*Se*	*Sp*	*Ac*	*Se*	*Sp*	*Ac*	*Se*	*Sp*	*Ac*	*Se*	*Sp*
cumulated	92.4	91.59	95.93	94.38	98.59	93.71	98.67	82.09	99.38	98.85	89.9	98.92
median	94.45	92.21	95.42	96.17	99.55	95.71	99.38	80.65	99.84	99.11	90.91	99.11
IQR	6.33	10.08	11.91	5.17	1.64	8.62	1.76	19.35	0.61	1.58	23.33	1.49

Table 1.3: performance comparison between the proposed method and five benchmark algorithms in terms of V and S classification accuracy using 11 common records in MITDB.

Methods	VEB			SVEB		
	Ac	Se	Sp	Ac	Se	Sp
Proposed	96.6	98.2	92.4	98.63	88.89	99.41
Hu *et al.* [12]	94.8	78.9	96.8	N/A	N/A	N/A
de Chazal *et al.* [9]	96.4	77.5	N/A	N/A	N/A	N/A
Jiang and Kong [15]	98.8	78.9	96.8	97.5	74.9	98.8
Ince *et al.* [16]	97.9	90.3	98.8	96.1	81.8	98.5
Kiranyaz *et al.* [17]	98.9	95.9	99.4	96.4	68.8	99.5

For methods proposed in the literature, the same evaluation metrics are commonly used applied to records from MITDB. To standardize the metrics, we select 11 ECG records which are common among all 5 methods and compare the median of each classification metrics over these 11 records. The comparison results are presented in Table 1.3. Generally speaking, the proposed method shows a higher sensitivity for both types V and S. Especially for type S, the proposed method shows an advantage over all three metrics compared to the five reference methods.

1.6.2 Prediction Performance

As an important feature of the proposed method, yellow alarms triggered by the personalized classifier indicate a higher probability of observing subsequent abnormalities. In order to verify this functionality, all beats following a yellow alarm of a specific type is investigated to asses the chance of upcoming red alarms of different types. This process is repeated for yellow alarms of all types. We only account for the first abnormality type which occurs after the yellow alarm. As we used confusion matrix to evaluate the classification accuracy, the performance of prediction can be summarized by a confusion matrix with the 3 abnormal types. Probabilities of observing a certain type of abnormal beat after a yellow alarm is

Table 1.4: predictive probability versus prior probability without windowing

		# of predicted ground truth			% of predicted ground truth		
		V	S	F	V	S	F
yellow alarm	V	467	122	14	**77.45**	20.23	2.32
	S	36	15	0	70.59	**28.41**	0
	F	40	60	5	38.10	57.14	**4.76**
total		543	197	19	**71.54**	**25.96**	**2.50**

calculated using the prediction confusion matrix and compared to the prior probability of observing the abnormality of the same type. This process is formulated in the following two equations:

$$\begin{aligned} P(\hat{y}_{k+i} = X_r | \hat{y}_k = X_y) &= \frac{\text{\# of } y_{k+i} = X \text{ after } \hat{y}_k = X_y}{\text{\# of true alarms after } \hat{y}_k = X_y} \\ P(\hat{y}_{k+i} = X_r) &= \frac{\text{\# of true alarm of type } X \ (y_k = X)}{\text{\# of all true alarms}} \end{aligned} \tag{1.10}$$

The prediction power of each abnormality type is evaluated by comparing $P(\hat{y}_{k+i} = X_r | \hat{y}_k = X_y)$ and $P(\hat{y}_{k+i} = X_r)$. A shown in Table 1.4, the probability of observing a certain type of abnormalities after a yellow alarm is higher than its prior probability and this fact is consistent for abnormality types. For example, without knowing the type of a yellow alarm, the probability of observing a type V sample is 71.54%, while the probability of observing a type V sample after observing a yellow alarm of a type V is 77.45% (XXX% higher than the prior probability). The improvement are consistent among all three types of abnormalities but the system shows a stronger prediction power for type S.

In the above analysis we consider the first subsequent red alarm regardless of the time passes since the preceding yellow alarm. In order to study the impact of the timing window (the time between the yellow alarm and the subsequent red alarm), we also studied a window of 10

Table 1.5: predictive probability versus prior probability within 10 beats' window

		# of predicted ground truth			% of predicted ground truth		
		V	S	F	V	S	F
yellow alarm	V	290	85	12	**74.94**	21.96	3.10
	S	22	13	0	62.86	**37.14**	0
	F	29	37	6	40.28	51.39	**8.33**
total		341	135	18	**69.03**	**27.32**	**3.64**

consecutive samples following a yellow alarm. Similarly, the prior and posterior probabilities are compared to evaluate the performance of the prediction capacity as shown in Table.1.5. RZ: We can generate more insightful results for the journal paper, like a curve to show the prediction improvement vs time, etc...

Compared with the result without windowing, the prediction performance within a 10-beat window shows that the proposed algorithm can better predict the occurrence of abnormalities if a certain timing window is used. Especially for type S, the probability of observing a sample of type S within 10 beats after a yellow alarm of type S is 27.32%, i.e. the posterior probability rises to 37.14%. With almost 10% increase, it is proved that the yellow alarm types are informative. The results shows that same improvements are made within the 10-sample window as well. In general, the predicting performance are promising, indicating the efficiency of personalized classifier and deviation analysis. We believe that this concept is worthy of further more in-depth studies for different physiological signals.

1.7 Summary of Contributions

We have such a section in Chapter 3. Add a short section here and at the end of Chapter 2 for consistency. Make tis section quite short: one or two para at most.

Bibliography

[1] C. J. Murray and A. D. Lopez, "Measuring the global burden of disease," *New England Journal of Medicine*, vol. 369, no. 5, pp. 448–457, 2013.

[2] D. Lloyd-Jones, R. J. Adams, T. M. Brown, M. Carnethon, S. Dai, G. De Simone, T. B. Ferguson, E. Ford, K. Furie, C. Gillespie, *et al.*, "Heart disease and stroke statistics2010 update," *Circulation*, vol. 121, no. 7, pp. e46–e215, 2010.

[3] W. H. Organization, "Cardiovascular diseases (cvds)," 2017.

[4] S. C. Smith, R. Jackson, T. A. Pearson, V. Fuster, S. Yusuf, O. Faergeman, D. A. Wood, M. Alderman, J. Horgan, P. Home, *et al.*, "Principles for national and regional guidelines on cardiovascular disease prevention: a scientific statement from the world heart and stroke forum," *Circulation*, vol. 109, no. 25, pp. 3112–3121, 2004.

[5] E. Besterman and R. Creese, "Waller–pioneer of electrocardiography.," *British Heart Journal*, vol. 42, no. 1, p. 61, 1979.

[6] B. E. Kreger, L. A. Cupples, and W. B. Kannel, "The electrocardiogram in prediction of sudden death: Framingham study experience," *American heart journal*, vol. 113, no. 2, pp. 377–382, 1987.

[7] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo, "Clustering ecg complexes using hermite functions and self-organizing maps," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 7, pp. 838–848, 2000.

[8] G. K. Prasad and J. Sahambi, "Classification of ecg arrhythmias using multi-resolution analysis and neural networks," in *TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region*, vol. 1, pp. 227–231, IEEE, 2003.

[9] P. de Chazal, M. O'Dwyer, and R. B. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," *IEEE Transactions on Biomedical Engineering*, vol. 51, pp. 1196–1206, July 2004.

[10] R. Ceylan, Y. Özbay, and B. Karlik, "A novel approach for classification of ecg arrhythmias: Type-2 fuzzy clustering neural network," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6721–6726, 2009.

[11] S. Osowski, L. T. Hoai, and T. Markiewicz, "Support vector machine-based expert system for reliable heartbeat recognition," *IEEE transactions on biomedical engineering*, vol. 51, no. 4, pp. 582–589, 2004.

[12] H. H. Yu, P. S., and J. T. W., "A patient-adaptable ECG beat classifier using a mixture of experts approach," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 9, pp. 891–900, 1997.

[13] P. de Chazal and R. B. Reilly, "A patient-adapting heartbeat classifier using ecg morphology and heartbeat interval features," *IEEE Transactions on Biomedical Engineering*, vol. 53, pp. 2535–2543, Dec 2006.

[14] M. Llamedo and J. P. Martínez, "An automatic patient-adapted ecg heartbeat classifier allowing expert assistance," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 8, pp. 2312–2320, 2012.

[15] W. Jiang and S. G. Kong, "Block-based neural networks for personalized ECG signal classification," *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1750–1761, 2007.

[16] T. Ince, S. Kiranyaz, and M. Gabbouj, "A generic and robust system for automated patient-specific classification of ecg signals," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1415–1426, 2009.

[17] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ecg classification by 1-d convolutional neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 3, pp. 664–675, 2016.

[18] P. W. Wilson, R. B. DAgostino, D. Levy, A. M. Belanger, H. Silbershatz, and W. B. Kannel, "Prediction of coronary heart disease using risk factor categories," *Circulation*, vol. 97, no. 18, pp. 1837–1847, 1998.

[19] M. A. Whooley, P. de Jonge, E. Vittinghoff, C. Otte, R. Moos, R. M. Carney, S. Ali, S. Dowray, B. Na, M. D. Feldman, *et al.*, "Depressive symptoms, health behaviors, and risk of cardiovascular events in patients with coronary heart disease," *Jama*, vol. 300, no. 20, pp. 2379–2388, 2008.

[20] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati, "Classification of ecg signals using machine learning techniques: A survey," in *Computer Engineering and Applications (ICACEA), 2015 International Conference on Advances in*, pp. 714–721, IEEE, 2015.

[21] S. Kiranyaz, T. Ince, and M. Gabbouj, "Personalized monitoring and advance warning system for cardiac arrhythmias," *Scientific Reports*, vol. 7, no. 1, p. 9270, 2017.

[22] L. S. Green, R. L. Lux, C. W. Haws, R. R. Williams, S. C. Hunt, and M. J. Burgess, "Effects of age, sex, and body habitus on QRS and ST-T potential maps of 1100 normal subjects.," *Circulation*, vol. 71, no. 2, pp. 244–253, 1985.

[23] R. Hoekema, G. J. H. Uijen, and A. van Oosterom, "Geometrical aspects of the interindividual variability of multilead ecg recordings," *IEEE Transactions on Biomedical Engineering*, vol. 48, pp. 551–559, May 2001.

[24] A. Houghton and D. Gray, *Making sense of the ECG: a hands-on guide.* CRC Press, 2014.

[25] G. A. Ng, "Treating patients with ventricular ectopic beats," *Heart*, vol. 92, no. 11, pp. 1707–1712, 2006.

[26] A.-A. EC57, "Testing and reporting performance results of cardiac rhythm and st segment measurement algorithms," *Association for the Advancement of Medical Instrumentation, Arlington, VA*, 1998.

[27] A. L. Goldberger, L. A. 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," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[28] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[29] J. Chen and A. Razi, "A predictive framework for ecg signal processing using controlled nonlinear transformation," in *Biomedical & Health Informatics (BHI), 2018 IEEE EMBS International Conference on*, pp. 161–165, IEEE, 2018.

[30] J. Chen, H. Peng, and A. Razi, "Remote ECG monitoring kit to predict patient-specific heart abnormalities," *Journal of Systemics, Cybernetics and Informatics*, vol. 15, no. 4, pp. 82–89, 2017.

[31] B. N. Singh and A. K. Tiwari, "Optimal selection of wavelet basis function applied to ecg signal denoising," *Digital signal processing*, vol. 16, no. 3, pp. 275–287, 2006.

[32] N. V. Thakor, J. G. Webster, and W. J. Tompkins, "Estimation of qrs complex power spectra for design of a qrs filter," *IEEE Transactions on biomedical engineering*, no. 11, pp. 702–706, 1984.

[33] Y. Lian and P. C. Ho, "Ecg noise reduction using multiplier-free fir digital filters," in *Signal Processing, 2004. Proceedings. ICSP'04. 2004 7th International Conference on*, vol. 3, pp. 2198–2201, IEEE, 2004.

[34] Y.-W. Bai, W.-Y. Chu, C.-Y. Chen, Y.-T. Lee, Y.-C. Tsai, and C.-H. Tsai, "Adjustable 60hz noise reduction by a notch filter for ecg signals," in *Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE*, vol. 3, pp. 1706–1711, IEEE, 2004.

[35] O. Sayadi* and M. B. Shamsollahi, "Ecg denoising and compression using a modified extended kalman filter structure," *IEEE Transactions on Biomedical Engineering*, vol. 55, pp. 2240–2248, Sept 2008.

[36] K. Park, K. Lee, and H. Yoon, "Application of a wavelet adaptive filter to minimise distortion of the st-segment," *Medical and Biological Engineering and Computing*, vol. 36, no. 5, pp. 581–586, 1998.

[37] N. Nikolaev, Z. Nikolov, A. Gotchev, and K. Egiazarian, "Wavelet domain wiener filtering for ecg denoising using improved signal estimate," in *Acoustics, Speech, and Signal*

Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on, vol. 6, pp. 3578–3581, IEEE, 2000.

[38] S. Poungponsri and X.-H. Yu, "An adaptive filtering approach for electrocardiogram (ecg) signal noise reduction using neural networks," *Neurocomputing*, vol. 117, pp. 206–213, 2013.

[39] V. X. Afonso, W. J. Tompkins, T. Q. Nguyen, and S. Luo, "Ecg beat detection using filter banks," *IEEE transactions on biomedical engineering*, vol. 46, no. 2, pp. 192–202, 1999.

[40] D. Sadhukhan and M. Mitra, "R-peak detection algorithm for ecg using double difference and rr interval processing," *Procedia Technology*, vol. 4, pp. 873–877, 2012.

[41] S. Mehta and N. Lingayat, "Svm-based algorithm for recognition of qrs complexes in electrocardiogram," *IRBM*, vol. 29, no. 5, pp. 310–317, 2008.

[42] R. V. Andreão, B. Dorizzi, and J. Boudy, "Ecg signal analysis through hidden markov models," *IEEE Transactions on Biomedical engineering*, vol. 53, no. 8, pp. 1541–1549, 2006.

[43] J. P. Martínez, R. Almeida, S. Olmos, A. P. Rocha, and P. Laguna, "A wavelet-based ecg delineator: evaluation on standard databases," *IEEE transactions on biomedical engineering*, vol. 51, no. 4, pp. 570–581, 2004.

[44] S. Banerjee, R. Gupta, and M. Mitra, "Delineation of ecg characteristic features using multiresolution wavelet analysis method," *Measurement*, vol. 45, no. 3, pp. 474–487, 2012.

[45] Z. Zidelmal, A. Amirou, M. Adnane, and A. Belouchrani, “QRS detection based on wavelet coefficients,” *Computer methods and programs in biomedicine*, vol. 107, no. 3, pp. 490–496, 2012.

[46] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis.* Cambridge university press, 2004.

[47] B. Schölkopf, C. J. Burges, and A. J. Smola, *Advances in kernel methods: support vector learning.* MIT press, 1999.

[48] T. Evgeniou, M. Pontil, and T. Poggio, “Regularization networks and support vector machines,” *Advances in computational mathematics*, vol. 13, no. 1, p. 1, 2000.

[49] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods.* Cambridge university press, 2000.

[50] C. A. Coello Coello, “Mopso: A proposal for multiple objective particle swarm optimization,” *Proc. Congr. Evolutionary Computation (CEC’2002), Honolulu, HI, 5*, vol. 1, pp. 1051–1056, 2002.

[51] J. E. Alvarez-Benitez, R. M. Everson, and J. E. Fieldsend, “A mopso algorithm based exclusively on pareto dominance concepts,” in *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 459–473, Springer, 2005.

[52] L. Blumenson, “A derivation of n-dimensional spherical coordinates,” *The American Mathematical Monthly*, vol. 67, no. 1, pp. 63–66, 1960.

[53] G. W. Stewart, *Matrix algorithms volume 1: Basic decompositions*, vol. 2. Society for Industrial and Applied Mathematics, 1998.

[54] G. Arfken, “Gram-schmidt orthogonalization,” *Mathematical methods for physicists*, vol. 3, pp. 516–520, 1985.