Benchmarking Beat Classification Algorithms

IT Nabney[1], DJ Evans[1], J Tenner[2], L Gamlyn[2]

[1]Cardionetics Institute of Bioinformatics, Aston University, UK
[2]Cardionetics Ltd., UK

Abstract

The aim of this study is to compare the accuracy of a range of advanced and classical pattern recognition algorithms for beat and arrhythmia classification from ECG using a principled statistical framework. These are to be used in an application where no patient-specific adaptation of features or model is possible, which means that models must be able to generalise across subjects. Our results demonstrate that non-linear classification models offer significant advantages in ECG beat classification and that with a principled approach to feature selection, pre-processing, and model development, it is possible to get robust inter-subject generalisation even on ambulatory data.

1. Introduction

Our goal is to evaluate the suitability of a range of pattern analysis algorithms for use in a device worn for 24 hours by the patient (such as the C.Net 2000 developed by Cardionetics Ltd. [1]). Such a device must be capable of analysing each heart beat (that is about 100,000 beats in a 24 hour test) for a large number of different patients without adaptation of the model parameters. In addition, since the device is worn during normal activities, it must be light and use small batteries [2]. This means that the computational demands of both feature extraction and classification must be scrutinised to minimise CPU usage, and hence power consumption.

2. Data modelling

In this section we describe the comparative experiments that were used to decide which algorithms were most suitable for the problem of classifying ventricular ectopic beats.

2.1. Description of data

Single channel ECG was collected from 131 subjects using a digital data collection device at 100 Hz with the same frequency response characteristics as a C.Net 2000 ambulatory monitor. This was divided on a per-subject basis into training (35403 beats), validation (11800) and test (23606) sets so that each dataset contained beats from separate subjects (i.e. there was no data for a given subject that was included in more than one dataset). This ensures that the generalisation results achieved are realistic measures of the expected performance of the models in a clinical environment.

The three classes of beat, supraventricular (including sinus) and two varieties of ventricular ectopic (VE) beats, had approximate prior probabilities C_i0.85, 0.05 and 0.10 respectively. 24 time-domain features were extracted automatically from the data and normalised on a per-beat basis to reduce their subject specificity. These were chosen in consultation with experienced clinical cardiologists.

2.2. Data visualisation

A principled approach to pattern recognition requires a good understanding of the data. Our exploratory analysis consisted of feature histograms and correlation plots. The histograms identified outlying values, quantisation effects, and multi-modality suggesting clustering within the VE classes. The plots showed that most features were nearly uncorrelated, but that there was some scope for feature reduction (see Section 2.5).

For visualisation, we used Principal Component Analysis (PCA), a classical linear technique, and compared it with Neuroscale [3], a neural-network based enhancement of the Sammon mapping which can generalise to unseen data, allowing it to be trained on a smaller sub-sample, and thus speeding up the process of generating a new plot. Normalisation of the data to zero mean and unit variance considerably improved the results of visualisation, and was used in all subsequent analysis and modelling. Visualisation plots showed that there was good class separation, particularly between

0276-6547/01 $17.00 © 2001 IEEE

Computers in Cardiology 2001;28:529-532.

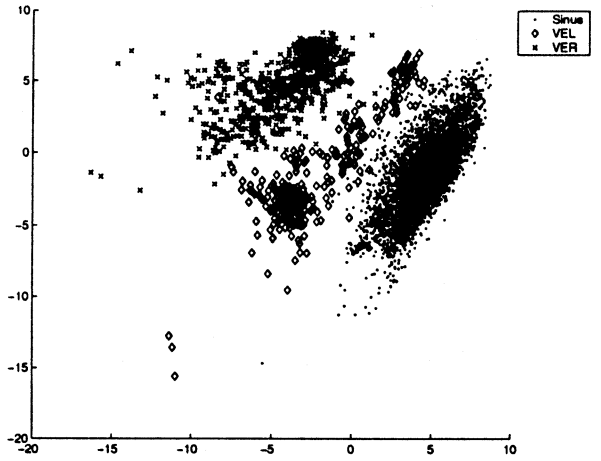

Figure 1. Neuroscale visualisation applied to normalised ECG data.

supraventricular and the VE classes. The VE sub-class structure was also clearly visible. Neuroscale plots (see Figure 1) showed better separation, suggesting that a non-linear classification model might give better results than a linear model.

2.3. Description of models

The problem was represented as a three-way classification since earlier experiments showed that this gives more accurate results than multiple two-way classification models. Early stopping was adequate to avoid overfitting, owing to the large quantity of training data. The classification models used were logistic regression (or Generalised Linear Model, GLM), MLP, and RBF neural networks [4]. The density models used were Gaussian Mixture Models (GMMs), Generative Topographic Mapping (GTM) [5], and Kohonen SOM [6], with one model per class. All model selection (e.g. number of hidden units) was performed using only the validation dataset. All the selected models were finally evaluated on the test set. Each model was trained five times with randomly initialised weights; in the case of the SOM and GTM this was a random perturbation of the weights following a PCA initialisation. The NETLAB toolbox[1] was used for all the experiments.

The classifiers were trained to model the Bayesian posterior distribution of the probability of a data point $\mathbf{x}$ belonging to class $\mathcal{C}_k$, formally written as $P(\mathcal{C}_k|\mathbf{x})$, the probability of $\mathcal{C}_k$ *given* an input vector, $\mathbf{x}$. The GLM, MLP and RBF models used a softmax output activation function to ensure that the outputs lay in the range $[0, 1]$ and summed to one.

For the density models we compute $P(\mathcal{C}_k|\mathbf{x})$ by modelling the class conditional density $P(\mathbf{x}|\mathcal{C}_k)$ (i.e. we train a model for each class) and then applying Bayes' theorem to compute the posterior distribution, $P(\mathcal{C}_k|\mathbf{x})$:

$$P(\mathcal{C}_k|\mathbf{x}) = \frac{P(\mathbf{x}|\mathcal{C}_k)P(\mathcal{C}_k)}{\sum_{i=1}^{K} P(\mathbf{x}|\mathcal{C}_i)P(\mathcal{C}_i)}. \quad (1)$$

2.4. Results

Tables 1 and 2 contain the confusion matrices for the trained models. Confusion between the two ventricular classes is of no clinical importance. Misclassifying sinus beats as ventricular ectopics may lead to over-reporting of symptoms, but misclassifications of ventricular beats as sinus (recorded in the first column below the first entry) are the most costly errors. These tables show that the most accurate model is the MLP, while the most accurate density model is the GTM.

GLM: Mean err: 0.1593%; std err: 0.0110%		
19984 (19982/19985)	3 (2/5)	0 (0/0)
4 (3/4)	1159 (1159/1160)	25 (25/25)
0 (0/0)	5 (4/8)	2426 (2423/2427)
MLP: Mean err: 0.0153%; std err: 0.0038%		
19985 (19984/19985)	2 (2/3)	0 (0/0)
0 (0/0)	1187 (1187/1188)	1 (0/1)
0 (0/0)	0 (0/1)	2431 (2430/2431)
RBF Mean err: 0.1288%; std err: 0.0057%		
19981 (19981/19982)	5 (5/6)	0 (0/1)
1 (0/1)	1175 (1174/1176)	12 (12/13)
0 (0/0)	11 (11/13)	2420 (2418/2420)

Table 1. Performance statistics for the classifiers. Results are quoted as: median (min/max)

SOM Mean err: 0.6219%; std err: 0.0746%		
19931 (19903/19945)	55 (40/84)	0 (0/2)
17 (4/48)	1132 (1096/1156)	28 (28/44)
0 (0/0)	42 (7/52)	2389 (2379/2424)
GTM 8x8 Mean err: 0.1432%; std err: 0.0436%		
19971 (19970/19975)	15 (11/16)	1 (1/1)
4 (2/6)	1174 (1160/1184)	11 (0/22)
1 (1/1)	4 (2/7)	2426 (2423/2428)
GMM Mean err: 0.1805%; std err: 0.0311%		
19966 (19954/19971)	21 (16/33)	0 (0/1)
1 (0/4)	1179 (1164/1180)	8 (7/20)
1 (1/1)	4 (2/9)	2426 (2421/2428)

Table 2. Density models

One of the two beats misclassified by the best MLP is shown in Figure 2; it can be seen that although it is a sinus beat (since there is a P-wave present and the timing is regular), the QRS complex is unusually broad and the T-wave (which is inverted) is relatively small; this morphology, which has similarities to a VE, is probably

[1] http://www.ncrg.aston.ac.uk/netlab

that of a *fusion beat*, which is caused by both sinus and ventricular initiation.

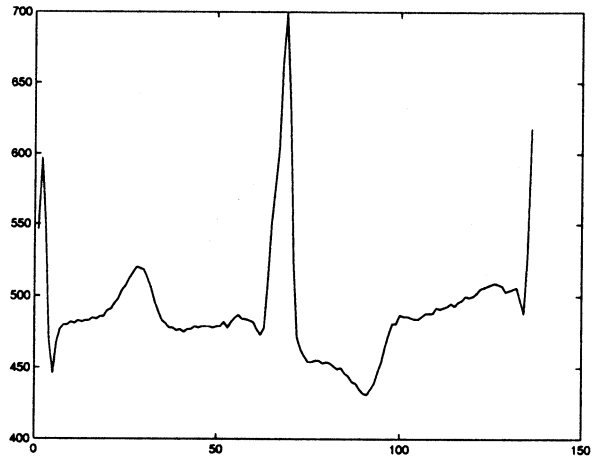

Figure 2. Sinus beat misclassified by the MLP.

2.5. Feature selection

Based on our exploratory analysis, we decided to investigate the effect of reducing the number of features on the accuracy of the models. Automatic Relevance Determination (ARD) is a Bayesian technique for feature selection [7]; it puts a zero mean Gaussian prior distribution on the parameters of a model with the special property that there is a separate hyperparameter (representing the inverse variance) for the weights fanning out from each input. During training, these hyperparameters are optimised using the evidence procedure [8], and their magnitude measures how important the corresponding input is for the model's output. Because hyperparameters represent the inverse variance of the weights, a small hyperparameter value means that large weights are allowed, and we can conclude that the corresponding input is important. A large hyperparameter value means that the weights are constrained near zero, and hence the corresponding input is less important.

After optimisation, the hyperparameters converged into 4 distinct groups consisting (from most to least important) of 10, 3, 4, and 7 variables (see Figure 3). We then trained MLPs on 10, 13, 17, and the full set of 24 inputs. The number of mis-classified beats on the independent test set were 9 (error rate 0.0381%), 14 (0.0593%), 9 (0.0381%) and 2 (0.0085%). The first two models classified just two VE beats as sinus, while the latter two made no such mis-classifications.

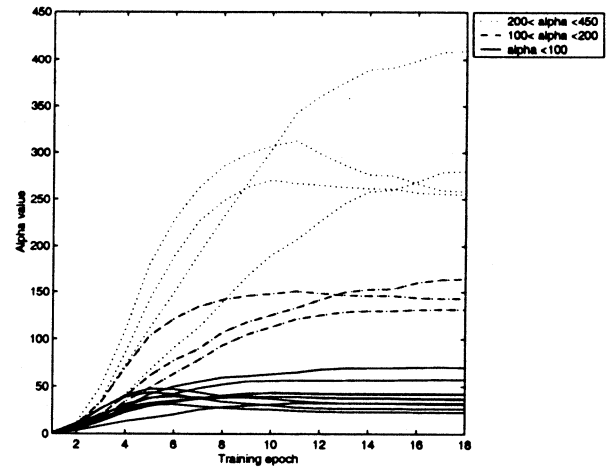

Figure 3. Significant hyperparameter values after training with ARD

3. Computational efficiency

The current C.Net device is powered by 4 AA batteries which are replaced after each test. Minimising power consumption is an important design goal, and for this reason the software does not use floating point operations. Instead most computations are carried out in integer arithmetic, normally in 32 bit precision, but also 16 bit or 8 bit where this is adequate. Where integers are not sufficient, fixed point arithmetic is used; normally 32 bit precision, again sometimes 16 bit or even 8 bit. We distinguished four classes of operation in order of increasing computational demand:

1. Light operations l such as assignments and comparisons.
2. Addition and subtraction a.
3. Multiplication m.
4. Division d. Division by a fixed quantity is treated as a multiplication, since the factor can be precomputed.

Table 3. Operation count per beat of feature extraction.

Analysis	l	a	m	d
Beat detection	202	43	1	0
Width extraction (worst)	815	1	3	0
Width extraction (typical)	215	1	3	0
Feature computation (sinus class)	6	8	23	0
Feature computation (3 classes)	18	24	69	0

There are two major phases of classification: feature extraction (Table 3) and running a classifier data model (Table 4).

All the models, apart from the SOM and the GLM used as a discriminant (i.e. without softmax outputs), require the calculation of non-linear activation functions.

Without the use of floating point library routines, special-purpose software is required. The method of choice is argument reduction to a limited range followed by a rational polynomial approximation; for 8 bit precision, a ratio of two linear polynomials seems to be adequate [9].

There are two possible modes of operation for the SOM. In the full mode used for all the results above, every node in all three models is computed and the beat is assigned to the winning node. In (a simplified version of) the reduced mode used in the C.Net, at first only the distance from the winning node from the previous beat is calculated. If this is closer than a threshold, then the beat is assigned to that class. Otherwise, the remaining nodes from the same network are computed (in fact, there may be intermediate groups of nodes, but we will just consider a simplified case here). Again, if one of these is close enough, then the beat is assigned to that class. Finally all the nodes in the other two networks are computed. The computational cost of this mode of operation will depend on the dataset and the distribution of different classes. A similar method can be used for the GMM and GTM. Further saving of computational effort in the SOM can be achieved by replacing the usual L_2 norm (i.e. sum-squared distance) by the L_1 norm $\sum_i |x_i - c_i|$.

Table 4. Typical operation count per beat for 24-input classifiers. H is the number of hidden units or centres.

Model	l	a	m	d
GLM (discriminant)	3	72	75	0
GLM (class probs.)	18	95	99	6
MLP ($H = 40$, discriminant)	203	1360	1403	40
MLP ($H = 40$, class probs.)	218	1383	1427	46
RBF ($H = 80$, Gaussian)	403	4087	2883	80
SOM ($H = 50$, full L_2)	150	7200	3600	0
SOM ($H = 50$, full L_1)	300	7200	0	0
SOM ($H = 50$, reduced L_2)	31	1469	734	0
SOM ($H = 50$, reduced L_1)	62	1469	0	0
GMM[a] ($H = 37$, full)	555	6216	3774	111
GMM[b] ($H = 85$, reduced)	258	2890	1754	52
GTM ($H = 64$, full)	960	10752	6528	192

[a] Average of 90, 12, and 8 centres.

[b] Approximate weighted average of 90, 12, and 8 centres.

4. Conclusions

In this study classification models gave significantly better performance than the density models. The best performance was achieved by an MLP with 2 mis-classified beats (error rate 0.0085%) with a mean error rate of 0.0153%, while the best from a classification model was a GTM with 19 mis-classified beats (0.0805%) with mean error rate 0.1432%.

Experiments with ARD showed that Bayesian methods can usefully be applied to non-linear classification models even in data-rich problems. At the cost of an additional 0.0551% in error rate, the number of features was reduced by 60%, a significant saving in computational effort for on-line classification in an ambulatory device. Another important issue for on-line use is the computational effort to classify a beat, which is least for the SOM and logistic regression.

These results demonstrate that non-linear classification models, such as neural networks, offer significant advantages over classical approaches in ECG beat classification, and that with a principled approach to feature selection, pre-processing, and model development, it is possible to get robust inter-subject generalisation even on ambulatory data.

References

[1] Gamlyn L, Needham P, Sopher SM, Harris TJ. The development of a neural network-based ambulatory ECG montior. Neural Computing and Applications 1999;8:273–278.

[2] Standing P, Dent M, Craig A, Glenville B. Changes in referral patterns to cardiac out-patient clinics with ambulatory ECG monitoring in general practice. Brit J Cardiol 2001;6:394–398.

[3] Lowe D, Tipping ME. Feed-forward neural networks and topographic mappings for exploratory data analysis. Neural Computing and Applications 1996;4:83–95.

[4] Bishop CM. Neural Networks for Pattern Recognition. Oxford University Press, 1995.

[5] Bishop CM, Svensén M, Williams CKI. GTM: The Generative Topographic Mapping. Neural Computation 1996;10(1):215–235.

[6] Kohonen T. Self-Organizing Maps. Berlin: Springer-Verlag, 1995.

[7] MacKay DJC. Bayesian Methods for Adaptive Models. Ph.D. thesis, California Institute of Technology, 1992.

[8] MacKay DJC. A practical Bayesian framework for back-propagation networks. Neural Computation 1992;4(3):448–472.

[9] Cody WJ, Waite W. Software Manual for the Elementary Functions. New Jersey: Prentice-Hall, 1980.

Address for correspondence:

Ian T. Nabney
Cardionetics Institute of Bioinformatics
Aston University
Birmingham B4 7ET
United Kingdom
i.t.nabney@aston.ac.uk