ECG Analysis

Michalis Smirnakis

MSc by Research in Pattern Analysis and Neural Networks

ASTON UNIVERSITY

September 2006

Acknowledgements

I wish to express my heartiest gratitude to my supervisor, Dr. Evans for his invaluable advice, guidance and constructive suggestions for completing this thesis. I am particularly indebted to my family for their continued concern and support. A special appreciation is due to Mouts, Kouro, Paidaki and Carnation for their understanding, patience and encouragement throughout this year of study. Grateful acknowledgments is accorded to the Public Welfare Institution Bakala for providing me with a scholarship to pursue this study.

ASTON UNIVERSITY

ECG Analysis

MICHALIS SMIRNAKIS

MSc by Research in Pattern Analysis and Neural Networks, 2006

Thesis Summary

The aim of this project is the PhysioNet's and Computers in Cardiology challenge of 2003, specifically the building of a model of ST segments, based on component analysis, and the creation of a classifier that can categorize these segments to ischaemic or non-ischaemic. Two techniques were used to visualize the data, plots of Principal Components and Neuroscale, with various datasets. However, these techniques performed poorly because they did not separate the two classes in two dimensions. These datasets were also used for classification. Using only the extracted Principal Components the results were poor when compared with the other entries of the challenge. Adding ΔST and ΔT into our dataset the results improved remarkably. The best classifier created with that dataset had accuracy of 89.1%. Finally, using Automatic Relevance Determination method we conclude that ΔT is the most significant variable in classifying ischaemia.

Keywords: ECG, ischaemia, Long Term ST Database, PhysioNet, MLP, Neuroscale, Bayesian inference, ARD,PCA

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This project is inspired by the 2003 and 2005 PhysioNet and Computers in Cardiology challenges. Computers in Cardiology is an international organization and PhysioNet is an on-line forum, part of the Research Resource for Complex Physiological Signals project. Physionet provides free access to PhysioBank which is a collection of different databases of physiologic signals.

This project is an effort to answer affirmatively to the opening question posed in the beginning of the introduction of the PhysioNet and Computers in Cardiology challenge of 2003, "Is it possible to tell the difference between transient ST changes in the ECG that are due to myocardial ischaemia, and those that are not?".

An Electrocardiogram (ECG) is a graph which records the electrical voltage in the heart in the form of a continuous strip graph. Each beat that is recorded in the ECG can be separated in different sections such as P wave, QRS complex, ST segment and T wave. Figure 1.1 depicts the segments of a heart beat as it is shown in an ECG.

Electrodes are used to measure the voltage of the heart and produce the ECG. Nine electrodes are placed at certain points on the human body and produce an ECG. According to their places, the form of an ECG will be different. There are twelve different ways to place the electrodes for an ECG and form signals, which are referred to as leads.

The leads are separated into two groups bipolar and unipolar. For bipolar leads, a single positive and a single negative electrode are utilized and the electrical potentials between them are measured. For unipolar leads, a single positive electrode and the average of two negative electrodes are used to produce the ECG signal [Boutkan, 1972]. There are three bipolar leads in an ECG, lead I, lead II and lead III. In lead I the

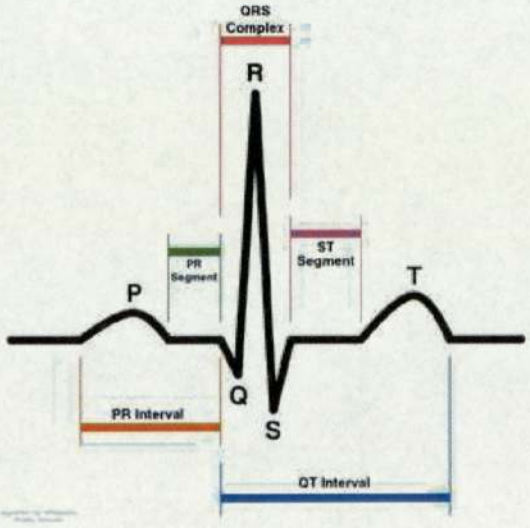


Figure 1.1: Segments of a normal heart beat as it occurs in ECG. (*http* : *//en.wikipedia.org/wiki/Electrocardiogram*)

electrodes measure the potential difference between the left arm and right arm, lead II measures the potential difference between the left foot and right arm and lead III measures the potential difference between left foot and left arm.

There are nine unipolar leads that are separated into two groups. The Augmented Limb Leads aVr, aVl and aVf. In these leads one can measure the potential difference between one of the mentioned electrodes and the mean potential of the remaining two. For instance aVr=R-(L+F)/2, where R is the electrode which is placed in the right arm, L is the electrode which is placed in the left arm and F is the electrode that is placed in the foot of the patient. If we choose the positions of the electrodes of the limb leads to be placed near the torso (Mason Likar positions), then these leads are called modified. The modified leads gives more accurate results for ST deviation [Feldman et al., 2005]. The other six leads are the precordial leads v1, v2, v3, v4, v5, and v6. They measure the potential difference between the V electrode and the mean of the other three electrodes Vlead=V-((R+L+F)/3). The positions the electrodes are placed for each lead can be shown in figure 1.2.

Myocardial ischaemia is one of the most common fatal diseases of the western industrial world. It is a heart problem that is caused by the lack of oxygen and nutrients to the contractile cells (muscles), and it is often difficult to detect from a routine ECG.

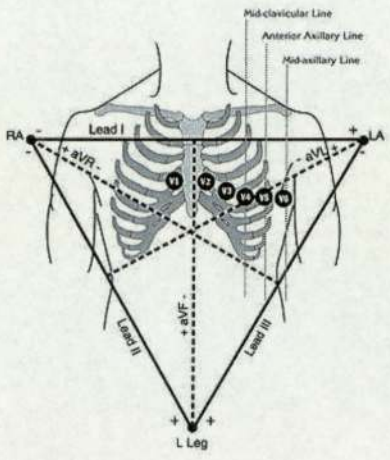


Figure 1.2: Positions of the body that the ECG's electrodes are fixed to perform a 12 lead ECG. ($http://medlib.med.utah.edu/kw/ecg/mml/ecgtorso.gif$)

There are several methods which are employed to detect myocardial ischaemia such as coronary angiography, which is the most accurate method, and exercise test. The former is an X-ray examination of the blood vessels or chambers of the heart. The latter shows whether there is lack of blood supply in the arteries that feed the heart, during the test. However coronary angiography and the exercise test are either very expensive or very exhausting. These are the most important reasons for applying the above mentioned methods only to high-risk patients.

Alternatively, ST segment analysis of the ECG record is cheaper and requires less effort from the patient. Despite this, we should bear in mind that ST elevations and depressions are caused by various factors, including changes in heart rate, the position of the subject, noise in the ECG and many other which make the classification a difficult procedure. Nevertheless classifying ST segments still remains cheaper and easier to apply when it is compared to coronary angiography and exercise test. Also the volume of ECG data that is recorded nowadays is large enough to provide the researchers with a satisfactory amount of ECGs that can be used for that kind of research. Consequently there is need to develop a classifier of ischaemic or non-ischaemic episodes based on ECG records. For the above reason PhysioNet and Computers in Cardiology created the challenge of 2003, so as to encourage the researchers to create a classifier of ischaemia based only on the ECG. This is also the aim of this thesis.

The remainder of this thesis is set out as follows. This chapter gives the background

	Non-ischaemic classified episodes	Ischaemic classified episodes
Non-ischaemic episodes	True negatives	False positives
Ischaemic episodes	False negatives	True positives

Table 1.1: Example of a confusion matrix

information that is relevant to the project. In the second chapter there is a literature review including papers of the competition and other relevant papers in chronological order. Then the third chapter consists of a description of the Long Term ST Data Base (LTSTDB) and also information about the data extraction and preprocessing. The results from visualization and classification techniques, using as dataset the extracted Principal Components, are analyzed in chapter four. The fifth chapter includes the results of analysis using as dataset not only Principal Components, but also more features that have been inspired from the literature survey. The final chapter presents conclusions, remarks and suggestions for future work.

1.1 Measures for quantification

The results of a classifier can be illustrated as a Confusion matrix. A Confusion matrix is a matrix that represents the true classification versus the results of the classifications from our algorithm [Bamia, 2003]. Table 1.1 depicts an example of a confusion matrix.

The measures that are most used widely to quantify the results of a classifier for such problems, which are based on the confusion matrix, are accuracy, sensitivity and specificity. These can be calculated through the formulas below [Bamia, 2003]:

$$accuracy = \frac{true \quad positives + true \quad negatives}{number \quad of \quad events} \times 100\% \tag{1.1}$$

Accuracy is the percentage of the correct classified predictions.

$$sensitivity = \frac{true \quad positives}{true \quad positives + false \quad negatives} \times 100\% \tag{1.2}$$

Sensitivity is the portion of the real positive cases of all the classified positive cases.

$$specificity = \frac{true \quad negatives}{false \quad positives + true \quad negatives} \times 100\% \tag{1.3}$$

Specificity is the percentage of the real negative cases of all the classified negative cases.

Neither sensitivity, nor specificity, alone, can describe correctly the results of a classification. That is because if we create a classifier that labels all the events positive or negative, the sensitivity and specificity will be 100% respectively. But such a classifier would be useless. So sensitivity and specificity are to be used together as quantifying measures of a classifier.

These three measures are used to quantify the results of the entries of the competition and the papers which are represented in the second chapter.

Chapter 2

Information about the data

In this chapter there is an extended description of the LTSTDB, giving information about its structure and the files that it contains. Also there is an analytic description of the way the data was extracted from LTSTDB and other features that were extracted and used to visualize and classify the ST events.

2.1 Description of Long Term ST Database

The LTSTDB was a project that began in 1995 with the aim of contributing to the field of automatic detection of ischaemia using the results of an ECG. Until then the ESTDB was used. The ESTDB contains 90, 2 hour records fully annotated beat by beat. This database does not contain enough patterns of ST changes that are not caused by ischaemia, which is the most common phenomenon in the real world ECG. For that reason the LTSTDB was developed.

The LTSTDB contains 86 records (21-24 hour ambulatory ECG) from 80 patients. From these 43 are available from Physionet as a training set for the competition. Out of these records 34 are two-lead ECG and the other 9 are three-lead ECG. [Jager et al., 2003] PhysioNet called these leads, lead 0, lead 1 and lead 2.

Each record contains twelve files which are the signal, annotations and some files which are needed for SEMIA to work. SEMIA is a computer program which is used for semi-automatically labeling of events in ECG records. Also for the needs of the competition one more file was provided with the annotations of the type of each episode

in each record: ischaemic or non-ischaemic respectively. The frequency of the digitized signal was 250 Hz.

According to PhysioNet an ST episode was identified using the following three criteria:

1. the ST deviation, the difference between the ST level and the baseline, reached 50 μV;

2. the ST deviation must be equal or greater than a threshold value Vmin for at least for a time period Tmin;

3. the episode ends when the ST deviation is smaller than 50 μV for 30 sec.

There were three different types of annotation that were used for the location of the ST episodes according to different values of Vmin and Tmin. These annotations were in the .sta, .stb, .stc files. The annotations that used in this project were these of the .stb files. There are 1772 events according to these annotations where 1369 were non-ischaemic and the other 403 were ischaemic.

The other annotation files that are relevant with the measurements of the ST segment were the .stf files which provided the ST deviation. Also there were the .16a files that contained the J points of each beat based on a 16 second moving average.

2.2 Extraction of the Data

The WaveForm DataBase (WFDB) tools package was used initially to transform the signal and the annotation files to a format that is compatible with Matlab. WFDB tools are a set of wrappers that are provided by PhysioNet to convert the binary annotation and signal files to Matlab variables. Firstly, the J points of the first beat for every ST event were extracted, using the annotation files. The length of the ECG signal that was initially extracted was from each J point to the next R-peak. These segments include the ST segment of the first beats. Moreover these also include the R-peak of the next beat. That R-peak would dominate the results of PCA. A sample of 100 episodes was chosen randomly from the training set to calibrate the length of

the data. Based upon to the measurements of this sample the end of each signal was set to 80 milliseconds before the extracted R-peak.

Repeating this procedure for all the events for the three leads we have constructed three matrices, each of which contained 1772 rows. The elements of the rows of the matrix depicting lead 2 were equal to zero for the episodes that came from two lead ECG records. Fifty percent of each matrix was selected as a training set randomly. From the rest 50% half of it used as validation set and the other 25% were used as test set. The events for each data set were selected randomly from the 1772 events of each lead. The events which compose each dataset, training, validation and test set, were the same for each of the three leads. This way of data separation allows episodes of a specific patient to be used in training, validation and test set. As conclusion generalization of the results to datasets which are based on different patients are risky.

Chapter 3

Literature Review

3.1 Introduction

The aim of this chapter is to give a brief description of the work that other researchers have done until now to the field of automatic detection of ischaemia. This will include six different approaches. The first two appeared before the challenge of 2003 and are about the detection of ischaemia, using as dataset European ST Data Base (ESTDB) instead of the LTSTDB which has been collected more recently. Three of the other papers are considered to have participated the challenge of 2003 or 2005. Finally is mentioned a paper which was an effort to improve upon the classifier of the winning paper of the 2003 challenge. The papers are presented in chronological order.

3.2 Stamkopoulos et al 1998

In this work a non-linear approach for feature extraction is used. The authors assume that the features that are important for detection ischaemia using ECG cannot be extracted from linear functions of the data [Stamkopoulos et al., 1998]. For that reason PCA wasn't used for feature extraction and dimensionality reduction of the data. Stamkopoulos et al implemented a Non Linear Principal Components Analysis (NLPCA) method that was developed by Kramer [Stamkopoulos et al., 1998] in chemical reprocessing. NLPCA assumes that for a n dimension input vector $\mathbf{x} = [x_1, \ldots, x_n]'$ the underlying feature vector is $\phi = [\phi_1, \ldots, \phi_n]'$ where $x_1 = f_1(\phi_1), \ldots, x_n = f_n(\phi_n)$

and f_i is a non-linear function.

To find these features an auto-associative neural network was trained using back propagation algorithm. The input of this network was the ST segments consisted from 40 samples, starting from the J point, of a 250 Hz sampling frequency. J point is the point where the ST segment of each beat begins. To classify these beats, a Radial Basis Function (RBF) neural network was used, minimizing the mean square error:

$$G = E \parallel x - g(h(x)) \parallel^2, \tag{3.1}$$

where g is the coding function from R^n to R^m and h is the decoding function from R^m to R^n. Thirty-four of the ninety files of the European ST Database were used for classification. As training set for the RBF neural network, only the normal beats were used. Twenty-five percent of the normal beats were used as training set. The sensitivity of that algorithm was approximately 75% and the specificity 80%. Compared with other works up until 1998, these results were better than those of classifiers that used as feature extraction method Principal Components Analysis (PCA)

3.3 Papaloukas et al 2002

In this paper a neural network produced a classifier with sensitivity of 86% and accuracy of 87% [Papaloukas et al., 2002]. Firstly the QRS complex of each beat was detected. Thereafter there was a filter applied so as to minimize or eliminate noise distortion, such as A/C interference, baseline wandering or electromyographic contamination. Moreover during the preprocessing phase they also pinpointed, with edge detection, the location of J point. Each data sample contained 100 observations after the J point. If the beat ended before the 100_{th} observations then the input was padded with zeros. The end of each beat was set as $Beat_{end} = QRS + 0.6RR - 60$. QRS is the location of the R-peak and RR is the duration between the R-peak before the J point and the R-peak after the J point. That procedure was repeated for all the events. After finishing data extraction, PCA used to reduce the dimensionality of the data. Then the data projected by PCA were fed in the neural network. The neural net comprised of four input units, a hidden layer with 10 sigmoidal units and an output layer with one linear output unit.

Researcher	accuracy	sensitivity	specificity
Langley et al 2003	90.7%	-	-
Zimmerman et al 2003	79.1%	80.6%	78.9%
Povinelli 2005[1]	50.3%	2%	98.5%
Povinelli 2005[2]	54%	74.6%	33.5%

Table 3.1: Table with the results of entries in the challenges of 2003 and 2005.

So as to train the Artificial Neural Network (ANN) they used a training set of 11 hour of two channel ECG recordings from ESTDB. Three medical experts annotated independently each beat in three different groups: Normal, Ischaemic and artifact. If there was discrepancy then the annotation was performed with total agreement with each other.

The outputs of the neural network, that is the classification of each beat, were led to a sliding adaptive window [Papaloukas et al., 2001]. It is a technique that is used to identify the data windows which will be classified. Thirty second intervals that contain more than 75% ischaemic beats were used to produce the ischaemic windows. Thereafter all the possible ischaemic windows were merged so as to obtain the ischaemic episodes in each recorded lead. Furthermore the detected episodes in every lead are also merged and the overall ischaemic episodes were defined.

It is worth to mention that the results of that classifier were better than those of Stamkopoulos et. al. who used NLPCA as feature extraction technique.

3.4 Langley et al 2003

This was the winning paper of the 2003 challenge. It achieved the best accuracy from all the entries of the competition [Langley et al., 2003]. As we can observe in Table 3.1 Langley's algorithm has achieved the better accuracy from all the other entries for both challenges of 2003 and 2005.

Langley's team classified the ST events using a rule-based classifier with ΔST as main variable. ΔST is the difference between the voltage of the ST in the time that is examined and the baseline level for the same time interval. The algorithm is initialized

with the detection of the value of the ΔST for the beginning of the event T_s from LTSTDB files. After that they compared the value of ΔST with a threshold value V_{thres}. In other words a smaller value than V_{thres} meant a non-ischaemic episode. On the other hand if the value of ΔST was greater than the threshold then proceed to the next step of the algorithm. Following the previous there was the identification of the time that the event was ended, T_e. They set T_e to be the beginning of a period of time where ΔST would be smaller than V_{thres} for a specific time interval T_{thres}. Then they extracted from LTSTDB the values for ΔST for the interval from T_s to T_e. The next step was to find if there was another time interval equal or bigger than T_{min} that ΔST remained bigger than V_{min} which is another threshold of the algorithm. If such an interval existed then the event was classified as an ischaemic episode otherwise as non-ischaemic. Figure 3.1 [Langley et al., 2003] depicts the thresholds that were used for Langley's algorithm for an example event. The steps of the algorithm are illustrated in Figure 3.2. [Langley et al., 2003]

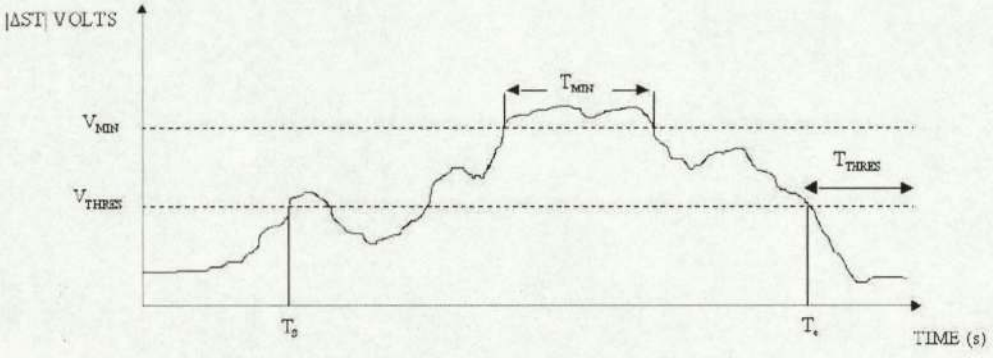


Figure 3.1: Langley's algorithm thresholds.

Two different features were used to optimize the algorithm. The Mahalanobis distance of the ST level from the five first Principal Components and the number of the ST crossovers. Both different optimizations were based to the already classified ischaemic events. The events classified as ischaemic before were reclassified using these two features.

The difference from the initial algorithm was that, in the ST crossovers optimization, instead of using V_{min} and T_{min} to specify an event as ischaemic or not, two new thresholds were used, N_{cross} and V_{cross}. V_{cross} is a threshold value like V_{min} and N_{cross}

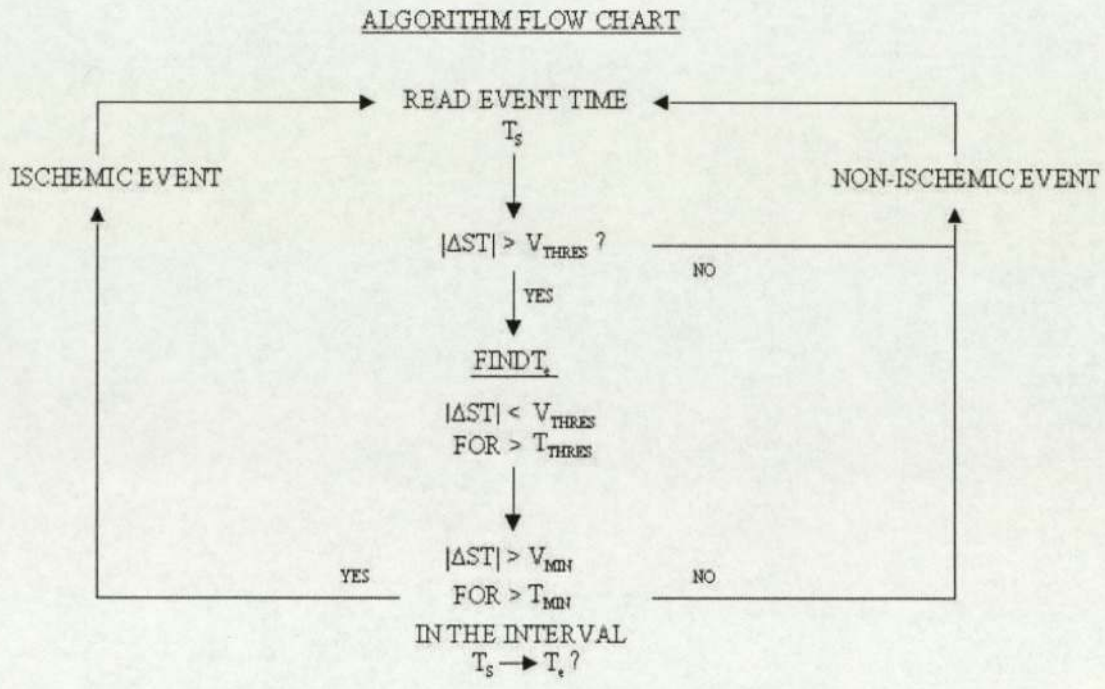


Figure 3.2: Flow chart of Langley et al algorithm.

is the number of times that ΔST crossed that threshold. If ΔST crossed the threshold value for at least N_{cross} times, an event was characterized as ischaemic

The same reasoning was used for the optimization which was based in the Mahalanobis distance of the ST level and the principal components. In essence the substituted V_{min} and T_{min} with V_{KLT} and T_{KLT} respectively. Where KLT is the Karhunen-Loeve transform coefficients which are equivalent to the Principal Components. The same as previously, if ΔST was greater than the threshold value V_{KLT} for a period of time T_{KLT} the event was classified as ischaemic.

These two efforts to optimize the algorithm, using the ST crossovers and the Mahalanobis distance from the Principal Components, were unsuccessful. The initial algorithm achieved the best results. For the training set the accuracy was 91.1% was achieved with sensitivity 99% and specificity 88.8%. An overall accuracy of 90.7% was achieved using PhysioNet's test set. Also it is mentioned from the authors that their algorithm is better when identifying ischaemic episodes than non-ischaemic.

3.5 Zimmerman et al 2003

This paper was an entry in the 2003 challenge. The Reconstructed Phase Space (RPS) method was used to extract the features of the data. Firstly the J points of eight beats before the beginning of the episode and the J points eight beats after the beginning of the episode were extracted. As input data 100 samples (400ms) of each J point were used. After that procedure the data embedded in a RPS of the following format:

$$x_n = [x_{n-(d-1)\tau}, \ldots, x_{n-\tau}, x_n] \tag{3.2}$$

where n=1+(d-1)τ,..., d is the dimension of the RPS and τ is the time lag.

The embedded dimension and the time lag had been chosen empirically to be 6 and 5 respectively. For classification a mixture of 25 Gaussians and a Bayesian maximum likelihood classifier was used. The accuracy on the validation set for that algorithm was 79.1%. The sensitivity and the specificity of the validation set was 80.6% and 78.9% respectively [Zimmerman et al., 2003]. There was a big difference between the results of the validation set and the results of PhysioNet's test set. The accuracy, the sensitivity and the specificity dropped to 55.7%, 63.8% and 49.9% respectively.

3.6 Zimmerman, Povinelli 2004

Another paper that we are going to describe briefly, is an attempt to improve the algorithm that was proposed in [Langley et al., 2003] using Support Vector Machines (SVM). The first step of this algorithm was to implement Langley's algorithm. If the result was non-ischaemic then that event was classified as non-ischaemic. If the result was ischaemic then some new features were extracted from the database. These features were: the Maximum ST deviation which is maximum value of the ΔST variable in the time interval between T_s and T_e, the Mean ST deviation which is the mean of ΔST for the same time interval and the initial ST deviation which is the value of ΔST at the beginning of the episode. ΔST , T_s and T_e are the same variables as in Langley's algorithm.

After extracting these new features a SVM was used for classification. The results were not the expected since they did not improve the results of Langley et al. For the

authors' test set the accuracy was 95.6%, the sensitivity 99% and the specificity 92.3% [Zimmerman and Povinelli, 2004]. The best results that could improve the specificity, which seemed to be Langley's algorithm weak point, were 94.3%. But this led to a reduction in specificity from 99% to 80.5% and the accuracy from 95.6% to 89.3%.

3.7 Povinelli 2005

The next paper we are going to describe was an entry in the 2005 competition. Two methods were used to define the features which were used for classification: Reconstructed Phase Space (RPS) and the KLT coefficients of the ST segment. For both methods a 400ms signal after each J point of the previous 30s of the starting event was used as input data. Given that time series the points were reconstructed according to Equation (2.2). Then a sixteen component GMM was employed using the EM algorithm. For the KLT approach the ST segments of the previous 30s from the beginning of the event were used to extract the KLT coefficients. Also for that approach the same type of mixture model and the Bayesian classifier were used.

For the RPS method the accuracy was 50.3%; the sensitivity was very low at 2% but there was a big percentage of the correct classified non-ischaemic events 98.5%. The results of that classifier were not far from "tossing a coin" i.e 50% accuracy. The accuracy for the KLT approach was better, 54%. The results in sensitivity were improved compared with the RPS method since the sensitivity was 74.6%. That affected the specificity of the model since there was a big drop compared with the RPS method since specificity was 33.5%. [Povinelli, 2005]

3.8 Summary

Many different approaches have been adopted by researchers that are involved in automated detection of ischaemia. Some of them have applied neural networks, some rule-based classifiers or time series techniques for their algorithms. Table 3.2 summarizes the feature extraction techniques and the types of classifiers that have been used. For neural network classifiers the one which used PCA and multilayer perceptrons achieved the best results. For that reason these techniques constitute the basis of the

Researcher	Feature extraction methods	Classifier
stamkopoulos et al	Non-linear PCA	RBF neural network
Papaloukas et al	PCA	Multi-layer perceptron
Langley et al	PCA, ΔST	Fuzzy Logic Algorithm
Zimmerman et al	RPS	GMM
Zimmerman, Povineli	PCA, ΔST	SVM
Povineli	PCA, RPS	GMM

Table 3.2: Summary of the feature extraction techniques and classifiers that used by other researchers.

classifiers that will be used later.

Chapter 4

Feature extraction and classification using Principal Component Analysis

In this chapter we are going to present the results of the visualization and classification analysis. The dataset which is used in this chapter consists of the Principal Components of each lead. Firstly we are going to describe PCA briefly, and then the procedure of Principal Components extraction. The rest of this chapter can be separated into two parts. The first presents the results of the data visualization, and the second the results of the classification.

For consistency reasons, in this chapter there are some brief definitions of the methods that are used. For a more detailed and comprehensive description of them can been found at the corresponding books and articles.

4.1 Principal Component Analysis

Principal Component Analysis (PCA) is a linear transformation that projects the data to a new basis. PCA is used to reduce the dimensionality of multivariate data [Karlis, 2004]. We select the first n largest principal components that have a good representation of our data with respect to a small loss of information. Since PCA is a linear transformation, for k variables the k principal components we will have the following:

$$\begin{aligned}
\mathbf{Y}_1 &= a_{11}\mathbf{X}_1 + a_{12}\mathbf{X}_2 + \ldots + a_{1k}\mathbf{X}_k \\
\mathbf{Y}_2 &= a_{21}\mathbf{X}_1 + a_{22}\mathbf{X}_2 + \ldots + a_{2k}\mathbf{X}_k \\
&\ldots \\
\mathbf{Y}_k &= a_{k1}\mathbf{X}_1 + a_{k2}\mathbf{X}_2 + \ldots + a_{kk}\mathbf{X}_k
\end{aligned}$$

In a matrix form the principal components can be written as $\mathbf{Y} = \mathbf{AX}$, where $\mathbf{Y}, \mathbf{X}$ are $k \times 1$ vectors and $\mathbf{A}$ is a $k \times k$ matrix. Since the principal components are uncorrelated we are searching for an orthonormal matrix $\mathbf{A}$ which diagonalizes the covariance matrix of $\mathbf{Y}$ ($\mathbf{S}_y$). So:

$$\begin{aligned}
\mathbf{S}_y &= \frac{1}{n-1}\mathbf{Y}\mathbf{Y}^T \\
&= \frac{1}{n-1}(\mathbf{AX})(\mathbf{AX})^T \\
&= \frac{1}{n-1}\mathbf{A}\mathbf{X}\mathbf{X}^T\mathbf{A}^T
\end{aligned} \tag{4.1}$$

$\mathbf{XX^T}$ is the covariance matrix of $\mathbf{X}$ ($\mathbf{S_x}$) which is a square symmetric matrix. The solution to that equation is for A to be the eigenvectors of the covariance matrix $\mathbf{S_x}$

4.1.1 Extraction of Principal Components

From the dataset, which had been extracted as described in the previous chapter, the mean of the training set was subtracted from the training set and also from validation and test set. That is because the mean of each dataset (training, validation and test set) would dominate the results of PCA. After that the eigenvalues and their corresponding eigenvectors of the training set were extracted. Then the number of principal components, that should represent the data for each lead was decided. Afterward all the data set was multiplied with the appropriate eigenvectors. So these three different datasets were used as an initial input in the classifiers.

There are several methods which are used to decide how many principal components should be used for dimensionality reduction. Two of the most common methods used are the percentage of the variance that the principal components can represent and plotting the eigenvalues. An acceptable percentage for the method that uses the portion

number of eigenvalues	lead 0	lead 1	lead 2
2	81.6%	85.53%	67.20%
3	87.38%	89.83 %	75.15%
4	90.28%	91.62%	81.57%
5	92.94%	93.26%	85.14%
6	94.02%	94.27%	88.30%
7	95.03%	95.14%	90.59%

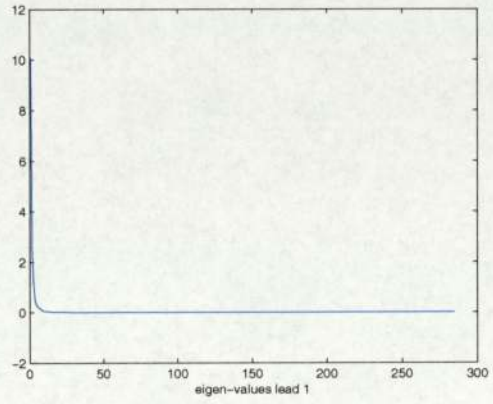
Table 4.1: Percentage of the variance that principal components can explain for the three leads.

of the variance as criterion, is of ninety or ninety-five percent of the variance. For the eigenvalues plot, the point that after that the eigenvalues tend to be zero can be set as an accepted number of Principal Components. Netlab's function PCA is used to extract the eigenvalues and the eigenvectors of the covariance matrix of the data.

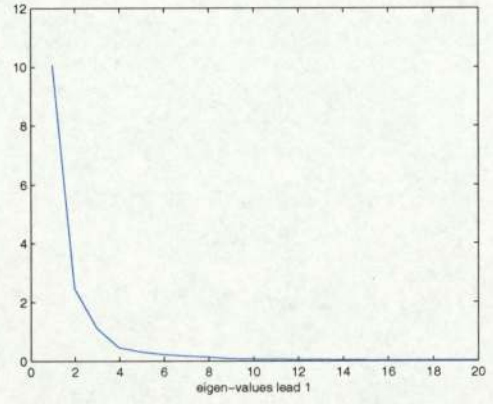
The eigenvalues from the training data of lead 1 are plotted in Figure 4.1(a). It is not possible from this graph to select the point that principal components begin to be close to zero. For that reason it is useful to plot only a part of the first few eigenvalues. A second plot with the twenty larger eigenvalues was used. The results are shown in Figure 4.1(b). From Figure 4.1(b) we can see that four or seven principal components are appropriate to represent the data. The figures for lead 0 and lead 2 are also two-fold like the previous results. The figures of these leads are in Appendix A.

The above method is not an objective way to decide how many components should be kept. That is because the number of the principal components that finally will be used depends to the opinion of the researcher. For that reason the criterion used for the decision is the percentage of the variance the principal components explain, as described previously. The accepted percentage of the variance that eigenvalues should explain was set empirically to ninety. In Table 4.1 there is the percentage of the variance that different number of principal components can explain for the three leads.

From Table 4.1 we can see that four principal components should be kept for the first two leads. For the third lead using the same criterion and according to the results of table 4.1 seven principal components should be used. The number of the principal components that were extracted for the first two leads is smaller than the number



(a) All eigenvalues



(b) First twenty eigenvalues

Figure 4.1: Plot of eigenvalues.

of Principal Components that PhysioNet provide with the LTSTDB. In lead 2 more Principal Components were needed to reach the threshold of 90%. An explanation for that phenomenon is the small amount of data that used to extract the Principal Components of that lead.

4.2 Data Visualization

In this section we use the Principal Components that we had extracted using the above procedure, in order to separate the ischaemic and non-ischaemic episodes into two separate groups. Two techniques were utilized for that purpose. The first one involved plots of the Principal Components for each lead. This method uses linear transformations of the data (Principal Components) to project the data into two dimensions. The second one is Neuroscale. Neuroscale utilizes non-linear transformations and neural networks to project the initial data points into smaller dimension.

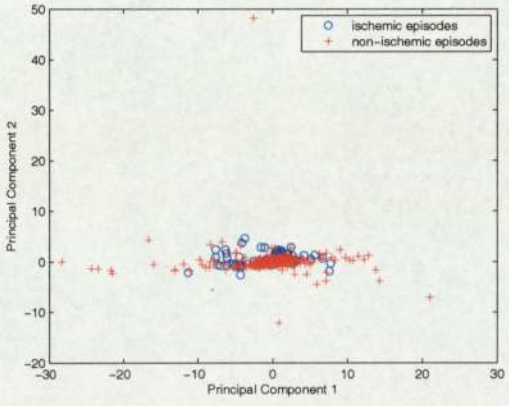
4.2.1 Plots of Principal Components

The results obtained by extracting the Principal Components were used for visualization. The figures of the first principal component versus the second one for the three leads can been seen in the Figure 4.2(a), 4.2(b) and 4.2(c) respectively. The figures for visualization between all principal components of each lead are included in Appendices B, C and D respectively.

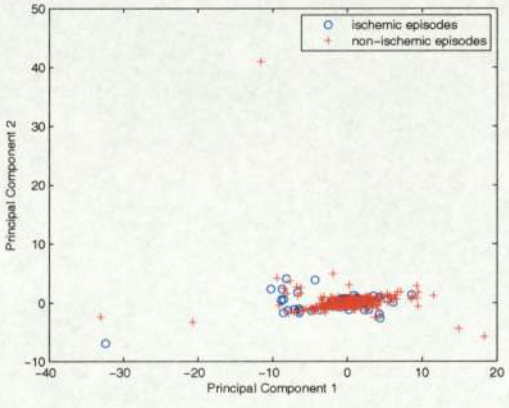
We observe that it is not possible to separate the two classes using linear functions. The ischaemic episodes (crosses) cannot be separated from the non-ischaemic episodes (circles) for any of the three leads. Moreover, whatever the combinations of principal components are, the results are still of poor quality. After that, we can conclude that we cannot separate the events into two dimensions using only a linear transformation of the initial data.

4.2.2 Neuroscale

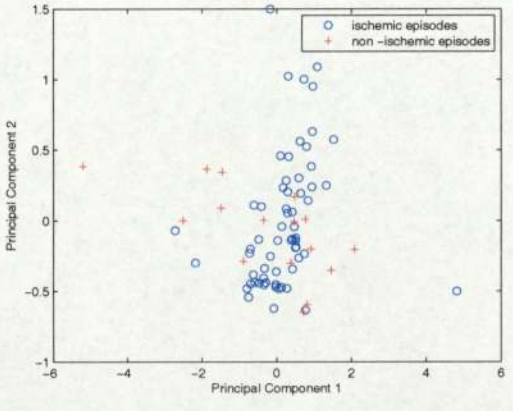
Since the results of the visualization using the principal components were not the expected ones, there was an attempt to separate the data using nonlinear functions. In



(a) Principal Component 1 vs Principal Component 2 for lead 0.



(b) Principal Component 1 vs Principal Component 2 for lead 1.



(c) Principal Component 1 vs Principal Component 2 for lead 2.

Figure 4.2: Plot of the first principal component versus the second for the three leads.

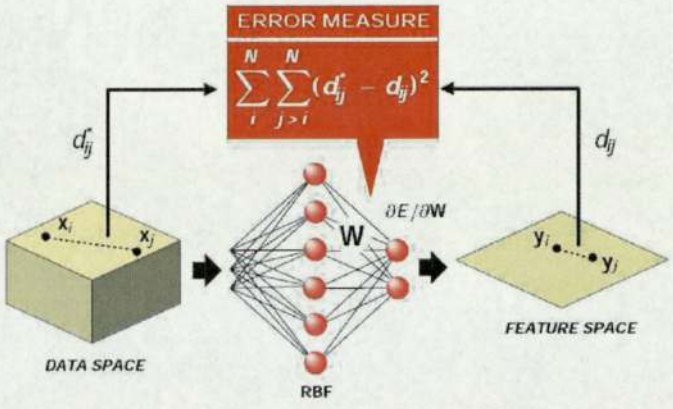


Figure 4.3: Neuroscale algorithm as it appears in [Lowe and Tipping, 1996].

contrast with PCA which is a linear transformation of the data, Neuroscale is a non linear topographic feature extraction and visualization technique. [Lowe and Tipping, 1996]

Neuroscale maintains the distances between the data points in the transformed space as similar as it is possible to those in the data space. That is feasible by using an error term of the form :

$$E = \sum_{q}\sum_{q>p}^{N}(d_{pq}^{*} - d_{pq})^2, \tag{4.2}$$

where d_{pq}^{*} is the Euclidean distance between the data points in the data space and has the form $\sqrt{(x_q - x_p)^T((x_q - x_p)}$ and $\sqrt{(y_q - y_p)^T((y_q - y_p)}$ is the distance in the projected space, d_{pq}. The projected space has smaller dimension than the initial dataset. Usually the dimensionality of the projected space is two. Figure 4.3 [Lowe and Tipping, 1996] illustrates the Neuroscale algorithm. Neuroscale is a *relatively supervised* method. That is because we don't know the target points in the space with the reduced dimensionality. A Radial Basis Function (RBF) neural network is trained to create the data of the projected space, with respect to the minimization of the error term.

4.2.3 Visualization using Neuroscale

The Principal Components of the initial dataset were used to train Neuroscale. Moreover the early stopping technique was used for regularization for all the RBF neural networks that were trained.

In essence the RBF neural network that had been used for the lead 1 had four input

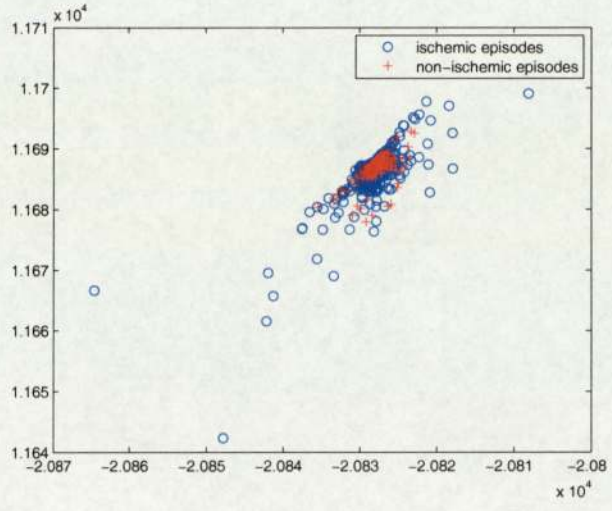


Figure 4.4: Results of using principal components.

units, twenty hidden units, two outputs and trained for 1200 iterations.

The results are depicted in Figure 4.4. The results for the other two leads were similar. There was overlapping between the two classes. The ischaemic and non-ischaemic events cannot be separated. In the Appendices E and F there are the figures with the results of Neuroscale for the other two leads.

The results weren't improved even with the use of the Neuroscale. The projected points could not be separated in two classes in the visualization plot.

4.3 Classification

The aim of this project is to create a classifier of ischaemic episodes using the ST segments of ECG. This section presents an initial attempt to create that classifier. The input data for that classifier were the extracted Principal Components of the initial ECG signal. The neural network that was selected for that purpose was the Multi-Layer Perceptron (MLP)

4.3.1 Multi-layer Perceptron

The MLP is a feedforward neural network. For N input units, H hidden units and K outputs the formula of the MLP is the following:

$$y_k = f\Big(\sum_{j=0}^{H} w_{kj}^{(2)} g\Big(\sum_{i=0}^{N} w_{ji}^{(1)} x_i\Big)\Big) \tag{4.3}$$

where y_k is the k^{th} output of the classifier, f the activation function of the output $w_{kj}^{(2)}$ are the weights of the j_{th} hidden unit of the k_{th} output, g is the hidden layer activation function, $w_{ji}^{(1)}$ are the weights of the i_{th} input of the j_{th} hidden unit and x_i is the i_{th} input. For classification problems usually a logistic $f(a_i) \equiv \frac{1}{1+exp(-a_i)}$ or a softmax $f(a_i) = \frac{\exp(a_i)}{\sum_j a_i}$ activation function is used. That because they converge faster and their encoding is between zero and one. The logistic activation function is used fr two class classification problems and softmax when the number of the classes is greater than two. To avoid overfitting the early stopping technique was applied. In this method many classifiers are trained with different number of hidden units and different number of iteration for optimization. Then the error of validation set for the classifiers with the same number of hidden units is compared. The classifier of each number of hidden units which have the smaller error is stored. Then these errors are compared and the classifier with the smaller error is chosen. For a detailed representation of the MLP and backpropagation the reader could refer to [Bishop, 1995].

4.3.2 Receiver Operating Characteristic curve

The outputs of the MLP using the logistic activation function are between zero and one. A question that arises from the above is how to determine efficiently a threshold that separates ischaemic from non-ischaemic episodes. An answer is the Receiver Operating Characteristic (ROC) curve. The ROC curve describes the trade off between the sensitivity (true positives) and one minus the specificity (false negatives). The points of the ROC curve of a classifier whose results are not better than a random process are situated in the bisector of the axes. A "mostly ideal" ROC curve has most of its points in the upper lefthand corner so the classifier has a big percentage of true positive classifications and a small number of wrong negative classifications.

	lead 0	lead 1	lead 2
accuracy	76.52	78.33	88.89
sensitivity	38.68	46.23	94.59
specificity	88.23	88.43	62.50

Table 4.2: Results on validation set using the principal components for the three leads.

	lead 0	lead 1	lead 2
accuracy	73.46	77.57	72.22
sensitivity	33.65	32.69	88.46
specificity	85.89	91.59	30.00

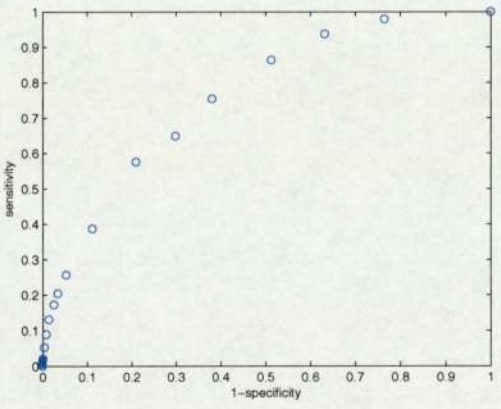
Table 4.3: Results on test set using the principal components for the three leads.

4.3.3 Results of classification using only the principal components as dataset

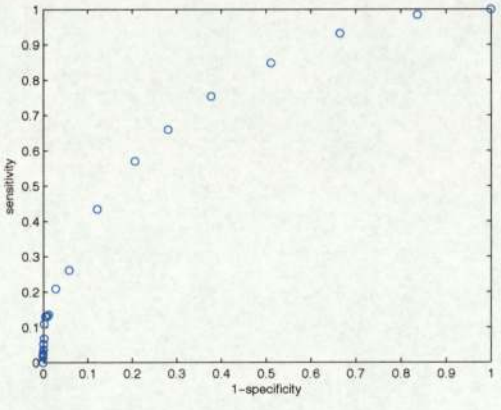
After the training procedure and the application of the early stopping technique an MLP was used with 4 input units, 10 hidden units, 1 output and trained for 1000 iterations for lead 0. The MLP which was used for lead 1 had 4 input units, 10 hidden units, 1 output and trained for 800 iterations. Finally for the third lead a MLP was trained for 700 iterations with 7 input units, 10 hidden units and 1 output. For the three leads the output activation function was a logistic one and for the optimization the scaled conjugate gradient algorithm was used. The threshold between the ischaemic and non-ischaemic events was determined using the ROC curve. Figures 4.5(a), 4.5(b) and 4.5(c) depict the results of the ROC curves for the three leads respectively. The threshold values that gives the best combinations of accuracy, sensitivity and specificity are 0.3, 0.3 and 0.25 for lead 0, lead 1 and lead 2 respectively.

The results of the classification for the validation and test set are seen in Table 4.2 and 4.3 respectively.

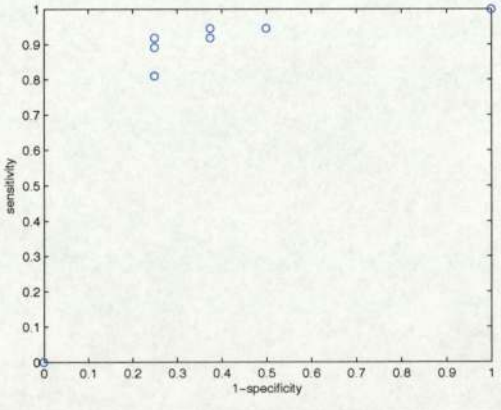
Comparing the two first leads, lead 0 and lead 1, we can observe that lead 1 has better results both in validation and test set. It worth noticing that both leads have a very low percentage of the ischaemic events that they can identify. The accuracy for both leads is greater than 70% instead of the small percentage of sensitivity. Lead 2 has inverse results. That is because of the nature of the dataset. The LTSTDB contained only nine records with three leads. The number of the events in lead 2 were



(a) ROC curve for lead 0



(b) ROC curve for lead 1



(c) ROC curve for lead 2

Figure 4.5: ROC curve for the three leads for the MLP with input the principal components.

164, with 122 of them ischaemic. From the 83 that were used for the training test the 64 of them were ischaemic and the rest non-ischaemic. The network trained had more ischaemic patterns to "match". That is the main reason that the results in lead 2 were more sensitive to identify ischaemic events.

4.4 Chapter conclusions

Two different techniques were applied for visualization, with similar results. Neither the plots of Principal Components, nor Neuroscale was able to provide a visualization of the data, with regards to their separation in two dimensions. The results for the classification weren't the expected ones since they performed worst than most of the entries, especially in sensitivity. We concluded that more features are needed to improve the data set. This is due to the fact the Principal Components alone weren't able to provide us with a classifier that could separate the two classes efficiently, in comparison to the results obtained by the other entries of the challenge.

Chapter 5

Extended feature extraction and classification using empirical features

The next step in our research after obtaining the results of Chapter 4, was the extraction of new features according to the literature review (Chapter 3). This chapter presents the visualization and classification results when these two features were added to the dataset. Firstly, there is a description of the way these features were extracted. Afterwards the results of Neuroscale for the new dataset are represented. Finally we analyze the results obtained by the classification procedure.

5.1 Feature extraction

More features were extracted inspired by the literature review [Langley et al., 2003]. These features were the ST deviation and the duration of the episode combined with knowledge we have for the ST deviation. Adopting Langley's notation these variables are referred as ΔST and ΔT respectively. ΔST is the difference between the ST level voltage and the baseline of the moment that the event begins. The values of ΔST were provided from PhysioNet. Measurements for the three leads were extracted. Also a part of Langley's algorithm was implemented to find ΔT. ΔT is the difference between T_e and T_s, where T_s is the beginning of each event. When ΔST became smaller than the

threshold value of the algorithm V_{thres}, the event was characterized as non-ischaemic from the beginning. Hence T_e is equal with T_s and consequently ΔT is equal to zero. If the value of ΔST was bigger than V_{thres}, the value of T_e could be determined from

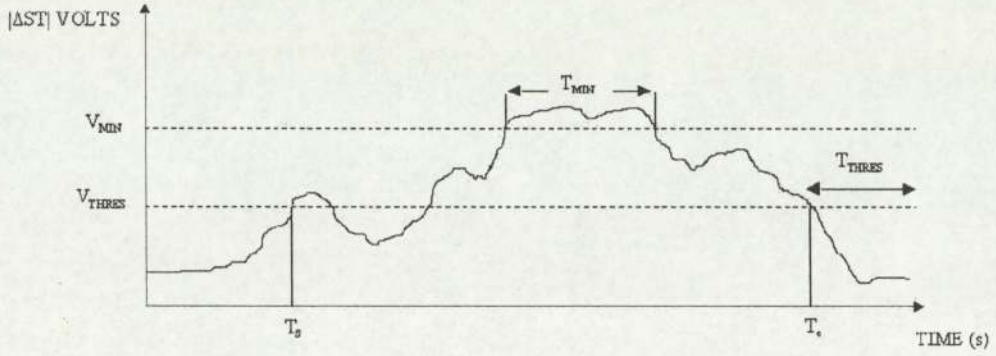


Figure 5.1: ΔT extraction procedure

the starting point of a time interval where the value of ΔST was smaller than V_{thres} for at least T_{thres} seconds. Figure 5.1 [Langley et al., 2003] depicts the procedure that used to determine ΔT for our algorithm. V_{thres} and T_{thres} were set to 50 μV and 40 seconds respectively.

5.2 Visualization

Two attempts were made to improve the visualization results of the previous chapter. Initially Neuroscale was trained using ΔST combined with the principal components. Afterwards Neuroscale was trained using all the extracted features, Principal Components, ΔST and ΔT.

5.2.1 Results of visualization using the new features

For the first attempt the RBF neural network was used to create the projected points for the lead 1 having five input units, thirty hidden units, two outputs and was trained for 1100 iterations.

The results of the projection are depicted in Figure 5.2. Unfortunately, the two classes are not separated again. The results for the two other leads, lead 0 and lead 2 are included in Appendixes E and F respectively.

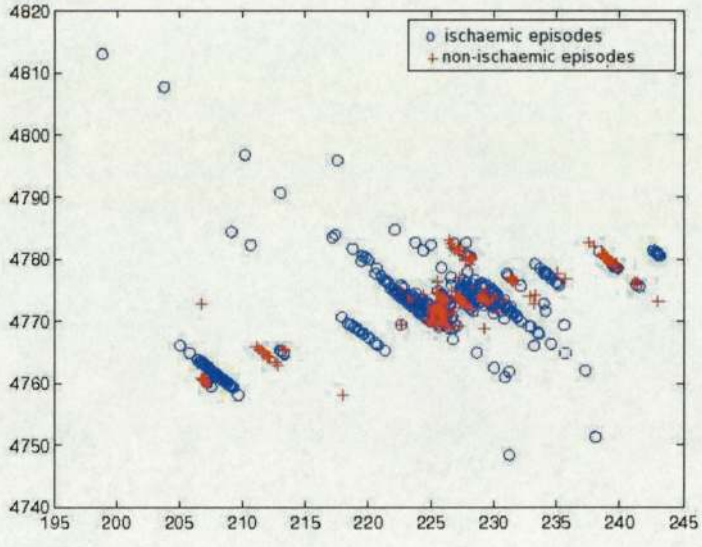


Figure 5.2: Results of using principal components and ΔST.

After the last results a final effort was employed to visualize the data into two dimensions. A Neuroscale model was trained using the previous dataset and also ΔT.

The RBF neural network that had been used to create the projected points for the lead 1 had six input units, thirty hidden units, two outputs and was trained for 600 iterations.

Figure 5.3 displays the results of Neuroscale. The overlapping between the two classes still remains, and the results are the same also for the other two leads. Figures with the above results can be found in the Appendices E and F respectively.

We notice that there is a structure in the results of in both attempts to visualize the data. An explain to that abnormality is depicted in the Histograms 5.4 and 5.5. In these figures we observe that ΔST and ΔT are two clustered variables.

ΔST can be separated in three groups and ΔT can be separated in two classes. At the first one belongs the majority of the observations and it is the first class of the histogram and the other class is formed from the rest data. We can observe that even after using the new features, we cannot separate the two classes in the projected space.

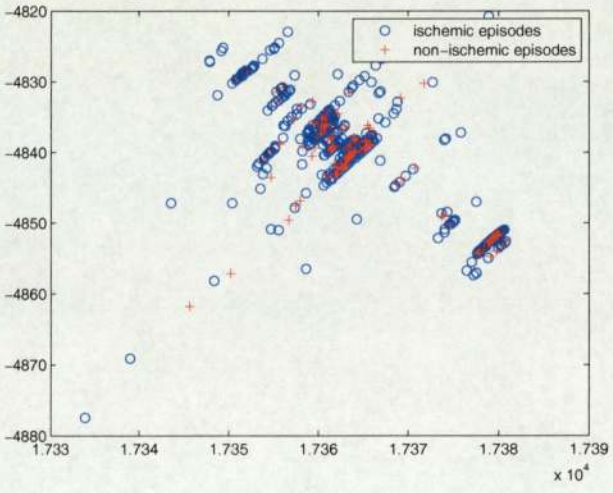


Figure 5.3: Results of using principal components,ΔST and ΔT.

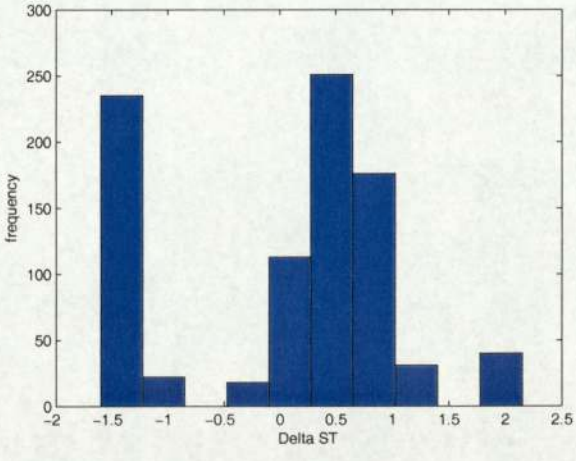


Figure 5.4: Empirical distribution of ΔST

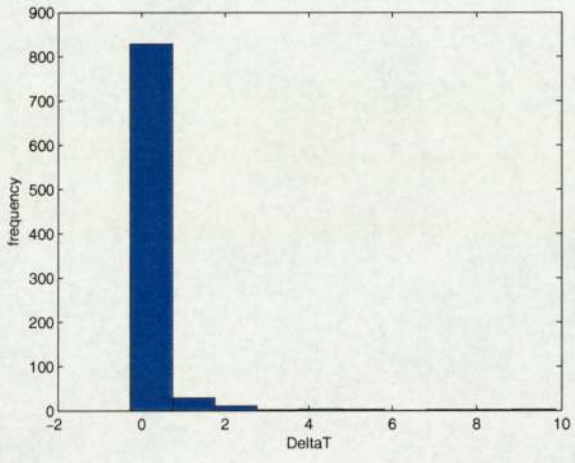


Figure 5.5: Empirical distribution of ΔT.

5.3 Classification Results

Using the new features, ΔST and ΔT, a new classifier was built. Afterwards Automatic Relevance Determination (ARD) was employed to find out which of the variables used, until now, were the most important. An MLP with Bayesian inference was needed for the ARD method, which was also used for classification.

5.3.1 Results of classification using the principal components and ΔT and ΔST

A new classifier was trained using the new dataset, which consisted of the Principal Components and the variables ΔT and ΔST. After the training procedure and the utilization of the early stopping technique an MLP was used with 6 input units, 10 hidden units, 1 output and trained for 1000 iterations for lead 0. The MLP which was used for lead 1 had 6 input units, 8 hidden units, 1 output and was trained for 1300 iterations. Finally for the third lead a MLP was trained for 1500 iterations with 9 input units, 20 hidden units and 1 output. For the three leads the output activation function was a logistic one and for the optimization the scale conjugate gradient algorithm was used. The ROC curve was used to determine the thresholds between the two classes. The results of the ROC curves for each lead are depicted in Figure 5.4(a), 5.4(b) and 5.4(c) respectively. The threshold values that gives the best combinations of accuracy,

	lead 0	lead 1	lead 2
accuracy	87.58	86.91	86.67
sensitivity	84.91	81.13	97.30
specificity	88.43	88.72	37.50

Table 5.1: Results of validation set using as dataset the principal components ΔT and ΔST for the three leads.

	lead 0	lead 1	lead 2
accuracy	84.90	87.19	77.78
sensitivity	71.15	82.69	84.60
specificity	89.19	88.59	60.00

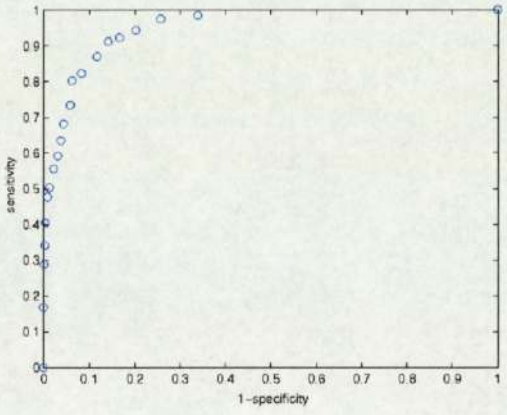
Table 5.2: Results of test set using as dataset the principal components ΔT and ΔST for the three leads.

sensitivity and specificity are 0.3, 0.35 and 0.3 for lead 0, lead 1 and lead 2 respectively. The results for the validation set and test set of the classification are in Tables 5.3 and 5.4 respectively.

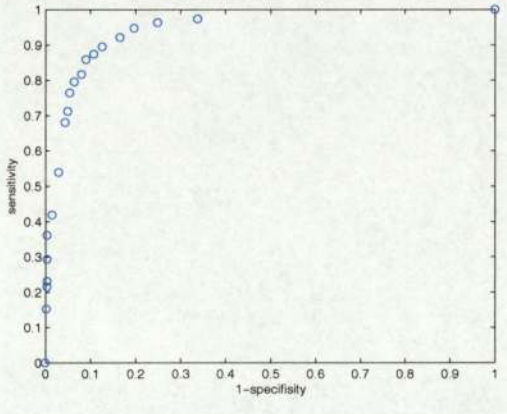
The results for the three leads are not as good as the results of the winning paper (accuracy 90.7%). Comparing the results of lead 0 and lead 1 with the other entries of the competition, we can conclude that they had been better except from the sensitivity of lead 0 in the test set which was smaller than that Zimmerman et al have achieved (79.1 %).

The results in the lead 0 are better in the validation set but in test set there is a big decline in the sensitivity from 84.91% to 71.15%. Lead 1 gives more trustworthy results since the results in test and validation set are quite similar. Using the two new variables the MLP became more sensitive to identify the real ischaemic events. In both leads the results were improved spectacularly comparing with the sensitivity of the classifiers with the data set containing only the principal components. Compairing the results of that classifier with the results of the Tables 4.2 and 4.3 we can observe that the sensitivity of the MLP for lead 0 increased from 38.68% to 84.91% for the validation set and from 33.65% to 71.15% for the test set. Similarly, the sensitivity of the MLP for lead 1 increased from 46.23% to 81.13% for the validation set and from 32.69% to 82.69% for the test set.

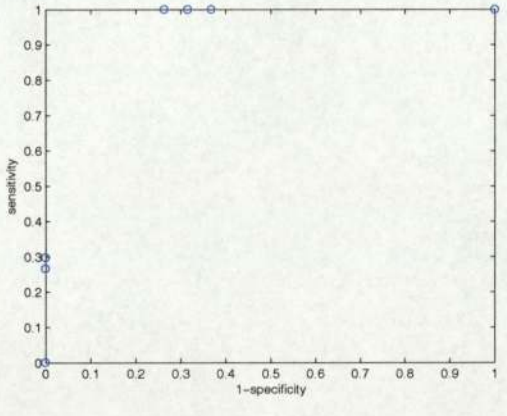
The new variables had a small effect in the results of the lead 2 which were very



(a) ROC curve for lead 0



(b) ROC curve for lead 1



(c) ROC curve for lead 2

Figure 5.6: ROC curve for the three leads for the MLP with input the principal components ΔST and ΔT.

similar with those of the classifier with the dataset with only the principal components. An interesting fact is that using the new dataset the very poor specificity result (37.5%) is at the validation set instead of the test set which was in the previous classifier at lead 2.

5.3.2 MLP with Bayesian Inference

We are interested in the features that are most important for classifying the data. The Automatic Relevance Determination method (ARD) was employed to identify these inputs. For that reason an MLP with Bayesian inference was trained. A Bayesian inference MLP combines the Bayes' theorem with MLP. Initially that is a quite strange idea since the nature of these techniques seems to be different. Bayesian inference can be used to avoid the over-fitting problem that neural networks have, by controlling the complexity of the model. At the beginning we should define the distribution of the weights of a MLP given the dataset. From Bayes' theorem and adopting Bishop's notation [Bishop, 1995] we have that :

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)}, \tag{5.1}$$

where $p(D|w)$ is the probability of the data given the weights and $p(w)$ is the prior distribution of the weight. Usually a Gaussian prior is used for the weight distribution. The form of that prior is $p(w) = \frac{1}{Z_w(\alpha)} \exp(-\alpha E_W)$, where $Z_w(\alpha)$ is a normalization factor of the form $\int P(D|w)P(w) \quad dw$ and E_W is a regularization factor of the form $E_W = \frac{1}{2}\|w\|^2 = \frac{1}{2}\sum_{i=1}^{W} w_i^2$. Since the parameter α determines the distribution of weights and biases it is called hyperparameter. Finally $p(D)$ is a normalization factor. For the classification problems a cross-entropy error function is used. The error function log likelihood becomes $p(D|w) = exp(-G(D|w))$ where G is the cross-entropy function. The function of the weights become :

$$p(w|D) = \frac{1}{Z_s} exp(-G - aEw)), \tag{5.2}$$

where Z_S is a normalization constant. Since we have defined the distribution of the weights of the classifier we should determine the form of the output distribution of our network. The output will have the following form:

$$p(C_1|x, D) = \int g(\alpha)p(\alpha|x, D)d\alpha, \tag{5.3}$$

where g is the logistic activation function. An approximation of that integral proposed by MacKay is the following:

$$p(C_1|x, D) = g(k(s)a_{MP}), \tag{5.4}$$

where $k(s) = (1+\frac{\pi s^2}{8})^{-1/2}$, α_{MP} is the hyperparameter α which maximize the posterior distribution of the weights, and s is the standard deviation of the hyperparameters distribution. To determine the α_{MP} we could integrate over the hyperparameters or use the evidence procedure [MacKay, 1992] which is an iterative method and is equivalent to type II maximum likelihood estimator.

5.3.3 Automatic Relevance Determination

Automatic Relevance Determination (ARD) is a method that uses Bayesian inference to identify the variables of the model which are more important than the others [MacKay, 1993]. A different hyper-parameter α is assigned to each variable. Since the hyper-parameter α is equal to the inverse of the variance, the smaller that α is, the larger will be the variance of the corresponding weight distribution. That is important because wider distributions means that the range of the weights for the specific variable is large. If the hyperparameter allows the weights to have a big number, that means that they are very important for the final result since they will dominate the results in contrast with the other variables that have smaller values. So we can compare the values of the hyper-parameters to decide which of them are important [Nabney, 2001]. Again there is no impartial approach to decide which is a big and a small hyperparameter. Especially in the case that each variable has different mean and variance we cannot compare the values of the hyperparameters since the differences will be the result of the different mean and variance. That is the reason why in ARD method the variables are normalized to zero mean and unit variance.

5.3.4 Results of classification using MLP with Bayesian Inference

The Bayesian MLP was trained only for leads 0 and 1 due to the time constraint of the MSc. There is no need to employ the early stopping technique since the Bayesian

	lead 0	lead 1
accuracy	81.48	89.10
sensitivity	83.81	82.38
specificity	80.75	91.19

Table 5.3: Results of test set for the Bayesian MLP for lead 0 and lead 1.

inference includes the regularization factor. For that reason the results of the error of the training set are used as a decision measure for the best MLP. A Bayesian MLP with 6 input, 8 hidden units and one output was trained for 1400 iterations was chosen for lead 0, and one with 6 input, 8 hidden units and one output was trained for 1000 iterations was chosen for lead 1. For the estimation of the hyperparameter α the evidence procedure was used. The results for the test set are illustrated in Table 5.3.

The best accuracy was achieved for lead 1. The overall results are better than all the other classifiers that had been trained so far, but are still 1.6% lower than the winning paper of the competition [Langley et al., 2003].

In lead 1 the specificity was 91.19% but the sensitivity was smaller than that of the lead 0. The reason the results are not as good as in lead 1 is the low percentage of specificity which is 10% smaller than lead 1. Comparing these with the previous classifiers the results in lead 1 were improved using the Bayesian inference in accuracy and specificity. In general, the results for lead 0 are not better than the ones achieved from the MLP that was trained using the principal components, ΔST and ΔT as inputs. The results for both lead 0 and lead 1 are still better than those of all the other entries of the competition apart from the winning paper.

5.3.5 Results of ARD

After the training of a Bayesian MLP with a dataset of zero mean and variance one for each variable the extracted hyperparameters α using the evidence procedure, for each variable are depicted in Table 5.4.

From the previous table we can see that the most significant variable for both leads is ΔT which has the smallest hyperparameter α. However we cannot specify with certainty which variable will be next significant one due to the fact that for both leads it has different value of significance. For example in lead 0 second most important

Variable	Value of hyper-parameter α lead 0	Value of hyper-parameter α lead 1
First Principal Component	0.215	0.252
Second Principal Component	0.235	0.634
Third Principal Component	0.186	0.101
Fourth Principal Component	0.252	0.091
ΔST	0.034	0.231
ΔT	0.019	0.015

Table 5.4: Results for the ARD method for both leads.

	Validation set	Test set
accuracy	79.91	85.81
sensitivity	66.34	78.31
specificity	83.92	87.57

Table 5.5: Results for the validation and test set of the MLP that used only ΔT.

variable is $PC4$ (0.0912) whereas in lead 1 it is not the same (ΔST 0.034).

5.3.6 Classification using only ΔT

For both leads the results show that the variable with the smallest hyperparameter α is ΔT. For that reason an MLP was trained only with that variable. That classifier selected using the early stopping technique for regularization had only one input, four hidden units, one output. The training stopped after 700 iterations. The results of that classifier are in Table 5.5

The results were very good for that simple MLP. In the test set the results were better than those of the validation set. In the validation set the sensitivity was only 66.34% but there was a big increase at the test set.

The accuracy of the test set was over 85% which is better than most of the entries of the competition. That means that the combination of the time an event lasts with the difference between the ST level and the baseline are of great importance for an automatic detector to classify correct ischaemic and non-ischaemic events.

5.4 Conclusion

Two different datasets were used for visualization. The results that obtained were similar with these of the previous chapter. Unfortunately overlapping occurred between ischaemic and non-ischaemic episodes. So we can assume that this phenomenon occurred because the data cannot be separated in two dimensions.

The new dataset improved the results of the classification. Then a Bayesian approach was implemented with better overall results. Finally one, of the most interesting results was the good ones obtained by an MLP which had as an input only the ΔT. Its accuracy was greater in the test set than all the entries of the challenge except for the winning paper. The last MLP was employed after the implementation of the ARD procedure. The results of ARD showed that the variable ΔT was the most important variable during the classification.

To sum up, we can observe that the Bayesian inference MLP can be used instead of the rule-based classifier of the winning paper since the results are very similar.

Chapter 6

Conclusions and Future Work

6.1 Thesis summary

This project was inspired by 2003 and 2005 PhysioNet and Computers in Cardiology challenges. The aim of this project is to develop an algorithm to distinguish ischaemic from non-ischaemic ST changes of an ECG.

A different approach from the other researchers, whose work was presented in the second chapter, was adopted for the extraction of the initial dataset. Instead of taking the ST segment from the beats of the whole episode, or a number of beats near to the beginning of the episode, the ST segment of the first beat only, for each episode that was chosen as a dataset. The experiments of visualization and classification were applied for the three combinations of leads that was provided from Physiobank.

Two techniques were used for feature extraction and visualization, PCA and Neuroscale. Additional features were used from the winning paper to find whether or not these features could improve the results of the MLP that had as input only the principal components. Many datasets were used for classification. Firstly an MLP with Principal Components as inputs was employed. The accuracy for that MLP was 76.52%, 78.33% and 88.89% for lead 0, lead 1 and lead 2 respectively on the validation set.

To improve the results ΔST and ΔT were added to the dataset. After the improvement of the results ARD method was employed to identify which of the input variables were more important. A Bayesian inference MLP also used as classifier. That MLP achieved the best accuracy from the classifiers that were trained. Finally an MLP with

input the most significant variable, ΔT, according to the results of ARD was implemented, with very good results since the accuracy of the test set was 85.81% which is greater than the most of the entries of the competition.

6.2 Conclusion

Summing up the results of the project we can conclude that:

- The visualization techniques that employed were not able to separate two classes of the data in two dimensions.
- Using the results only of PCA for feature extraction the results were compared with the other entries of the challenge.
- ARD results showed that ΔT is important in classifying ischaemia.
- The Bayesian inference MLP had the best results compared with the other classifiers which were trained.
- A classifier which is based in ΔT can classify more accurate the non-ischaemic episodes.

In the rest of that section we describe briefly these conclusions. The results of this project can be separated in two groups, the ones of visualization and the results of classification. Neither the plots of Principal Components, nor Neuroscale had useful results. In all the graphs that produced the two classes weren't separable. The overlap between ischaemic and non-ischaemic events was preserved even after using and the new dataset in .

Concerning the classification results of two first leads, we can observe that the first classifier using as an input the principal components had poor results compared to other work. Comparing them with the other entries they were better only than Povinelli's results, which were not better than chance. A notable point to these results was also the very small percentage of the ischaemic episodes that had been classified correctly for both leads. After adding the two new variables ΔST and ΔT into our dataset the results were improved spectacularly. The Bayesian inference MLP improved the results

in specificity and accuracy of lead 1. On the other hand there was a drop in specificity in lead 0.

Using the ARD method ΔT was the most important variable for both leads. After training a MLP using ΔT as the only input, the results were very good, the accuracy for the test set was 85.81%. According to these results we can conclude that the amplitude of the event combined with the ST deviation is a very important feature for the characterization of an event as ischaemic or non-ischaemic. Another remarkable point is the increase in the sensitivity when the principal components, ΔST and ΔT were used. The best sensitivity was for the validation set of lead 0, 84.91%. The dataset of lead 2 was different than the other two. The records that had three leads were containing more ischaemic episodes than non-ischaemic. After employing the first MLP using only Principal Components, the results were better than the other two leads. The accuracy and the sensitivity were bigger than these of lead 0 and lead 1 in validation set. Also at the test set the sensitivity was increased twofold from the other two leads. Contrary with accuracy and sensitivity, specificity was small, 60% for the validation set and 30% for the test set. After the usage of ΔST and ΔT this condition didn't change. In the particular experiment the specificity of the test set was 60% and for the validation set was 37.5%. From these results we derived to the conclusion that ΔST and ΔT are more accurate to identify the ischaemic episodes, but they can't contribute a lot to models that cannot identify non-ischaemic episodes.

6.3 Future work

For future work a data fusion model can be developed combing the data of the three leads. The ARD method could be used to identify which principal components are important for classification and then use them to the data fusion model. So the best features of the three leads will be used to train the classifier. The results can be compared with the results of each lead separately.

Bibliography

[Bamia, 2003] Bamia, C. (2003). *Biostatistics and Epidemiology.* Athens University of Economics and Business Publications.

[Bishop, 1995] Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford University Press.

[Boutkan, 1972] Boutkan, J. (1972). *ABC of the ECG.* Macmillan.

[Feldman et al., 2005] Feldman, C., Milstein, S., Neubecker, D., Underhill, B., Moyer, E., Glumm, S., Womble, M., Auer, J., Maynard, C., Serra, R., and Wagner, G. (2005). Comparison of the five electrode derived EASI Electrocardiogram to the Manson Likar Electrocardiogram in the Prehospital Setting. *Am J Cardiol*, 96:453–456.

[Jager et al., 2003] Jager, F., Taddei, A., Emdin, M., Moody, G., Antolic, G., Dorn, R., Smrdel, A., Marchesi, C., and Mark, R. (2003). Long-term ST database: a reference for the development and evaluation of automated ischaemia detectors and for study of the dynamics of myocardial ischaemia. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 41:172–182.

[Karlis, 2004] Karlis, D. (2004). *Multivariate Statistical Analysis.* Athens University of Economics and Business Publications.

[Langley et al., 2003] Langley, P., Bowers, E., Wild, J., Drinnan, M., Allen, J., Sims, A., Brown, N., and Murray, A. (2003). An algorithm to distinguish Ischaemic and Non-Ischaemic ST changes in the Holter ECG. *Computers in Cardiology*, 30:239–242.

[Lowe and Tipping, 1996] Lowe, D. and Tipping, E. M. (1996). Neuroscale: Novel Topographic Feature Extraction using RBF Networks. Advances in Neural Information Processing Systems 9.

[MacKay, 1992] MacKay, D. (1992). The Evidence Framework applied to Classification Networks. *Neural Computation*, 4:720–736.

[MacKay, 1993] MacKay, D. (1993). Bayesian Non-linear Modeling for the 1993 energy prediction competition. *Maximum Entropy and Bayesian Methods.*

[Nabney, 2001] Nabney, I. T. (2001). *Netlab algorithms for Pattern Recognition.* Springer.

[Papaloukas et al., 2001] Papaloukas, C., Fotiadis, D., Liavas, A., Likas, A., and Michalis, L. (2001). A knowledge-base technique for automated detection of ischemic episodes in long duration electrocardiograms. *Med Biol Eng Comput*, 39:105–112.

[Papaloukas et al., 2002] Papaloukas, C., Fotiadis, D., Likas, A., and Michalis, L. (2002). An Ischemia detection method based on Artificial Neural Networks. *Artificial Intelligence in Medicine*, 24:167–178.

[Povinelli, 2005] Povinelli, R. J. (2005). Towards the Prediction of Transient ST Changes. *Computers in Cardiology*, 32:663–666.

[Stamkopoulos et al., 1998] Stamkopoulos, T., Diamantaras, K., Maglaveras, N., and Strintzis, M. (1998). ECG Analysis Using Nonlinear PCA Neural Networks for Ischemia Detection. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 46:3058–3066.

[Zimmerman and Povinelli, 2004] Zimmerman, M. and Povinelli, R. (2004). On Improving the Classification of Myocardial Ischemia Using Holter ECG Data. *Computers in Cardiology*, 31:377–380.

[Zimmerman et al., 2003] Zimmerman, M., Povinelli, R., Johnshon, M., and Ropella, K. (2003). A Reconstructed Phase Space Approach for Distinguishing Ischemic from Non-Ischemic ST Changes using Holter ECG Data. *Computers in Cardiology*, 30:243–246.

Appendix A

Figures of eigenvalues for lead 0 and lead 2

Here are represented the graphs of the eigenvalues for lead 0 and lead 2. These figures, as the figure 4.1, are not informative about the number of Principal Components that should be used. Figure A.1 depicts the plot of all the eigenvalues and the first twenty eigenvalues for lead 0 respectively. Figure A.2 depicts the plot of all the eigenvalues and the first twenty eigenvalues for lead 2 respectively.

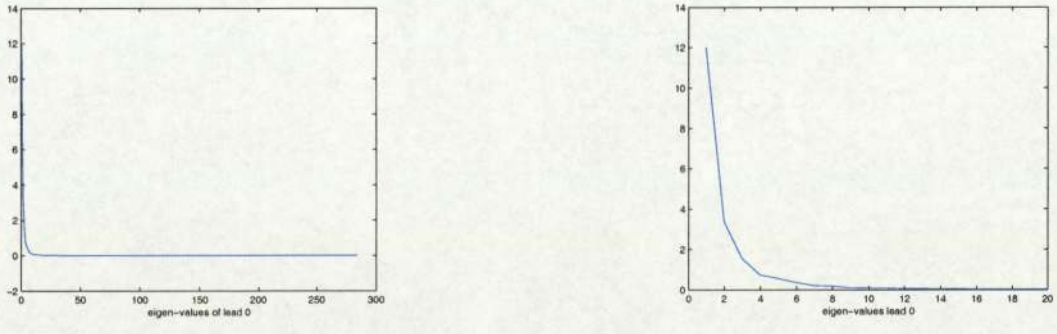


Figure A.1: Plot of eigenvalues for lead 0.

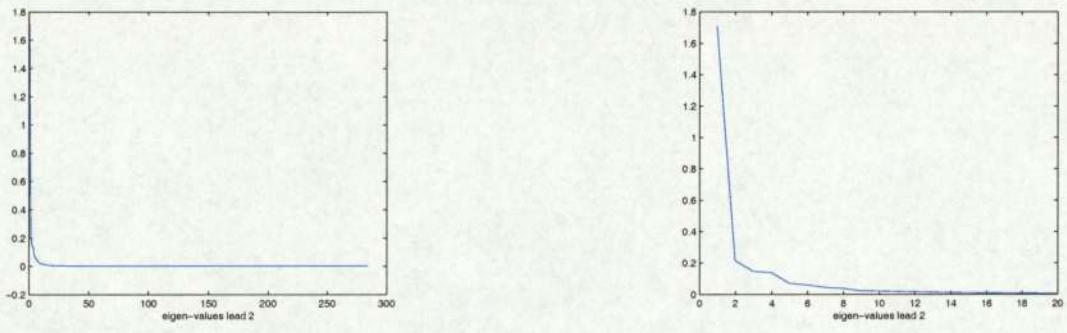


Figure A.2: Plot of eigenvalues for lead 2.

Appendix B

Visualization for all the Principal Components of lead 0 which have been extracted

This section represents the figures that depict the results of visualization, for different combinations of the plots of principal components for lead 0. The two classes are not separated in these figures.

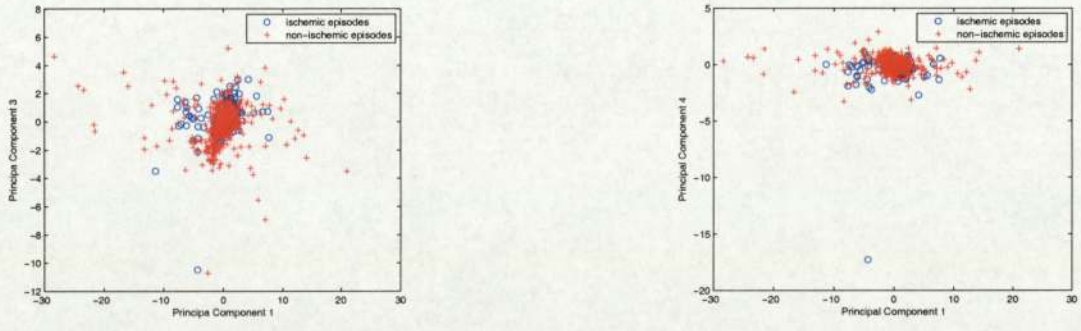


Figure B.1: Plots of Principal Component 1 versus the other extracted Principal Components for lead 0.

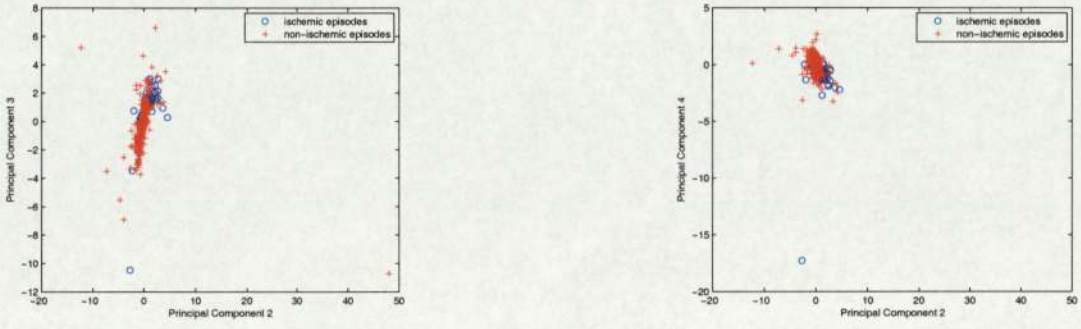


Figure B.2: Plots of Principal Component 2 versus the other extracted Principal Components for lead 0.

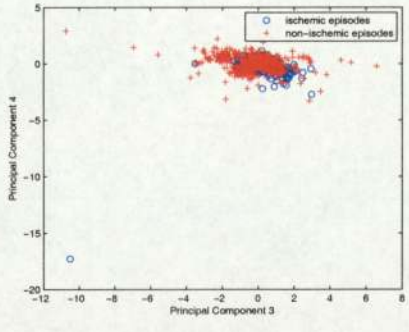


Figure B.3: Plot of Principal Component 3 versus the fourth Principal Component for lead 0.

Appendix C

Visualization for all the Principal Components of lead 1 which have been extracted

This section represents the figures that depict the results of visualization, for different combinations of the plots of principal components for lead 1. The results are the same as in lead 0. There is an overlap between ischaemic and non-ischaemic episodes.

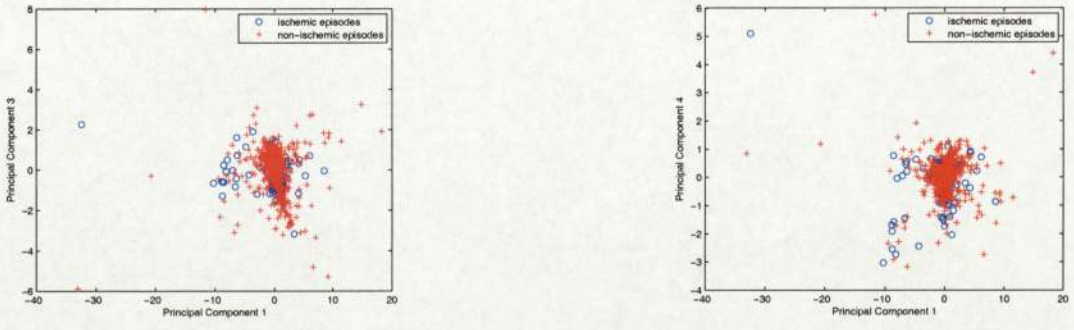


Figure C.1: Plots of Principal Component 1 versus the other extracted Principal Components for lead 1.

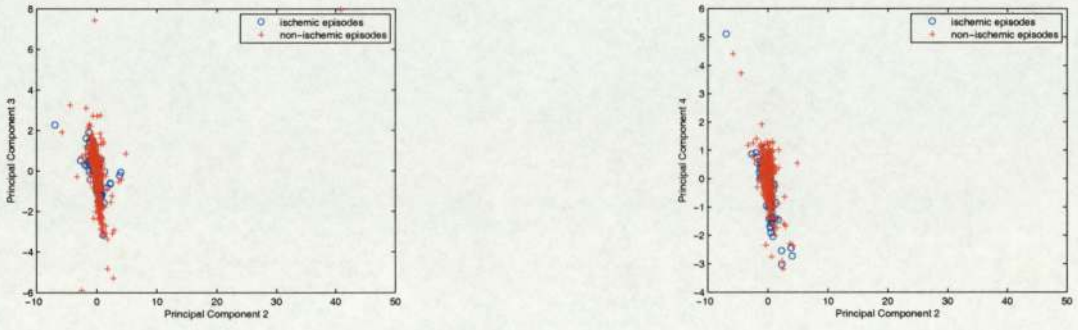


Figure C.2: Plots of Principal Component 2 versus the other extracted Principal Components for lead 1.

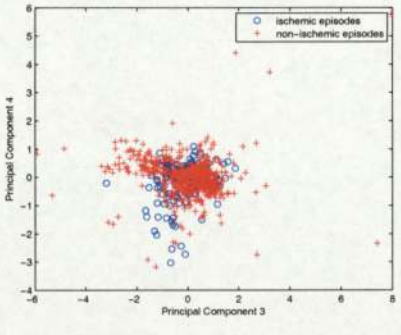


Figure C.3: Plot of Principal Component 3 versus the fourth Principal Component for lead 1.

Appendix D

Visualization for all the Principal Components of lead 2 which have been extracted

This section represents the figures that depict the results of visualization, for different combinations of the plots of principal components for lead 2. Again the results are the same as the two previous leads. The ischaemic events cannot be separated from non-ischaemic events.

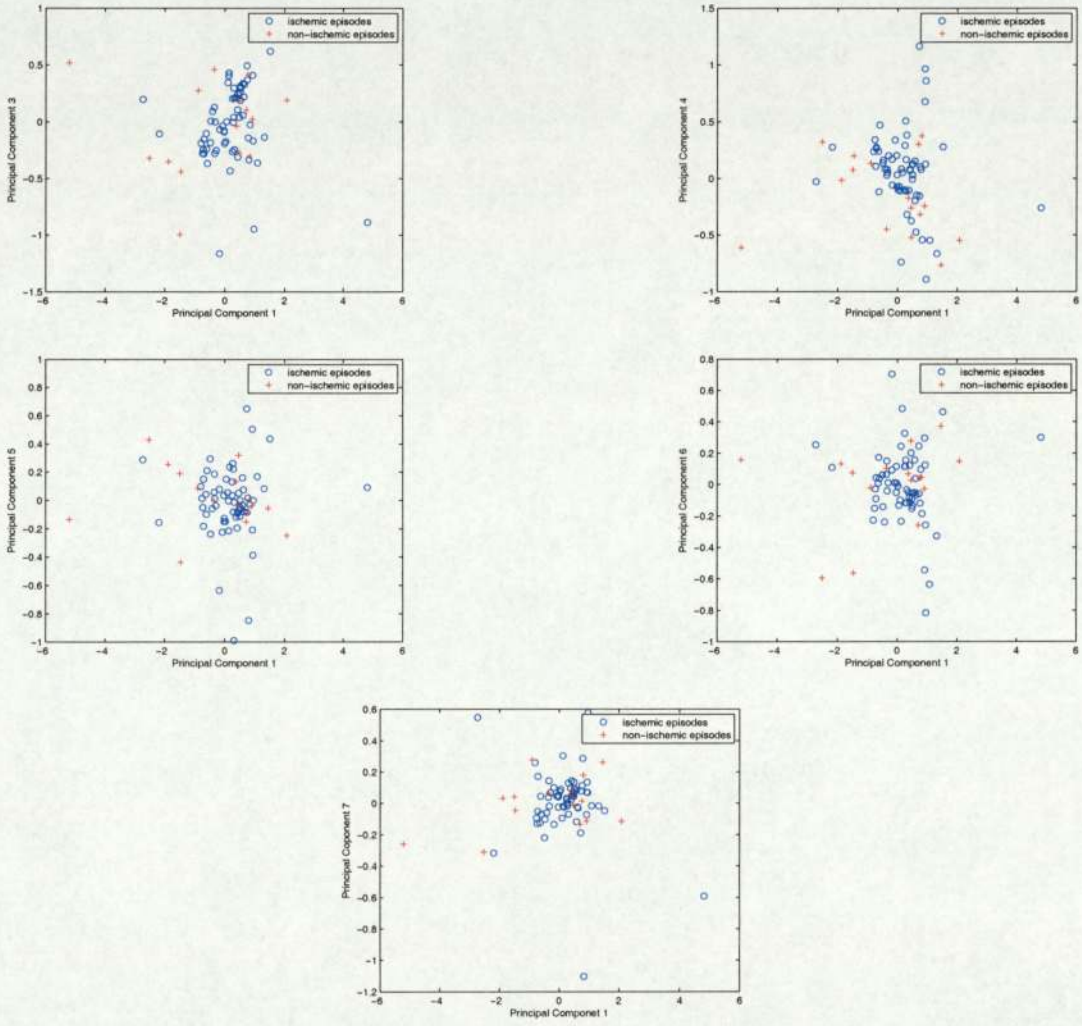


Figure D.1: Plots of Principal Component 1 versus the other extracted Principal Components for lead 2.

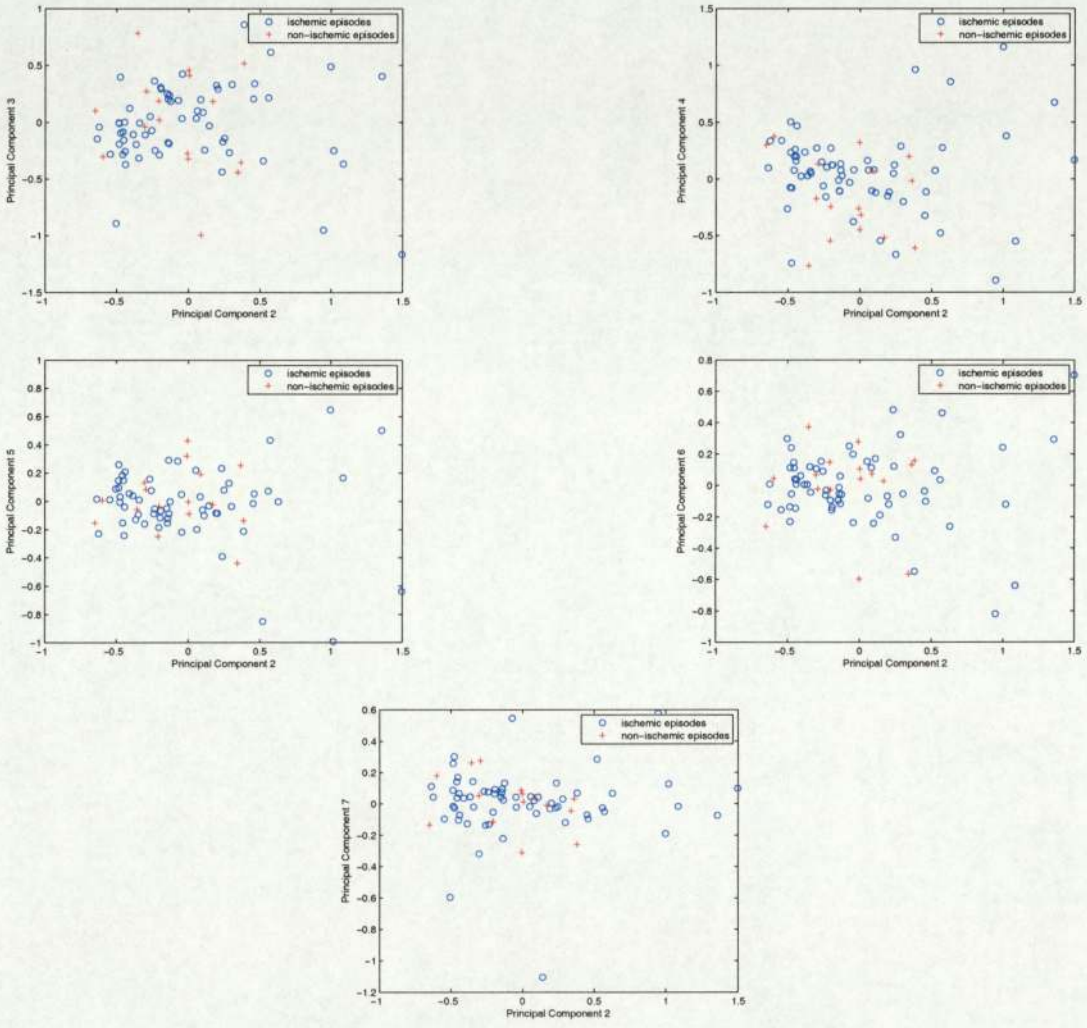


Figure D.2: Plots of Principal Component 2 versus the other extracted Principal Components for lead 2.

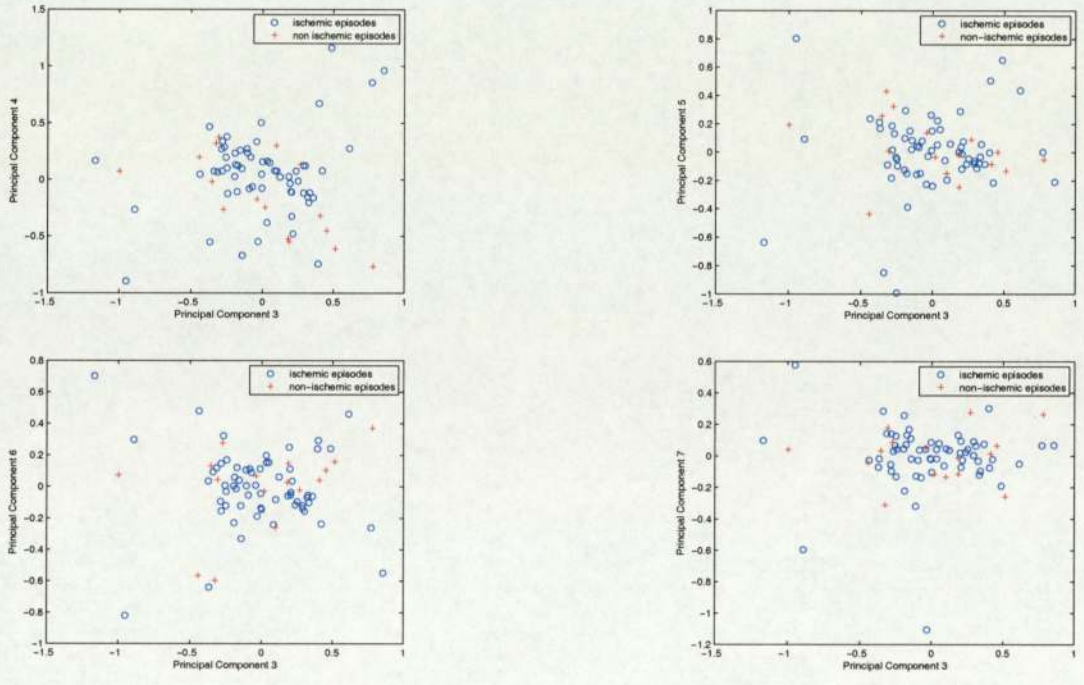


Figure D.3: Plots of Principal Component 3 versus the other extracted Principal Components for lead 2.

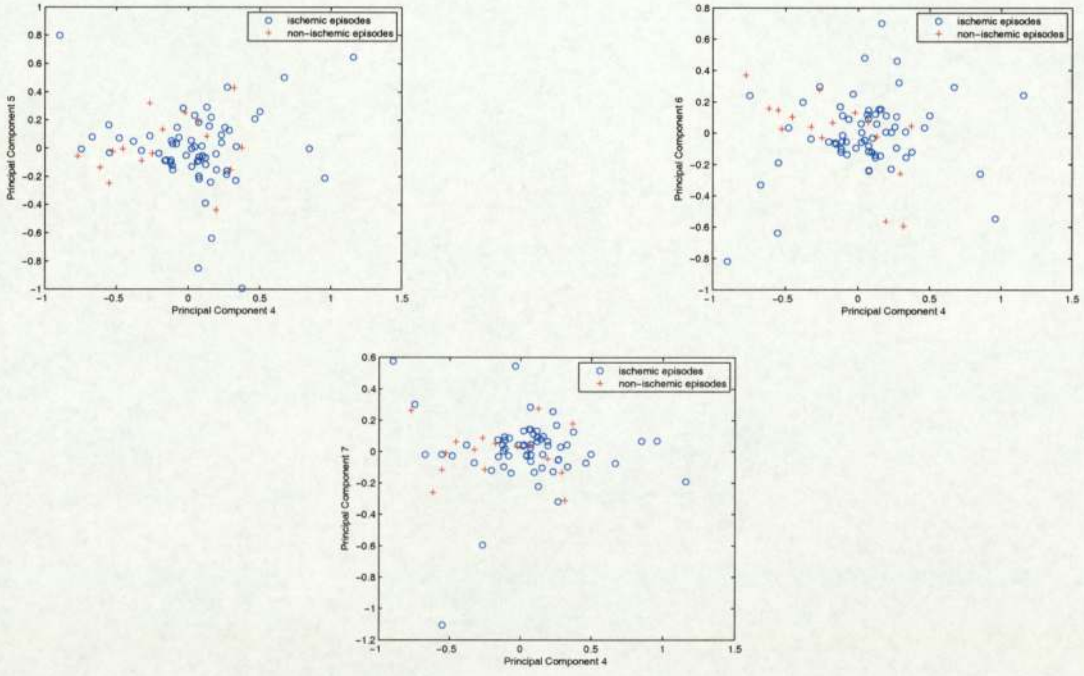


Figure D.4: Plots of Principal Component 4 versus the other extracted Principal Components for lead 2.

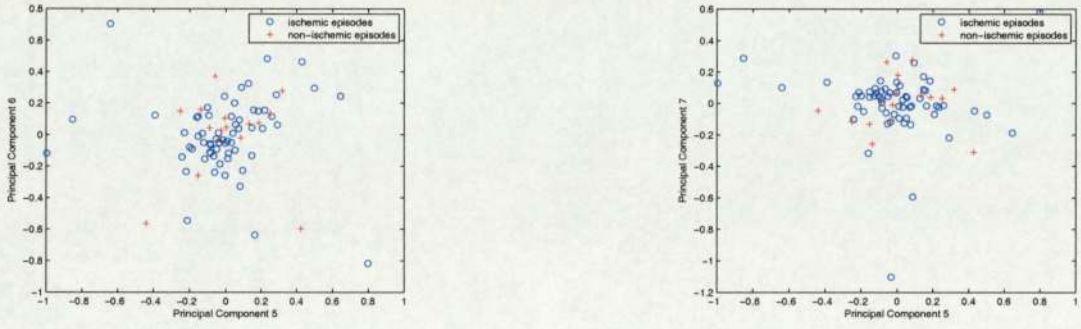


Figure D.5: Plots of Principal Component 5 versus the other extracted Principal Components for lead 2.

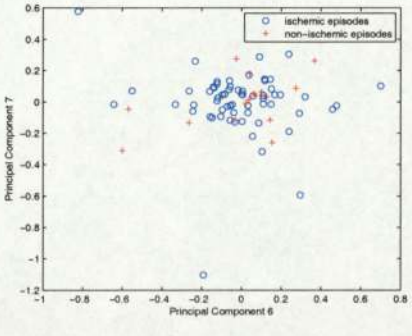


Figure D.6: Plots of Principal Component 6 versus the seventh Principal Components for lead 2.

Appendix E

Neuroscale results for lead 0

This appendix contains the results of for lead 0. The results are depicted in graph E.1 for the three different datasets that were used, one with only the Principal Components, one with the Principal Components and the ΔST,and finally the previous added the variable ΔT, respectively. As the graphs depict, we cannot separate the two classes in lead 0 whatever dataset we use.

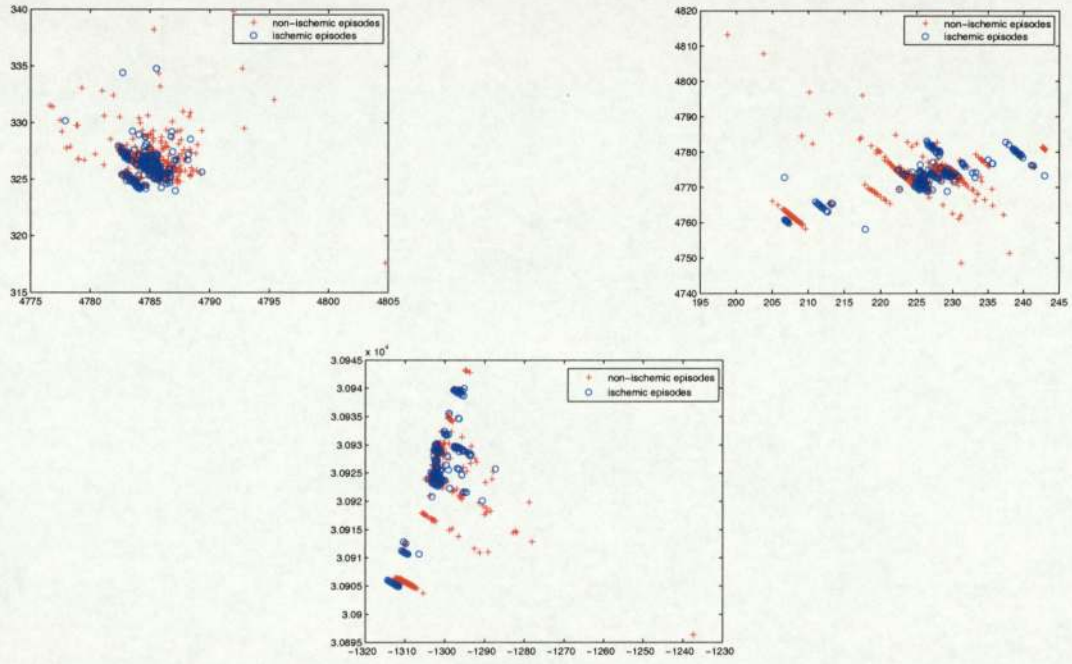


Figure E.1: Neuroscale results for the different datasets for lead 0.

Appendix F

Neuroscale results for lead 2

The results of for lead 2 are presented in this appendix. The results are depicted in graph F.1 for the three different dataset that were used, one with only the Principal Components, one with the Principal Components and the ΔST,and finally the previous added the variable ΔT, respectively. As it depicts in the graphs we cannot separate the two classes in lead 2 whichever dataset we use.

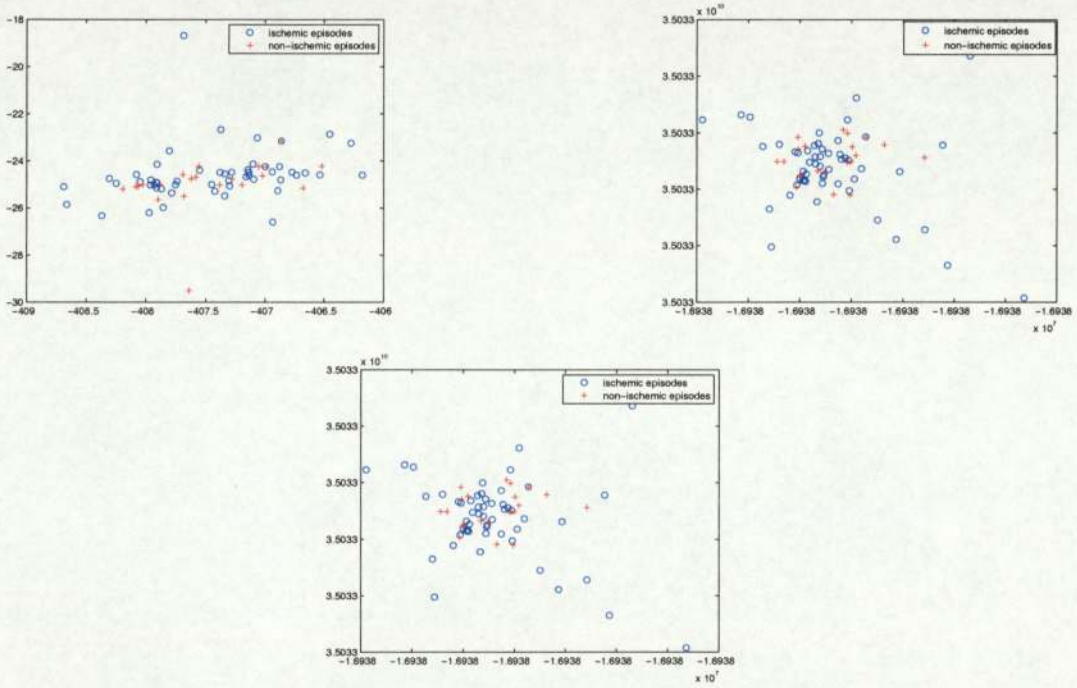


Figure F.1: Neuroscale results for the different datasets for lead 2.