Optimisation of a Multi-parameter monitor for Early Warning of Patient Deterioration

David Wong

Brasenose College

Supervised by

Professor Lionel Tarassenko

Submitted: Michaelmas 2008

This PRS Transfer is Submitted to the Department of Engineering Science

Summary

This report presents preliminary research describing the construction of a multi-parameter patient vital-sign monitoring system. We demonstrate optimisation of an existing system and also describe alternative methods of creating the probabilistic model of normality used to identify deterioration in patient condition. A number of results are presented that motivate the research proposed for the remainder of this D.Phil. The aim is to substantially improve upon work in the identification of patient deterioration using multi-parameter monitoring, focussing on the inclusion of recently-acquired data from clinical trials, incorporating temporal information to perform dynamical analysis, and creating systems personalised to individual patients.

A large number of adverse clinical events can be detected prior to their onset, and it is widely recognised that early detection and intervention of such events is likely to lead to better patient outcome. Vital signs have been shown to be good indicators of patient health, including heart rate, breathing rate, blood pressure, temperature and arterial oxygen saturation. However, on acute general hospital wards, these vital signs are typically observed once every four hours, during which a patient's condition can worsen considerably.

A model of normality constructed using four vital sign parameters (the heart rate, breathing rate, blood pressure, and arterial-oxygen saturation) has been previously developed which has been used to continuously identify abnormal vital signs that may be predictors of an adverse clinical event. The model was trained using "normal" patient data, such that vital signs differing considerably with respect to the training set are deemed abnormal. This technique has an advantage over "expert systems"; prior knowledge about clinically adverse events is not required, as the model is entirely data driven and thus simpler to train.

The set of training data was acquired from a representative group of patients. Machine learning techniques were used to estimate the probability density function (p.d.f.) of the four-dimensional vital sign data. The unconditional probability is calculated using the trained p.d.f., where a low probability indicates an abnormal recording. If the probability is below a critical threshold, an alert call can be triggered. The model has been validated on data from the University of Pittsburgh Medical Centre (UPMC).

In this report, attempts have been made to improve the data fusion model. Initially, the parameters in the current data fusion technique were adjusted to minimise a global error score.

After this, an alternative data fusion method, the Weighted Parzen Windows Algorithm, was tested. The new method was then qualitatively assessed by applying it on real patient data.

During this preliminary research, a number of tests required visualisation of high-dimensional data in order to gain an intuitive understanding of the techniques used. Limitations in current visualisation methods were discovered, and a novel visualisation algorithm is presented and applied to clinical data. The algorithm is a modification on Sammon Mapping [47], but extends the method so that it is applicable to very large datasets containing $> 10^4$ elements.

Work in the immediate future will focus on developing the data fusion model in three main areas. Firstly, the breadth of vital signs monitored by the model will be increased, including a score of patient consciousness based on the Glasgow Coma Scale (GCS). Initial investigations will be made to assess whether the GCS can be estimated continuously, using Electroencephalogram (EEG) recordings. Secondly, a dynamic model of normality, such that trends in vital sign data are incorporated. Finally, a patient specific model will be developed, which aims to improve the specificity of the model of normality by retraining it on individual patient vital signs as patients are first admitted onto the ward.

Contents

1 Introduction **5**

2 Literature Review **8**

2.1 Medical Emergency Teams and Current Systems 8

2.2 Multiple Channel Alarms 12

3 Creating a Model of Normality **16**

3.1 Training Data 16

3.2 Pre-Processing the Data 17

3.3 Clustering 21

3.4 The model of normality - Parzen Windows 22

3.5 Visensia Status Index (VSI) 23

4 High-Dimensional Data Visualisation **26**

4.1 Sammon Maps and Neuroscale 26

4.2 SASS - a novel visualisation tool 29

4.3 SASS generalisation - interpreting new data 33

4.4 Results 36

5 Optimising the Parzen Windows Model **40**

5.1 Optimising the Parzen Width 40

5.2 Optimising the number of Parzen Windows Kernels 44

5.3 Assessment of the Model of Normality 45

5.4 Weighted Parzen Windows 47

6 Conclusion and Future Work **52**

6.1 Patient-specific models of normality 52

6.2 Investigating the relationship between EEG and GCS 53

6.3 Timeline for Completion of Research 55

A Glossary **57**

B Conferences Attended **57**

1 Introduction

As medical practice improves, an increasing number of acutely ill patients are being admitted for major surgical procedures where previously the risk of complications was considered too high. Advances in technique have also enhanced medical procedures so that for many patients, the procedures are carried out during day surgery, removing the need for overnight stays [16]. Furthermore, the number of patients requiring emergency medical admission has increased significantly over the last two decades, and a rise of 50% was reported between 1984 and 1997 [8]. The combination of these factors has meant that current patients admitted into acute wards are typically "older, undergoing major surgical procedures, or are acutely ill" [12].

Because the proportion of seriously ill inpatients has increased, the number of patients who are at greater risk of catastrophic deterioration while on a general ward has also increased. Important indicators of well-being are a patient's vital signs. These are physiological parameters often recorded by healthcare professionals in order to assess the functioning of vital organs such as the heart or the lungs. The vital signs that are most commonly measured in clinical practice are the heart rate, breathing rate, temperature, blood pressure, and arterial oxygen saturation. Numerous studies have shown that abnormalities in vital signs often precede adverse clinical events, such as a cardiac arrest (that is, the life-threatening onset of a chaotic and unproductive heart rhythm caused by an abnormal heart rhythm), unanticipated Intensive Care Unit (ICU) admission, or unexpected death.

One such study reports that respiratory and mental instability are often early precursors of a cardiac arrest [48], while another shows that fifty percent of patients who were later admitted to ICU showed vital sign abnormalities for at least three days prior to their admission [36]. Kause et al. [28] conducted an international prospective study at 90 hospitals in the UK, Australia and New Zealand, which showed that 60% of the patients who suffered an adverse event had documented antecedents. Various other articles [53, 9] also report that in many cases, deterioration in a patient's condition can be readily observed prior to other adverse events requiring ICU admission.

If such deterioration in patient well-being can be detected quickly, appropriate prophylactic measures can be taken to avert further deterioration. Early clinical intervention should reduce the number of preventable adverse events and the number of unexpected ICU admissions. In

the UK alone, it is estimated that at least 20,000 unanticipated ICU admissions per year may be avoided, and that 23,000 cardiac arrests per year may also be preventable [24, 36]. Furthermore, patients still requiring admission to ICU could be transferred earlier from general wards - one study links an efficient transfer from general wards to ICU with a reduced mortality rate [65]. This makes sense, as one expects that patients who are quickly transferred to ICU will be less acutely admission, and would therefore expect a better outcome or a shorter length of stay on ICU. As each ICU bed costs £1000-£1800 per day (in 1999) [6], early intervention also has clear economic benefits.

However, in general wards, vital signs are routinely observed by nurses approximately every four hours [51], or even less frequently, and a nurse may be expected to look after up to ten patients. Both of these factors contribute to the problem of significant patient deterioration often going unnoticed. This is supported by a recent study conducted on 326 patients at a 24-bed Step Down Unit (SDU) at a University Medical Centre in the U.S., where clinically significant vital sign deterioration occurred in 59 patients, of which only 7 were detected by clinical staff [26]. A step-down unit is a transitional care area with a greater degree of monitoring than on a general ward, but with less monitoring than on an ICU.

In many instances the vital sign deterioration may not be obvious, especially if it manifests itself as small changes in several vital signs, rather than a major change in a single vital sign. Furthermore, even when clear deterioration can be seen, studies indicate that in up to 25% of cases, the optimal course of action is not taken, partly because communication between nurses and physicians is inadequate, and because staff training and hospital procedure are not best suited to deal with rapid response to patient deterioration [17, 4].

In summary, clinicians recognise the importance of detecting early patient deterioration, but hospitals have previously been unable to respond optimally, due to lack of manpower, and appropriate technology and infrastructure. Current systems have started to address this problem, and are outlined in the literature review chapter. The primary purpose of this report is to document the development of an alternative solution originally proposed in our research group that aims to significantly improve on existing methods for detecting early deterioration in patient vital signs [60].

The solution uses data fusion techniques to combine vital sign data recorded from patient

monitors. It differs from many of the systems presented in the literature, which use a classification approach requiring a large training set including examples of rare clinical events. Instead, the method presented here is based on a novelty detection approach, whereby novelty is identified in new data when it is sufficiently different from a training set of normal data.

2 Literature Review

This chapter provides an overview of methods used to monitor seriously ill patients on acute wards (i.e. outside ICU). It is split into two sections. Firstly, methods that are part of current practice are analysed and critiqued. In particular, the concept of the Medical Emergency Team is explained and we discuss the criteria used to call these teams out to the ward. Following this, current concepts in the relevant literature are presented, including alternative calling criteria for the Medical Emergency Team, and intelligent alarm systems.

2.1 Medical Emergency Teams and Current Systems

In the introduction, it was stated that early intervention by clinicians can avert further deterioration in patient health. In an increasing number of hospitals, recognition of this has led to the introduction of the *Medical Emergency Team* (MET). The MET was initially introduced in Liverpool hospital New South Wales, Australia, and is becoming part of common practice in the US [13]. In the UK, a similar concept called the 'Critical Care Outreach Team' is used. An MET is a specialised team targeted towards patients at risk of cardiopulmonary arrest and other adverse events such as unscheduled admission to the ICU or emergency surgical procedures. The team has specific calling criteria such that any member of the nursing team may summon the MET when a patient deteriorates, a strategy commonly known as "track and trigger". While this strategy allows a rapid response once deterioration is detected, its effectiveness is still limited by the frequency of patient observations.

A number of studies have reported a marked reduction in mortality and morbidity associated with the seriously ill and those at risk from cardiac arrest when the MET system is present [44]. Bellomo et al. record a 65% reduction in cardiac arrests and 26% reduction in overall hospital death rate [5]. However, the design of these trials has been questioned by Smith and Nolan [54]. To date, only one Randomised Control Trial, the MERIT study has been conducted for MET assessment [22]. The study showed no improvements in patient outcome in the hospitals where an MET was used. However, the MERIT study investigators noted that one cause for this may be the fact that "*Even in MET hospitals,... monitoring, documentation, and response to changes in vital sign were not adequate*" . Further analysis by Jacques et al. also suggest that the MET calling criteria used were inadequate, and only detected late signs of deterioration[27].

ACUTE CHANGES IN:	VITAL SIGNS
AIRWAY	Threatened
BREATHING	All Respiratory Arrests, Respiratory rate < 5 breaths/min, Respiratory rate > 36 breaths/min
CIRCULATION	All Cardiac arrests Pulse rate < 40 beats/min Pulse rate >140 beats/min Systolic blood pressure < 90 mmHg
NEUROLOGY	Sudden fall in level of consciousness. Repeated or prolonged seizures
Other	Any patient who does not fit the criteria above whom you are seriously worried about.

Table 1: Medical Emergency Team Original Calling Criteria from Hourihan et al. [25]

2.1.1 Early Warning Scores Calculated by Nursing Staff

The original MET calling criteria are listed in Table 1. A wide range of calling criteria similar to those in Table 1 are in clinical practice. These are single channel calling criteria, so the MET is called when any one of a single parameter reaches a critical value. While this has the advantage of being easy for the nursing team to follow, it may be considered over-simplistic, and does not account for the fact that deterioration may initially involve smaller changes in a combination of parameters. A recent evaluation suggests that single-parameter criteria have poor sensitivity, and are unable to identify patients at risk of in-hospital death using a set of one-off vital signs recorded on admission to the ward [56].

One alternative set of calling criteria, the Modified Early Warning Score (MEWS) is based on the Acute Physiology and Chronic Health status Evaluation (APACHE) system for assessing patient condition in ICU. MEWS is an aggregate scoring system for use in acute wards outside the ICU, based on the breathing rate, heart rate, temperature, systolic blood pressure and an indication of consciousness (the Alert-Verbal-Painful-Unresponsive AVPU score). Points are allocated according to criteria such as those shown in Table 2, and the MET is called if the overall score exceeds a critical threshold. In principle, this is a good strategy as it assesses a patient's health based on all of the vital sign measurements simultaneously. Goldhill demonstrates the effectiveness of the scheme by showing an instance where the MEWS score recognises a serious incident before any single vital sign is considered abnormal [20].

In practice, there have been mixed results using MEWS. In studies by Smith and Wood [53]

Score	Systolic Blood Pressure (mmHg)	Heart Rate (beats/inm)	Breathing Rate (breaths/min)	Temperature ($^{\circ}C$)	AVPU score
3	<70				
2	71-80	<40	<9	<35	
1	81-100	41-50			
0	101-199	51-100	9-14	35-38.4	Alert
1		101-110	15-20		Reacting to Voice
2	≥200	111-129	21-29	≥38.5	Reacting to Pain
3		≥130	≥30		Unresponsive

Table 2: The Modified Early Warning Score. A score of 5 or more was defined as a 'critical score'

and Subbe et al. [57], MEWS has been shown to be effective. In initial retrospective tests, it was shown that high MEWS scores correlated with a greater risk of death or ICU admission. Pittard assesses the use of MEWS in practice, and showed a reduction in unexpected ICU admission, with better outcome for the emergency patients [45]. However, other studies, such as a second study conducted by Subbe et al. did not detect any discernible difference in outcome when using the MEWS system [58]. As well as questions concerning its effectiveness, its practical implementation has also been called into consideration. A study by Prytherch et al. suggests that MEWS is being inaccurately calculated by ward staff in a number of instances, and from a data set of 2607 vital signs recorded at a Medical Assessment Unit of Queen Alexandra Hospital, Portsmouth, 2.5% contained scores that should have triggered an alarm, but had been miscalculated [46]. While the effectiveness of MEWS is still not clear, such physiological scoring systems have been recommended by the UK's Department of Health and have been implemented in many UK hospitals in combination with a MET team [1].

Despite the quantity of research invested in early warning score systems, the tables used to calculate the MEWS scores (see Table 2) are empirical, being based on expert opinion. In response, Duckitt et al. have conducted studies aimed at adjusting the scoring thresholds so that their scoring system, the Worthing physiological scoring system, outperforms the MEWS system when tested on patient data from a general hospital [15]. A large number of studies have also attempted to modify the MEWS system by either adjusting the thresholds on the vital sign parameters, or by introducing additional parameters such as analysing urine samples

(for example, see Gardner-Thorpe et al. [18]). One recent study by Smith at al. analysed the effectiveness of the MEWS system by assessing the correlation between the MEWS score recorded upon a patient's admission to hospital, and subsequent in-hospital deaths for these patients [55]. The results suggest that such scores can be used as a predictor of patient outcome (and thus as a indicator for early intervention).

2.1.2 Single Channel Alarms from Continuous Monitoring

A patient can be monitored more frequently using single channel monitors such as an ECG monitor or a pulse oximeter, and in practice, clinicians on a general ward can request one of these for patients considered to be at greater risk. In this scheme, a vital sign is continuously monitored, and alert thresholds are set according to expert clinical knowledge. When the threshold is reached, an alarm sounds to alert nursing staff. This setup has the advantage of providing continuous monitoring, and begins to address problems due to infrequent observations.

Single channel monitors have high sensitivity and low specificity, however, suffering from the drawback of producing a large number of false alarms. In a study of 298 hours of data collected from patient monitors on an ICU, 86% of alerts were found to be false, and another 6% were clinically irrelevant [63]. A number of other studies also confirm that there is a low percentage of true alarms compared to total alarms [31, 41, 30]. Because of the excessive number of alerts, the alert thresholds are often set to make the monitoring less sensitive, thus reducing the effectiveness of the system. In many instances, ward staff take the further action of deactivating the alarms, rendering the monitor redundant [37] and there is also evidence to suggest that clinicians learn to ignore alarm noise [38], which may have disastrous consequences including an increase in patient mortality. The false alarms are generated from a number of simple causes, such as monitor probes detaching from the patient or in some cases, excessive patient movement.

There are a number of proposals in the literature for fixing, or at least circumventing the problem. The aim is to reduce the number of false alarms while remaining highly sensitive to patient deterioration. Most simply, median filtering can be applied to a signal to reduce a monitor's sensitivity to spike artifacts. This has a large effect, and one study claims a substantial reduction in false alarms for haemodynamic monitoring in intensive care [35].

Aleks et al. [2] note that simple median filtering fails to remove artifacts from an arterial-line blood pressure sensor in an ICU, as many occur for a substantial (up to 45 minute) length of time. To detect these, they propose using a Dynamic Bayesian Network, so that the observed variables (the mean systolic and mean diastolic pressure over one minute intervals) are modelled as a function of the previous blood pressure state, the true (and unknown) blood pressure and the known artifact generating processes. Because the distribution of artifactual event durations is known, the most probable explantion for the observations can then be ascertained and alerts can be generated as necessary. Furthermore, the true blood pressure state can then be inferred despite the presence of artifacts. Preliminary results have been very positive, with a false positive rate of less than 10% on 300 hours of data collected at 1Hz.

Sieben and Gather [52] attempt to reduce false alarms by classifying all alerts from a commercially available monitoring device into true positives and false positives. The classification procedure consists of building a 1000 decision trees, known as a Random Forest, trained on data from an ICU at the University Hospital Regensburg. Each tree is trained on a random subset of 200 data vectors and each vector consists of 10 vital sign measurements. The CART recursive partitioning algorithm is used to build the classification trees. When a new data vector is presented, it is "dropped down" each of the decision trees, and the vector is deemed alarm-relevant if a proportion of the 1000 trees have classified the data as alarm-relevant. The method was tested using clinical vital sign data, and reduces false alarms by 56% and 30% while maintaining a sensitivity of 95% and 98% respectively.

2.2 Multiple Channel Alarms

The rest of this section describes research on means of generating alarms by integrating information from multiple channels. The work presented below has mainly been tested on ICU patients, where continuous monitoring is standard, but the techniques can also be applied to acute wards outside the ICU.

2.2.1 Data fusion techniques

Studies have reported that in some cases, there may be multiple antecedents to an adverse clinical event [48]. Data fusion techniques attempt to use data from a number of different sources

in combination, rather than considering each source on its own, in order to increase the total knowledge about a system. By doing this, correlations between parameters might be inferred, and early warning of patient deterioration can be achieved. In the patient monitoring context, this often involves collecting data from a multi-parameter patient monitor, and processing it to produce an overall score of a patient's well-being. It should also be noted that a well designed system using data fusion techniques is also likely to reduce the number of clinically insignificant of false alarms [42].

Oberli et al. propose an expert systems approach to the data fusion problem [40]. An expert, or knowledge-based system is one that uses a direct encoding of human knowledge to help solve complex problems. In Oberli's system, the vital signs are first converted into a set of quantitative classes, such as "bradycardia" or "normal heart rate", based on training information given by a set of clinicians. The classes overlap and are described using fuzzy logic, so it is possible to be "somewhat bradycardic". After this, the patient's condition can be assessed using a set of logical rules derived from expert knowledge. For instance, if a patient was *Asystole AND extremely hypotensive AND no pulse detected*, he or she would be classified as having a cardiac arrest. In this way, the system provides a diagnosis as well as an alarm of patient deterioration.

Schoenberg et al. use expert knowledge in a slightly different way for data fusion [50]. In their scheme, a customizable "logic engine" is produced that is able to interpret information from multiple single-channel vital sign monitors. The purpose of the system is to filter out the clinically insignificant results. The system works by analysing a set of user-defined features, that can be extracted from the raw vital sign information (e.g. the change in average heart rate between the current minute and three minutes previous). Thresholds are set for each feature, based on expert advice, and a feature is assigned a score if it exceeds the threshold value. The sum of the scores is then compared to a critical total, which triggers an alarm if exceeded. The aggregate scoring system has many similarities to the MEWS system, and this technique can be considered as an automated and generalised scoring system. During tests on 120 hours of ICU data, the logic engine had a positive predictive value of 32% compared to 3% for standard monitors.

2.2.2 Data Fusion with Temporal Features

The systems described above primarily rely on the most recent observations to calculate a score using data fusion. However, much clinical insight can be gained from previous observations, as trends in the vital signs give a good prediction of future observations. Charbonnier and Gentil have attempted to incorporate historical data into an alarm system by making using of trend analysis [10]. In their system, each parameter is converted into a semi-quantitative temporal feature. Typically, the features are *{Increasing, Decreasing, Steady}*, and the quantitative information is the start and end time of the event, and the start and end value of the vital sign parameter. In order to make best use of the data, the trend features are aggregated to form the longest possible episode. A set of rules (such as Increasing + Increasing = Increasing) is used to accomplish this task. The extended features can then be used in a rule-based system, that alarms when the trend is persistent and severe. The system can also be trained to recognise artefactual events, and tests using this scheme on an ICU resulted in a 33% reduction in false alarms, without missing any clinically relevant alarms.

Temporal information can also be used to detect artefactual readings and reduce the number of false alarms. Hoare and Beatty analyse the time series for a set of physiological features, and attempt to predict the next values [23]. A new data point is then classed as artefactual if its value is outside a predetermined range. Williams et. al. extend this work, tackling the problem of infant monitoring immediately after birth, by using a Factorial Switching Kalman Filter (FSKF) to model vital sign data in neonatal intensive care [64]. A Kalman filter is a recursive filter that calculates the optimum estimates of process variables in the presence of noise. The FSKF extends this by allowing the filter to use different linear dynamic models that are selected by a switch variable, allowing 'normal' and 'artefactual' conditions can be modeled. The switch variable itself is dependent on a number of individual factors, such as 'Bradycardia' and 'Temperature Probe Disconnection'. Given a set of observations, the FSKF is then used to calculate the most likely switching state. By establishing which factors activate the switching state, the specific cause of the artifact and an estimate of the true value of the observed data. The system was tested retrospectively on eight infants of 28 weeks gestation, and a total of nine parameters were monitored. Results were then presented showing examples where the start and end times of specific conditions (such as bradycardia) were accurately detected.

2.2.3 Personalised Models

In clinical medicine, many patients behave in highly individual ways that may differ from the population average. It has previously been mentioned that vital signs are age-dependent, and other factors such as lifestyle and diet may have an effect. In response to this, Zhang proposes a personalised model that attempts to increase the alarm specificity by automatically tuning alarm thresholds on a per patient basis [66]. In his study, both neural networks and classification trees were tested, and used to generate the personalised alert thresholds. From these, neural networks performed consistently better. Typically, the system required eight hours of data, and the performance improved as the length of training data increased. Overall, the algorithm outperformed a simple threshold alarm system, and showed that personalised models are a feasible approach for alarm algorithms.

3 Creating a Model of Normality

The data fusion system known as Visensia® (previously known as Biosign) has been proposed to monitor and fuse four key patient vital signs in order to provide early warning of patient deterioration. The system was developed over the last seven years using a model trained on pilot study data acquired at the John Radcliffe (JR) Hospital, Oxford, and is already showing promising results [26]. This chapter describes the creation of the system, how the training set is used to build a probabilistic model, and how new vital sign data is interpreted to provide alerts of patient deterioration. An overview of the training procedure is provided in Figure 1.

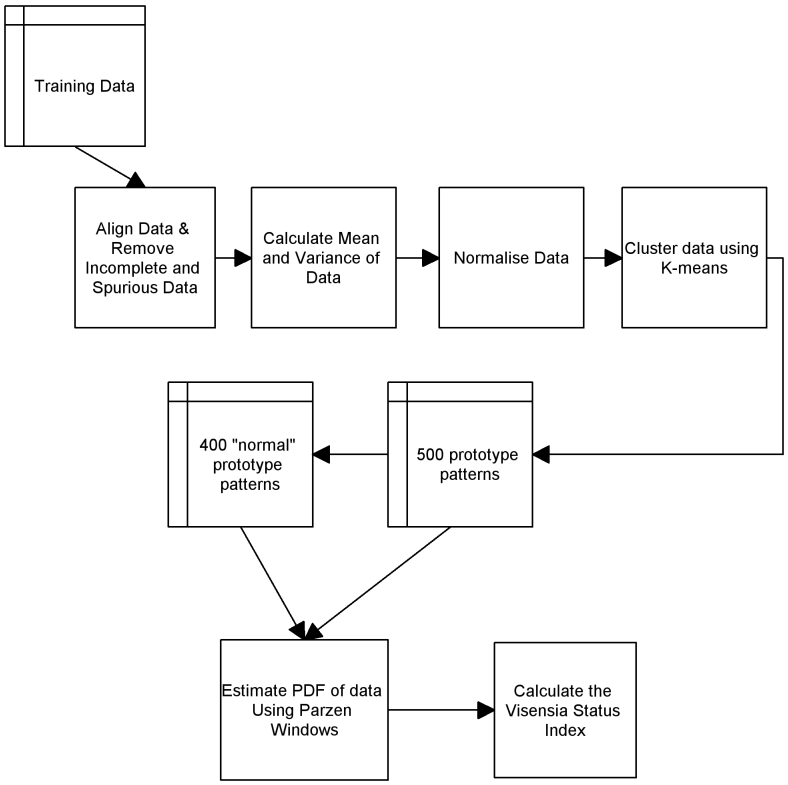

Figure 1: A flow diagram showing the steps involved in creating a model of patient normality and producing the Visensia Status Index (VSI)

3.1 Training Data

Pilot study data from the JR Hospital was used to train the Visensia model and also used to train methods for data visualisation. This data set consists of 3500 hours of data, with a set of five vital signs recorded for 150 patients during their stay on an acute ward. The vital signs measured were the Heart Rate, Breathing Rate, Temperature, Arterial-Oxygen Saturation (SaO_2) and Systolic-Diastolic average blood pressure. These were recorded using standard

patient monitors. Typically, the recordings were made at an appropriate rate to detect changes in vital signs. For most of the parameters, a sampling rate in the order of 1Hz was suitable - for instance, an abnormally high heart rate of 200bpm requires a sampling rate of approximately 6Hz. The exception to this was the blood pressure, which was recorded using a blood pressure cuff once every 30 minutes when the patient was awake, and once very hour while they were asleep in order to minimise patient discomfort. The monitor linearly interpolates the measurements for each vital sign and records values at twenty second intervals.

In the work described in this report, a more recent data set from Phase I of a three-phase clinical trial at the University of Pittsburgh Medical Centre (UPMC) is used to test the probabilistic model of normality. The UPMC data was collected in a 24-bed Step Down Unit (SDU), and contains measurements of the five vital signs mentioned previously. During Phase I, 28,782 hours of data were collected from 332 SDU patients, and the Visensia system was also run in the background for restrospective analysis. In the analysis, physiological data that met the MET calling criteria described in Table 1 were labelled as C' events. Each C' events was then reviewed by two clinicians independently, and they ascertained whether the event was of real clinical significance or not - those that were considered significant were subsequently labelled as C" events. In total, 237 C' and 112 C" events were identified.

Data from Phase III of the UPMC trial is also used later in this report to test visualisation algorithms and to validate the probabilistic model of normality. During Phase III, a further 18,692 hours of vital sign data were collected from 313 SDU patients and the Visensia system was also used 'online' to trigger patient review by nursing staff.

3.2 Pre-Processing the Data

In the first instance, the data in each of the sets was transformed into a form suitable for use in the training procedure and to enable basic statistical measures to be calculated. The vital sign data was recorded asynchronously, and so initially, it needed to be aligned into patterns, corresponding to 5D vectors of 'simultaneous' vital sign recordings. This was achieved by applying a zero-order hold between measurements, and collating patterns at ten-second intervals for each patient. Overall, 2.6×10^5 patterns were generated from the training data.

	Lower Bound	Upper Bound	*PI* - recordings rejected (%)
HR (bpm)	30	300	0.13
SDA (mmHg)	20	180	0.75
SaO_2 (%)	60	-	0.08
Temp. (°C)	32	39	48.00
BR (bpm)	3	45	0.00

Table 3: Physiological upper and lower bounds for the five Visensia parameters

3.2.1 Removal of Spurious Data

Patterns that were assessed to be physiologically implausible were rejected according to the lower and upper bounds given in Table 3. Furthermore, SaO_2 readings below 85% saturation were also discarded, as measurements below this are usually considered inaccurate by clinicians. This reduced the number of available training set patterns to 2.4×10^5.

3.2.2 Removal of Temperature Measurements

A large proportion of the temperature measurements recorded in the UPMC Phase I test data, just over 48%, were outside of the physiological limits described in given in Table 3. Figure 2 displays a typical temperature time series, showing a rapid decrease in temperature towards $22°C$ (ambient temperature). Most likely, this indicates an instance where the temperature probe had become detached. The figure also shows reattachment of the probe at 7800s. Despite setting the lower bound at 32^oC (see Table 3), the probe would record artifactual readings between $32°C$ and $36°C$ during the time it warmed up to the patient's true skin temperature of $36°C$, hence the number of false temperature recordings is even higher than the figure of 48% already given. Although methods of filtering the temperature data have been investigated[21], it was decided instead to discard the temperature recordings for robustness, so that only four parameters are used to build the model of normality.

3.2.3 Comparison of Data Sets

The variances and means for each vital sign were calculated and are shown in Table 4 for the UPMC Phase I and the original JR data. The other columns in the table show the absolute

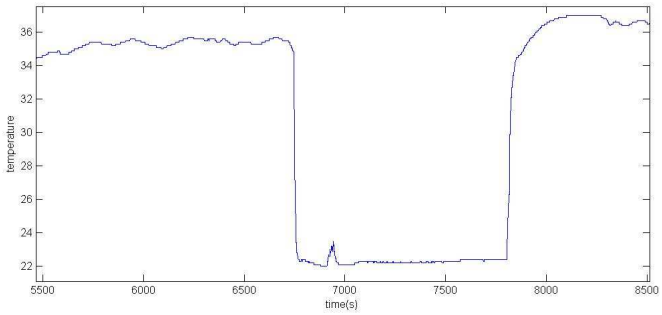

Figure 2: A short time series of temperature recordings for a patient, showing the detachment and reattachment of the temperature probe.

	μ_g	μ_{PI}	$\frac{\lvert\mu_g-\mu_{PI}\rvert}{\sigma_g}$
HR (bpm)	83.77	83.22	0.031
SDA (mmHg)	94.68	97.61	0.177
SaO_2 (%)	95.20	95.96	0.218
Temp. (°C)	36.05	35.87	0.094
BR (bpm)	18.30	18.61	0.061
	σ_g	σ_{PI}	$\frac{\lvert\sigma_g-\sigma_{PI}\rvert}{\sigma_g}$(%)
HR (bpm)	17.48	16.60	5.0
SDA (mmHg)	16.54	15.06	8.9
SaO_2 (%)	3.49	4.33	19.4
Temp. (°C)	1.27	1.16	8.7
BR (bpm)	5.06	5.12	1.2

Table 4: Comparison of the mean, μ_g, and standard deviation, σ_g of the training (JR) data set with those found in Phase I of the UPMC trial (μ_{PI}, σ_{PI}).

differences in means and variances between the two data sets weighted by the standard deviation. Most remarkably, the absolute differences are very small, indicating that the vital sign distributions are similar. This observation is confirmed by the histograms for each of the vital signs, which are presented in Figure 3. It is clear that the general shapes of the histograms are similar; for instance, the heart rate and breathing rate plots are positively skewed for each of the data sets. However, the Phase I UPMC data set contains many more measurements, so histograms for that data set are much smoother than those for the JR training data.

3.2.4 Normalisation

The data was normalised by assuming that each vital sign was normally distributed. However, it was noted that the heart rate distribution in Figure 3 is trimodal. This is due to the choice of very fine histogram bin widths, and the data appears Gaussian when plotted at a coarser scale. Beaing this in mind, the Gaussian zero-mean unit-variance transformation was applied

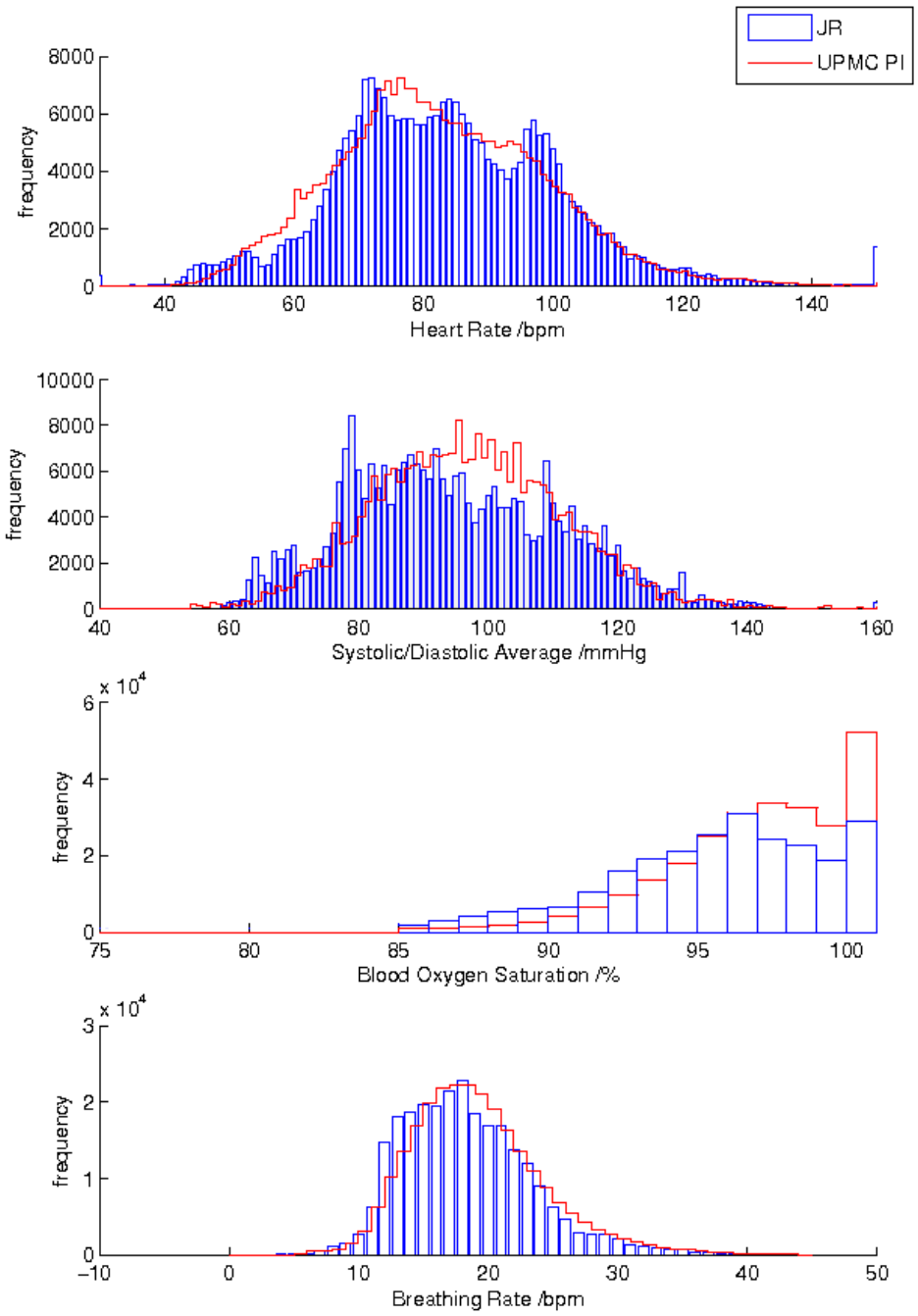

Figure 3: Histograms of the four vital signs in the JR(blue) training set and UPMC Phase I(red) test set. The graphs for the UPMC data have been scaled by an appropriate factor so that the distributions can be compared. Also, the temperature recordings have been removed from the data sets, as described in Section 3.2.2

on each measurement x.

$$x_n = \frac{x - \mu_g}{\sigma_g} \tag{1}$$

where x_n is the normalised value, μ_g is the training set mean, and μ_g is the training set standard deviation as calculated in the previous section. The zero-mean transformation was required to force each of the vital signs onto a similar dynamic range so that each has an equal influence on the model of normality. Finally, data patterns that contained missing information (for instance, where probes had become unattached) were discarded from the training set.

3.3 Clustering

After the initial processing stage, the JR training set consisted of 2.6×10^5 four-dimensonal vital sign patterns and the UPMC test set consists of 1.2×10^6 four-dimensional vital sign patterns. In order to solve the problem with a manageable number of patterns, the data was summarised into a smaller number of 'prototype' patterns. A clustering technique, the K-means algorithm, was used to select the prototype patterns.

The K-means clustering algorithm is outlined in *Algorithm 1* and was first introduced by Macqueen in 1967 [34] to cluster objects based on attributes into k partitions. Mathematically, the algorithm aims to minimize the error function:

$$E = \sum_{i=1}^{k} \sum_{x_{j \in S_i}} |x_j - \mu_i|^2 \tag{2}$$

where there are k clusters, S_i represents the i^{th} cluster, and μ_i is the centroid of all the points in the set S_i. k clusters are initially randomly placed in the data space, and each data point is assigned to its nearest cluster centre. The positions of the cluster centres are then recalculated, by moving each cluster centre to the centroid of its cluster. The data points are then reassigned to the nearest cluster centre, and the procedure is repeated until convergence. The algorithm is terminated when the difference in E between iterations falls below a user-defined criterion.

A modification of the Netlab[39] K-means[1] implementation was used on the training data in order to avoid computer memory limitations. The data was sorted into 500 clusters, and

[1]Netlab K-means was originally edited by edited by Susannah Fleming, Biosignal Processing Group, University of Oxford

Algorithm 1 The k-means algorithm

1. Place K initial points into the space represented by the objects that are being clustered
2. Assign each object to the group that has the closest centroid
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat until convergence

then the centroid of each cluster was defined as a prototype pattern. The number of prototype patterns selected was chosen empirically but will be reviewed later.

3.4 The model of normality - Parzen Windows

At this stage, we make an assumption that the vital sign data is drawn from some underlying probability distribution function (p.d.f.). Using the 500 prototype patterns selected previously, the training procedure aims to estimate this p.d.f. If the underlying p.d.f. is known, then the probability of any new vital sign pattern can be ascertained. The probability can then provide an indication of a patient's status, as improbable vital sign patterns will indicate abnormal health.

Parzen Windows is a statistical technique developed to estimate the probability density function of a random variable. The primary advantage of the Parzen Windows technique over other p.d.f. estimation methods such as Gaussian Mixture Models is that it is non-parametric, so it imposes no prior assumption on the shape of the distribution. The Argonauts reading group[2] provides an informal and succinct explanation of Parzen Windows: "Given a sample of a random variable X for which we want to estimate its p.d.f., we put a kernel function (eg. a Gaussian) on top of each point, and the estimated pdf is the linear combination of them, normalized so that the integral of the pdf is 1."

Mathematically, Parzen windows can be described as follows: if $x_1, x_2, ..., x_N \sim f$ is an independent and identically distributed sample of a random variable, then an approximation of its p.d.f. can be written as

$$\hat{f}_h(x) = \frac{1}{Nh} \sum_{i=1}^{N} K(\frac{x - x_i}{h}) \tag{3}$$

[2]Argonauts are part of the robotics research lab at Oxford University. Minutes of their meetings can be found at *http://www.robots.ox.ac.uk/~rcasero/wiki/index.php/Argonauts_website*

where K is some kernel function, N is the number of centres, and h is a smoothing parameter. In this case, the kernel is taken to be a Gaussian function with zero mean and unit variance:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} \tag{4}$$

Analysis has shown that the choice of kernel is not crucial for estimating the p.d.f. in the case of independent identically distributed random variables. The smoothing parameter or kernel width, h, can be chosen by calculating the mean euclidean squared distance of the m closest members in x for each of the $x_1, x_2 ... x_N$, and then calculating the mean of these N local means, as first proposed by Bishop [7]. Applying this to the prototype centres with *m=10* members, gives a value of $h = 1.49$.

In the training procedure, a Matlab function *parzen_windows.m* was created to build a Parzen Windows estimated p.d.f. of the 4-D vital sign data for the training data set, where values of the random variable x were described by using the 500 prototype points. As a sanity check, a successful search in 4-D space was attempted, varying one parameter at a time to confirm that the p.d.f. value at the 'most normal point', [0 0 0 0], is close to the maximum.

3.5 Visensia Status Index (VSI)

The Visensia Status Index, which is also known as the Patient Status Index and Novelty Index in related literature [21, 60], is calculated from the probability density function as follows:

$$VSI = log[\frac{1}{p(x)}] - log[\frac{1}{p_{max}(x)}] \tag{5}$$

Where $p(x)$ is the p.d.f. evaluated at the pattern x, and $p_{max}(x)$ is the p.d.f. evaluated for the pattern [0 0 0 0], which represents the maximum possible value of $p(x)$. This term is subtracted to adjust the scale so that the Visensia Status Index is zero when all the vital signs are normal. The unconditional probability is transformed in this way so that the score represents a measure of abnormality, where a high score indicates highly abnormal vital signs.

Figure 4 shows the change in the Visensia Status Index when each of the normalised vital signs are ramped from -4 to +4 in turn, apart from SaO_2, which reaches 100% saturation at a normalised value of 1.37. The VSI values for each parameter range between zero and five, and

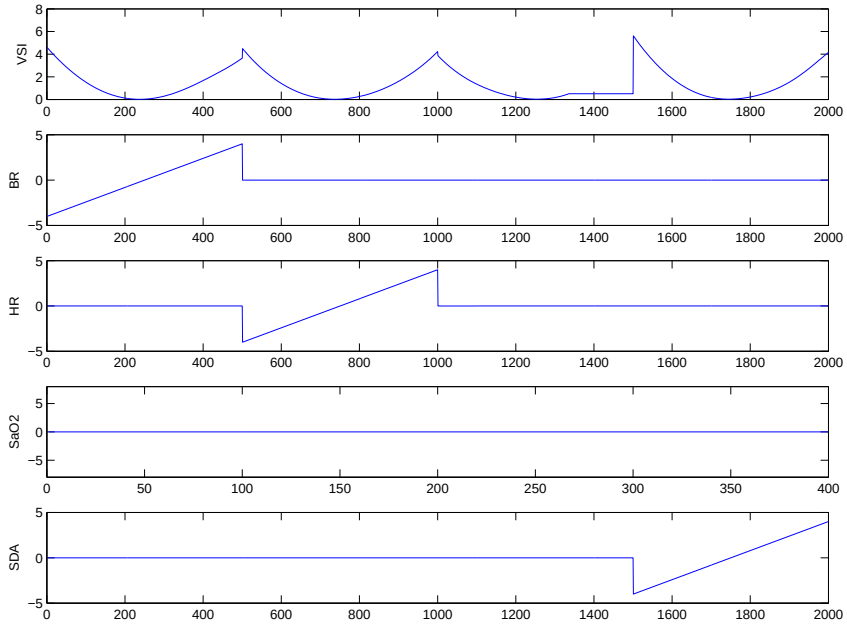

Figure 4: Visensia Status Index for the case when each vital sign parameter is ramped from -4 s.d. to +4 s.d. in turn, while fixing the remaining three vital signs at their normalised mean value of zero.

the VSI distributions are slightly asymmetrical for all of the vital signs. This may be expected as the histogram plots in Section 3.2.3 show that the original distributions were also skewed.

When varying SaO_2, the minimum VSI is approximately zero, which occurs at 95% blood oxygen saturation. The VSI then increases as blood oxygen saturation reaches 100%. This is anomalous as one would expect a patient to be healthier with higher levels of oxygen saturation, and is a result of the normalisation stage in Section 3, which implicitly assumed the data to be Gaussian distributed. The histograms in Section 3.2.3 shows that this is not the case. However, in this instance, the difference in VSI at 100% SaO_2 and at the minimum is small, and so the assumption is acceptable. Furthermore, alternative ways of normalising the SaO_2 data were tested by Hann [21] and were found to have negligible effect on the overall outcome.

The VSI can be used directly as a measure of patient abnormality, and can thus be used in an alarm system to provide early warning of patient deterioration. It can also be made an MET calling criterion, so that an MET is called once the VSI reaches a critical value. Because the VSI is a monotonic function of the unconditional probability $p(x)$, a VSI upper threshold is also equivalent to setting a lower threshold on $p(x)$.

A candidate threshold was set at 3 VSI units. Its suitability was assessed by observing

	lower threshold		upper threshold	
	Vital sign	Deviation from mean (σ)	Vital sign	Deviation from mean (σ)
HR (bpm)	25.91	-3.31	141.63	3.31
SDA (mmHg)	45.89	-2.95	149.10	3.29
SaO_2 (%)	83.00	-3.5	N/A	N/A
BR (bpm)	1.96	-3.23	34.95	3.29

Table 5: Values of individual parameters that cause the novelty to exceed VSI = 3.0, when the other three parameters are set to 'normal' (the mean value in the training set). Both actual values and normalised value (in units of standard deviation) are given.

the value at which the candidate threshold was reached for each individual vital sign, while fixing the value of the other three parameters to their means (i.e. when the other parameters are 'normal'). The result of this is shown in Table 5, and demonstrates that for any single vital sign, a VSI of 3 is reached between 3.0 and 3.5 standard deviations from its mean value. For a general distribution, Chebyshev's inequality shows that the probability of a single vital sign causing the threshold to be exceeded is between $P[(\frac{x-\mu}{\sigma}) > 3.0] = \frac{1}{3.0^2} = 0.11$ and $P[(\frac{x-\mu}{\sigma}) > 3.5] = \frac{1}{3.5^2} = 0.081$. However, the histograms in Figure 3 show that the data is well modelled by a Gaussian distribution, in which case the probability is far smaller, between $P[(\frac{x-\mu}{\sigma}) > 3.0] = 0.013$, and $P[(\frac{x-\mu}{\sigma}) > 3.5] = 0.0023$. The candidate threshold of 3.0 is thus suitable, as it will alert for highly improbable single-channel events.

With this method, an alarm will be generated when any single parameter reaches a critical value, OR when a combination of parameters produces a high VSI score. In order to reduce false alarms while maintaining a high sensitivity, alarms are only generated if the VSI remains over the critical value for a prolonged period of time. In particular, this eliminates instances where the threshold his exceeded momentarily due to movement artifacts. Previous analysis have shown that a high VSI for 240 out of 300 seconds is significant enough to warrant hospital staff attention [21].

4 High-Dimensional Data Visualisation

Interpreting results arising from the Visensia model described previously is a challenging task, as each data vector contains four independent vital sign readings, and the data set is very large. Any analytic tools or algorithms must deal with the data in a coherent and intuitive manner in order to provide useful insight, but must also be usable with large volumes of data. One important tool in high-dimensional data interpretation is visualisation. This involves transforming the original data to a visualisation space with fewer dimensions. Typically, two or three dimensions are chosen so that the results can be plotted for visual inspection. The transformation is chosen in such a way as to maintain key aspects of the data distribution; for example, topology may be preserved between the dimensions.

4.1 Sammon Maps and Neuroscale

A variety of visualisation algorithms have been proposed in the literature, including Kohonen's Self Organising Maps (SOMs) [29] and kernel Principal Component Analysis (kPCA) [49]. SOMs use a neural network to map data onto a 2D grid such that similar data (i.e. data close to each other in the original high-dimensional space) are grouped together on the grid. This provides insight into the spatial relations within the data. In kernel PCA, the appropriate choice of kernel allows the data to firstly be mapped to a higher dimensional space so that a standard PCA in kernel space has the effect of producing a non-linear mapping between the original data space and visualisation space.

One popular alternative to these methods is the Sammon Map algorithm [47]. This produces a mapping that attempts to keep the Euclidean distances between all pairs of data points in the 2-D visualisation space as close as possible as they were in the high-dimensional data space. Mathematically, this is equivalent to minimising the so-called Sammon STRESS for N data samples:

$$STRESS = \frac{1}{\sum_{i=1}^{N}\sum_{j>i}^{N} d_{ij}^{*}} \sum_{i}^{N}\sum_{j>i}^{N} \frac{(d_{ij}^{*} - d_{ij})^2}{d_{ij}^{*}} \tag{6}$$

where the Euclidean distances between patterns i and j in the data space are denoted by d_{ij}^{*}, and the corresponding distances in visualisation space are denoted by d_{ij}. The objective

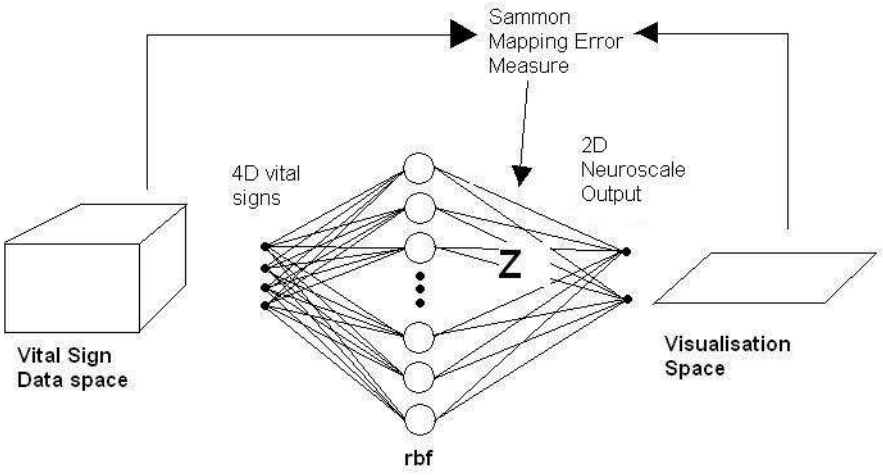

Figure 5: Schematic diagram of the RBF neural network used to create the Neuroscale mapping between vital sign data space and 2D visualisation space.

function is minimised by a gradient descent technique that adjusts the position of the points in visualisation space.

Unfortunately, there are two major drawbacks to the method. Firstly, creating a Sammon Map is not possible for very large data sets, as the STRESS calculation required involves order $O(N^2)$ point comparisons. On a typical desktop PC, a few thousand data vectors is the practical limit. This means that large data sets such as those obtained from the JR and UPMC trials cannot currently be visualised using Sammon Maps. Secondly, the Sammon Map does not explicitly define the transformation between data space and visualisation space and so cannot accommodate new data. The only way to map a new point is to add it to the data set and reoptimise the STRESS.

The Neuroscale algorithm was developed by Lowe and Tipping [33] and attempts to circumvent these problems by parameterising the Sammon mapping transformation such that $y=f(x,z)$, where y is the vector of points in visualisation space, x is the vector of points in the data set, and z is a parameter vector. The STRESS can then be differentiated with respect to z, so that the vector y is indirectly adjusted while minimising STRESS. In this scheme, the transformation is explicit, allowing new data points to be added to a map. A Radial Basis Function (RBF) neural network is used to learn the transformation between data and visualisation space, and a schematic diagram of the network is shown in Figure 5. In tests conducted using Neuroscale, the number of hidden nodes was chosen to be an order of magnitude lower

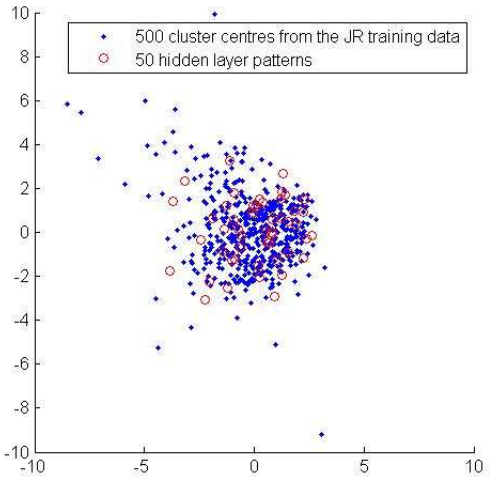

Figure 6: An example Neuroscale map showing the 50 hidden later points and the 500 K-means patterns generated from the training data.

than the number of training patterns used. However, Lowe and Tipping note that the algorithm is largely insensitive to model complexity.

The Neuroscale algorithm was tested using the JR training data; 50 patterns were chosen as the locations of the RBF network hidden nodes by using K-means to cluster the 500 prototype patterns originally selected from the JR training set. The width of each centre was then set to be the Euclidean distance of the centre to its furthest neighbour amongst the 49 other centres.

The RBF network was trained using the Netlab [39] implementation of the Neuroscale algorithm and then used to visualise the 4-D vital sign patterns. A sample output showing the hidden layer centres and the 500 K-means patterns generated from the JR training data is shown in Figure 6. The 500 centres are well distributed over the feature space, as one may expect. The figure also shows a small number of outlying centres, which may indicate that the training data includes periods of vital sign abnormality.

Although Neuroscale is useful in many situations, it suffers from some of the same drawbacks as the Sammon map. In particular, the STRESS calculation is still required, so the computational cost for large data sets remains very high. To some extent, this problem can be avoided by training on a subset of prototype points selected from the whole data set as first performed by Tarassenko et al. [62]. However, the quality of the transformation created by such a procedure is highly dependent on the method used to select the prototype points, and may be particularly poor in sparsely populated regions of the data space.

At present, there appears to be no method described in the literature for creating a Sammon Map for large ($> 10^4$ points) data sets in a reasonable time.

4.2 SASS - a novel visualisation tool

A novel alternative to these methods is proposed here, named the Sparse Approximated Sammon STRESS (SASS). SASS reduces the problem to one of order $O(n)$ by sub-sampling from the complete set of inter-point distance pairs to approximate the Sammon STRESS. In practice, it has been discovered that many of the inter-point distances can be removed from the STRESS calculation, with little effect on the Sammon Map visualisation. The method used to sub-sample is critical for obtaining an accurate mapping and is discussed further in the following section. Formally, if we define S to be a sparse subset of the index pairs (i, j) for which the Euclidean distance is calculated, then the modified STRESS function to minimise is:

$$SASS = \frac{1}{\sum_{i,j \in S} d^*_{ij}} \sum_{i,j \in S}^{N} \frac{(d^*_{ij} - d_{ij})^2}{d^*_{ij}} \tag{7}$$

For very large data sets consisting of at least $N = 10^6$ points, a sparse distance matrix with an average of 50 distance comparisons for each point has been tested and shown to work successfully. In this case, only one distance comparison is computed using SASS for every 20,000 comparisons required to be calculated for the original STRESS. By reducing the computational complexity in this way, the initial problem of large data sets is overcome. Furthermore, data storage is reduced by using memory saving techniques for sparse matrices. Further increases in speed are made by using an efficient optimisation algorithm, scaled conjugate gradients, in preference to gradient descent.

4.2.1 Algorithm Initialisation

In the initial tests, points in the visualisation space were initialised with random values, following the precedent set in Sammon's original paper. During these tests, it was clear that as the size of the data set increases and the number of computations required to calculate SASS increases accordingly, it becomes likely that the SASS optimisation procedure will get stuck in a local minimum. In addition, some of the points are likely to be initialised far from their final position, so the optimisation process converges slowly.

SASS can be initialised in a more principled manner by using a two-stage approach. Firstly, SASS is applied to a subset of the initial data set to produce a preliminary mapping. In this pre-mapping, the points in the visualisation space are initialised randomly. The Sammon map generated by this process creates a sparse outline, or a skeleton, of the data and so the second stage of the initialisation is to approximately map the remaining points into visualisation space using the skeleton. A variety of techniques for this are recorded in the literature, and are assessed in Section 4.3. In this case, a distance mapping technique introduced by Pekalska et. al. was used, which creates an explicit linear transformation between the data and visualisation spaces. This provides an approximation to the Sammon mapping transformation, which is generally non-linear. The result of this procedure is that all vectors in the data set are initialised to the correct region of the visualisation space. A fuller explanation of the distance map method implemented is provided in Section 4.3.2.

In preliminary tests on the UPMC Phase III data sets, initialisation takes nine minutes to perform for a 4800 point pre-mapping. The vast majority of this time is spent calculating the distance matrices between the 4800 points and each point in the 10^6 point data set on a desktop PC. Following this, SASS was run using the pre-mapped points to initialise points in the visualisation space. In general, the final SASS error was smaller than for random initialisation of the d_{ij} values, and the optimisation stage converged in fewer iterations.

4.2.2 Subset S Initialisation

The SASS method can fail when a subset of the data, by chance, only possesses inter-point comparisons within the subset. A pictorial representation of this problem is presented in Figure 7. It is unsurprising that such an initialisation results in an incorrect visualisation, as the algorithm will treat the subsets as two separate data sets.

Fortunately, the probability of such an event occurring is very small. For instance, the probability of two subsets forming, where one of the subsets contains only one vector (which is equivalent to one point having no connections to any other point in the data set), is given by

$$P(\textit{one point disconnected}) = N\left\{1 - \frac{2}{N}\right\}^{\frac{\lambda}{2}N} \tag{8}$$

where λ is the average number of connections per point such that $\frac{\lambda}{2}N$ is the number of

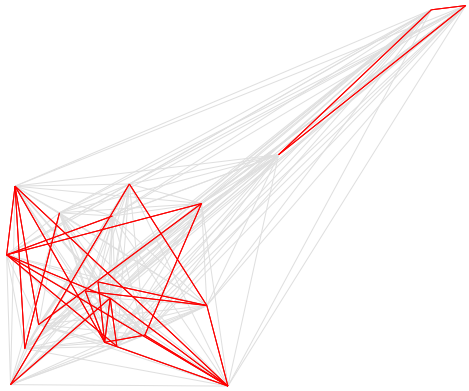

Figure 7: The graph shows an example of the connectivity between data points for the original Sammon Map (grey) and for the SASS algorithm (red). Each node represents a data point, and each edge represents an inter-point distance. In this example, the data has formed two unconnected subsets and SASS will produce an incorrect mapping.

elements in set S, and N is the number of data points as before. For a data set with over 10^6 points and an average of 50 connections per point, the probability of one point being disconnected is of the order of 10^{-16}. To prevent this problem from occurring at all, we ensure that the connections within the data set first form a minimum spanning tree. The simplest way to do this is to initially connect each data vector to its neighbours (in the Euclidean distance sense), so the n^{th} data vector in the set N has distance comparisons to the $n-1^{th}$ and $n+1^{th}$ vectors.

SASS can be further enhanced by considering the manner in which the subset S of inter-point connections is chosen. In order to test the effectiveness of alternative choices of S, a unit-cube synthetic data set was created. This consisted of a 3D unit-cube with normally distributed data at each of the corners, with 20×10^4 three dimensional vectors in total. Furthermore, the $[1, 1, 1]$ data vector was added twice to the set as two distinct data points to test whether data are mapped consistently.

In the initial tests, elements in S were chosen by selecting two data vectors at random. Figure 8 shows the results from SASS on the cube data set in red compared to results created directly from Sammon's algorithm in grey. The eight clusters corresponding to the corners of the cube are correctly mapped, and it is clear that SASS works satisfactorily. One should note that there is more than one correct solution, including any Sammon map that possesses reflective or rotational symmetry with respect to the solution shown.

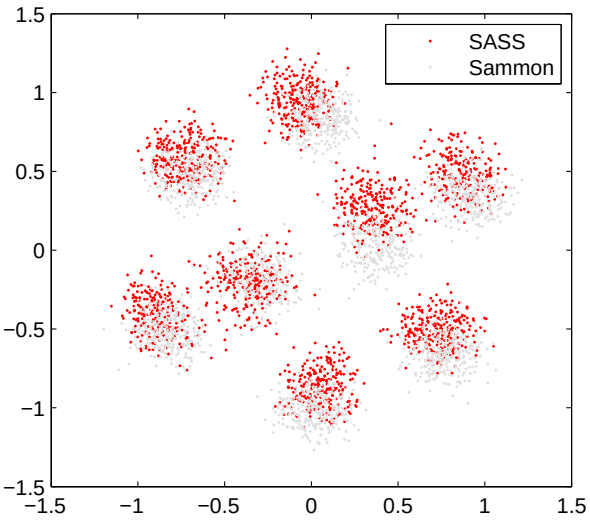

Figure 8: A Sammon Map for the unit-cube data set, containing 2×10^4 points. The SASS Sammon map is shown in red, and the output from the original method is shown in grey. In both instances, the separate data clusters are clearly visualised.

Although the results are acceptable, it is clear that, in order to maintain accurate local and global structure, the proportion of local and distant inter-point connections is of critical importance. One natural way to achieve this is to force each data point to have an equal number of comparisons to both near and far points in the data set. Local and distant points can be defined for any data set as follows. Firstly, the data is clustered using a technique such as K-means. Once the data has been grouped, half of the inter-point comparisons that form set S are selected from data vectors from within the same cluster. These are defined as 'local' connections. The remaining inter-point comparisons are chosen so that the data vectors are from different clusters. Alternatively, for time series data where variation is slow compared to the data collection rate, one would expect consecutive samples to appear locally in visualisation space. Therefore, local connections can also be defined by appropriately partitioning a time series.

The unit-cube data set was retested using this method to define local and distant connections. Again, Figure 9 shows that the global structure was adequately captured. The duplicate points (the [1 1 1] vectors) are highlighted in red, and visual inspection shows that they were mapped consistently. To quantify the accuracy of the mapping, the data set was visualised 200 times for both the randomly initialised set, and for the alternative method described above. In each of the 200 Sammon maps, the Euclidean distance between the duplicate points was

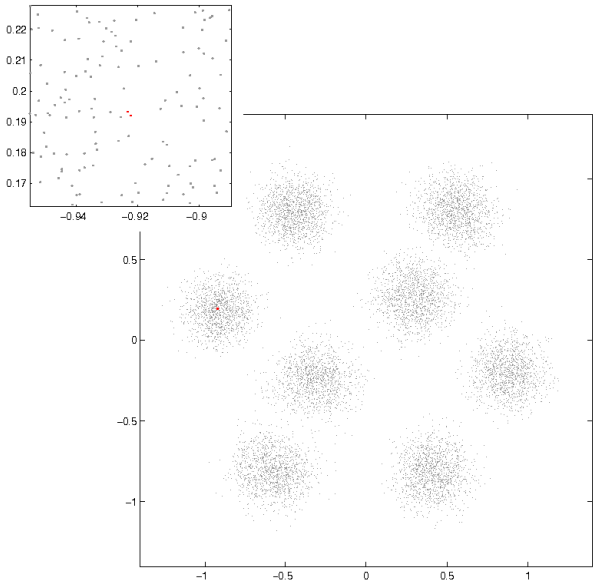

Figure 9: A Sammon Map for the unit-cube data set using the initialisation described in Section 4.2.2. The SASS Sammon map is shown in grey. In the left-most cluster, the visualisation of the duplicate points at [1,1,1] are highlighted in red. The sub-figure shows the left-most cluster in greater detail so that the duplicate points can be distinguished easily.

recorded and the mean of these was calculated. For the randomly initialised set, the mean distance was 0.05, while the mean distance for the K-means method was 0.02. This indicates that it is important to ensure a sufficiently high proportion of local connections, and that selecting S at random is sub-optimal.

4.3 SASS generalisation - interpreting new data

It has been mentioned that one of the significant drawbacks of all Sammon mapping techniques, including the SASS method, is that new data cannot easily be visualised on an existing Sammon Map. A number of algorithms attempt to interpolate new data points onto Sammon maps. These have proved moderately successful in practice, but often provide coarse estimates in visualisation space, and turn out to be inelegant solutions. An outline of two such techniques is provided below. Section 4.3.3 provides a novel adaptation of a distance mapping approach that takes advantage of the greater data density in the problems suited to SASS by creating a *local* distance-mapping that outperforms the standard method.

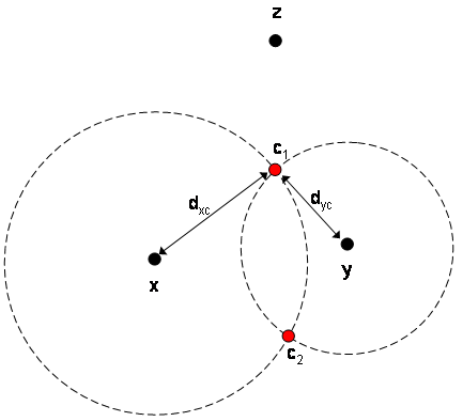

Figure 10: Schematic Diagram of a triangulation scheme. Points **x**,**y**, and **z** are already mapped. The distances $\mathbf{d_{xc}}$ and $\mathbf{d_{yc}}$ are the inter point distances in data space between points **x** and **c**, and **y** and **c** respectively. In 2D space, this defines up to two candidate points for **c**. The correct candidate is chosen by using **z** for a further distance comparison.

4.3.1 Triangulation

Triangulation techniques, first considered by Lee et. al.[32], are based on the idea that for any given data point, only the inter-point distances to N other points can be preserved exactly in an N-D visualisation space. For the 2D case, two data vectors that are local to the new data vector in the data space are used to define either zero, one or two candidate points for the new data vector to be mapped to. If two points meet the distance criterion, then a third data vector is used to identify the correct candidate point. A schematic diagram of the process is provided in Figure 10.

In the Sammon Map algorithm, the mapping of a candidate point is dependent on preserving distances to all the remaining points. However, in triangulation the mapping of the candidate point depends on only three previously mapped points, and so the method is prone to error. Large errors are especially likely to occur with a poor choice of triangulation vectors, such as when the vectors are colinear.

4.3.2 Distance Mapping

Distance mapping is a technique developed by Pekalska et. al. [43] that involves using a subset of data to create an explicit linear transformation from data space to visualisation space. This

provides an approximation to the transformation created by Sammon's algorithm, which is generally non-linear. In the original context, distance mapping was used to speed up the Sammon map visualisation. In this section, we describe how the algorithm can be applied to a large data set for incorporating previously unseen data.

Firstly, a subset, $\mathbf{X_{base}}$, of the training data set is chosen (for smaller data sets, the whole training data can be used). Ideally, this subset should provide good coverage of the data space. This can be achieved by using the process described in Chapter 3 for selecting the 500 prototype patterns from the whole training set.

Next, we assume that there is a linear transformation, $\mathbf{V}$, from the subset $\mathbf{X_{base}}$, to their corresponding points in visualisation space, $\mathbf{Y_{base}}$ that can be defined by

$$\mathbf{D_{base}V = Y_{base}}$$

where $\mathbf{D_{base}}$ is the distance matrix of all the pairs of vectors in$\mathbf{X_{base}}$. From this, it is simple to solve for $\mathbf{V}$, and to incorporate some new data, $\mathbf{X_{new}}$, by first calculating the distance matrix between the sets $\mathbf{X_{base}}$ and $\mathbf{X_{new}}$ to establish $\mathbf{D_{new-to-base}}$ and then finding $\mathbf{Y_{new}}$ by:

$$\mathbf{Y_{new} = D_{new-to-base}V}$$

4.3.3 Local Distance Mapping

In Pekalska et. al's distance mapping algorithm, it was assumed that a linear transformation between data space and visualisation space was adequate. Their work demonstrates that this is true in many cases. However, it is possible to improve upon the algorithm by only selecting points for the subset, $\mathbf{X_{base}}$, that are local to the data point to be mapped. This has the advantage of making the more general assumption that a linear mapping is accurate for a local region of space. A 'local' point can be defined by using the K-means algorithm to cluster the data and then selecting only points within the appropriate cluster. Compared to the standard distance mapping technique outlined previously, the main disadvantage is that extra computation is required to calculate neighbouring data points, and that a new linear transformation matrix, $\mathbf{V}$, must be calculated for each point added. The computational cost is offset when one considers that the size of the local subset can be much smaller and still provide good results.

We note that Pekalska et. al. do not consider this adaptation to their method, perhaps because a large data set is needed to demonstrate its effectiveness.

4.3.4 Comparison of Methods

In order to test the effectiveness of the three methods, the unit cube data set was again used. The test was conducted as follows: firstly, the data set was mapped using SASS and one point was removed from the visualisation space. Secondly, the data vector corresponding to the removed point was mapped back into visualisation space using the three interpolation methods, and the error was calculated as the distance between the SASS Sammon mapped point and the distance mapped approximation. The test was repeated for all data vectors in the set. It was discovered that the local distance mapping algorithm greatly outperforms the standard distance mapping algorithm, reducing the average error from 5.0×10^{-3} to 4.3×10^{-4}, and also outperforms triangulation (with an average error of 5.0×10^{-2}) but is much slower than the other two methods. Despite this, we note that local distance mapping of a single point takes hundredths of a second to compute on a desktop PC and is suitable for real-time applications.

4.4 Results

The SASS method has been used as a tool for initially exploring the extremely large vital sign data sets. Results so far have been encouraging, and have provided insight into ways of improving data fusion models for patient monitoring. The following results were generated using the UPMC phase I and phase III vital sign data.

In figure 11, a SASS map generated from Phase III data is shown, visualising approximately 10^6 data points. Each point in the figure is a 2D representation of a 4D vital sign vector. Using the Local Distance Mapping technique described previously, the Phase I data that corresponds to readings during the 10-minute window from 5 minutes before and after a C" event (that is, a vital sign recording during a clinically important event) are shown in red. The plot shows that most of the data from near a C" event lies towards the edges of the data distribution, as one would expect for abnormal vital signs. Some of the data is plotted at the centre of the Sammon mapped cluster. By checking the original data points, it can be shown that these readings indicate instances where a patient experienced a sudden deterioration, or else instances when

a patient's condition stabilised quickly after the C" event.

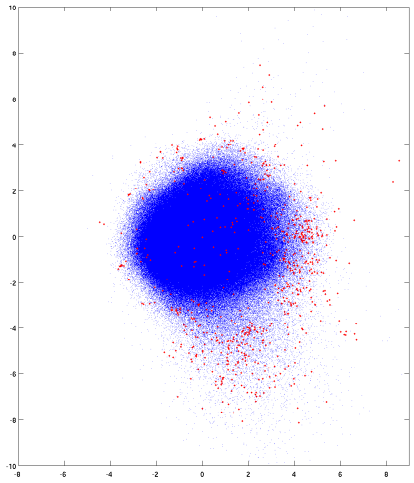

Figure 11: A Sammon map of the UPMC Phase III data in blue. The points in red represent vital sign data from 5 minutes before and after a C" event in the UPMC Phase I data, and have been added to the map using the local distance mapping technique.

The same technique can be used on a per-patient basis and allows one to see deterioration in patient health as time progresses. For the phase III data, a series of SASS maps of patient 151 were created to depict the final ten minutes of the patient's record (Figure 12). The vital sign record for patient 151 is coloured in white for reference, and the entire record of vital signs recorded during the trial are plotted in light grey. The maps clearly show how the patient begins with relatively normal readings, which lie towards the centre-left of the population's distribution. As time progresses, the patient's vital signs become increasingly erratic as the blood-oxygen saturation readings become dangerously low. The bottom row of plots correspond to the last three minutes of the patient record where it can be seen that a number of abnormal vital signs are recorded, denoted by the points towards the edge of the grey (whole population) cluster and far away from the white (single patient) cluster, and it can be seen that there is a general trend away from normality and increase in vital sign variability. This was caused by a combination of an increase in blood pressure, and a catastrophic drop in arterial oxygen saturation. The fact that the deterioration in patient health can be detected suggests that it is possible to use trends in time to improve the data fusion model.

A final SASS example is presented in figure 13. This Sammon Map depicts the vital signs for patient 173 and patient 182 from the Phase III data in red and blue respectively. It is

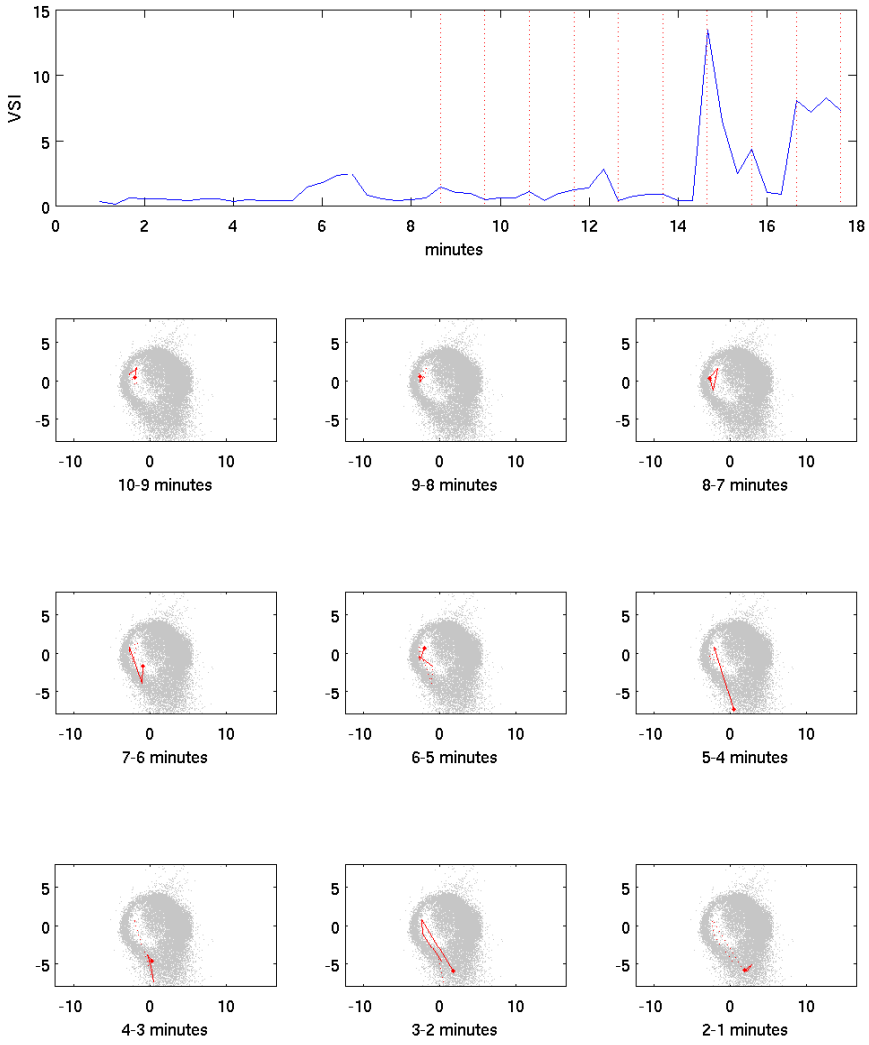

Figure 12: A time-lapse Sammon map showing the deterioration in health of patient 151 during the last 10 minutes of the patient's vital sign record. The points in light grey depict the vital sign distribution from the entire data set, while the points in white show the vital signs for the patient's entire stay on the ward. The lines in red mark the progression of the patient's vital signs over a one minute period. The VSI is plotted at the top, and the red lines indicate minute markers corresponding to Sammon maps.

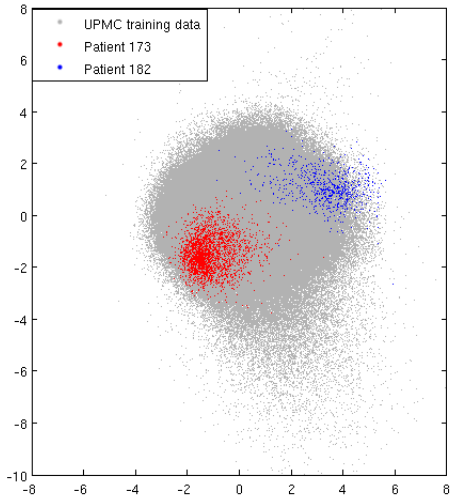

Figure 13: A Sammon Map for UPMC vital sign data. The whole data set, consisting of 961,031 4D vectors is visualised in grey. Points corresponding to vital signs for patient 173 and patient 182 are plotted in red (left) and blue (right) respectively.

noticeable that the vital signs for each patient are confined to small regions of the vital sign distribution. For both sets of data, the medical emergency team had not been alerted, and so the vital signs may be considered 'normal'. This indicates that there is considerable patient-to-patient variation within the bounds of vital sign normality. This is not an entirely unexpected result, as external factors such as patient age, physical fitness and reason for admission will have an effect on vital signs. Given that the patients' vital signs do not overlap on the map, the Sammon map provides important qualitative evidence that inter-patient vital sign variation is *significant* enough to motivate the design of personalised data fusion models for vital sign monitoring.

5 Optimising the Parzen Windows Model

5.1 Optimising the Parzen Width

The training procedure outlined in Chapter 3 raises some important issues that are now addressed. We previously noted that the choices for the Parzen Windows width and the number of kernel centres are critical for creating a good model. However, both of these parameters have not been optimised until now. The number of cluster centres to create the model was chosen arbitrarily at 500 centres, and the kernel width, h, was chosen on general principles outlined by Bishop [7], rather than considered on a case-specific basis.

The kernel width is particularly important; if h is too small, then the Parzen windows kernels do not fill the data space, and artifacts in the estimated p.d.f. begin to occur. Conversely, if h is too large, the model smooths out the p.d.f. so that its variance is larger than that of the underlying distribution of the data. Figure 14 demonstrates this effect, showing results obtained by using various values of h for a 1-D Gaussian modelled with 30 kernels.

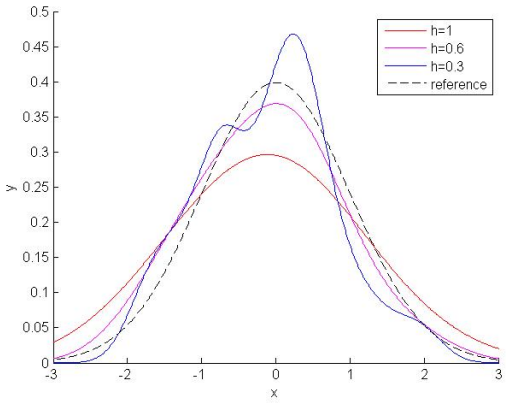

Figure 14: Parzen windows model of a zero mean, unit variance Gaussian distribution for 30 kernel centres and for various width parameters. The original distribution is shown in dashed lines.

During the training procedure, the Parzen windows width parameter was set at *h=1.49*. To maximise the predictive power of the model, it is highly desirable to find the optimum value of h. This was accomplished by conducting a thorough search, varying h over four orders of magnitude, from two orders of magnitude smaller to three orders of magnitude larger than the initial value, to investigate its effect on the resulting kernel density estimate.

The ideal Parzen Windows model trained on the training data is the one 'most similar' to the validation data set (UPMC Phase III data). This gives a natural method to assess the

optimality of h. If we define an error metric, ϵ, as the 4D integral:

$$\epsilon = \iiiint (F_{parzen}(x_1, x_2, x_3, x_4) - F_{data}(x_1, x_2, x_3, x_4))^2 dx_1 dx_2 dx_3 dx_4 \tag{9}$$

where F_{parzen} is the p.d.f. estimated by Parzen windows, F_{data} represents the underlying distribution from which the validation data is taken, and the subscripts of x represent the four different vital signs being measured (breathing rate, heart rate etc.) The error is then the integral of the squared difference between the estimated (Parzen Windows generated) p.d.f. of the training data, and the underlying p.d.f. of the validation data.

The error integral cannot be calculated directly, as F_{data} is unknown, so a computational approach must be used instead. By sampling from the validation set, we can estimate the integral of F_{data} by creating a 4D histogram, H_{data}, and calculating the hypervolume under the histogram. A similar histogram, H_{parzen}, can also be produced by sampling from the Parzen Windows model, F_{parzen}. Sampling is simple in this instance, as F_{parzen} is a sum of Gaussians with equal weightings. Using this, an approximation to the error can be calculated as:

$$\epsilon = \sum^{i \in R} \sum^{j \in R} \sum^{k \in R} \sum^{l \in R} (H_{parzen}(i, j, k, l) - H_{data}(i, j, k, l))^2 \tag{10}$$

Data is segregated into R bins for each vital sign, and H_{parzen} and H_{data} are the number of samples from the Parzen model and the validation data assigned to the bin centred at the vector (i, j, k, l). Care must be taken when sampling from the Parzen Windows model to collect a statistically significant number of points to create H_{parzen}. In fact, the curse of dimensionality means that an extensive sampling in 4D is likely to be unfeasible due to memory and computing time constraints.

The problem is avoided here by using a dimensionality reduction algorithm such as SASS or Neuroscale. Through this, it is possible to perform a limited sampling of the 4D distribution and then to transform it into a topologically similar 2D distribution. The error, ϵ, can then be calculated using histograms in the 2D visualisation space instead. This may be particularly useful as the method will remain tractable for higher dimensional distributions.

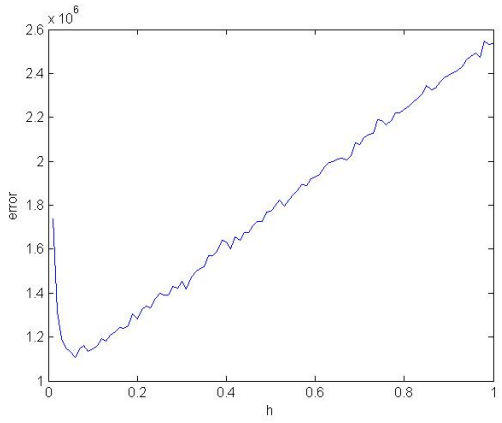

Figure 15: Plot of the sum-of-squares error between the vital sign Parzen windows model of normality and the validation set for the values of the width parameter, h, between 0 and 1

5.1.1 Results

The error between the Parzen Windows p.d.f. and the underlying p.d.f. of the validation data was calculated for values of h between 0.01 and 1000. The results are summarised in Figure 15 showing that the error is minimised at approximately $h = 0.06$, and that the error increases as h departs from this value. Between $0.1 < h < 1$, the error does not increase monotonically. This is likely to be caused by the limited size of the validation set, and the problem should become negligible if a greater number of samples are used.

5.1.2 Discussion

Although Figure 15 appears convincing, it presents an incorrect solution. Figure 16 depicts 1D slices through the width-optimised probability distribution, showing the probability of one vital sign, when the other three vital signs are fixed at their mean values. It is clear that the model is under-smoothed. This result is particularly interesting, as it indicates that it is not sufficient to minimise a global error. By visual inspection, a value of $h = 0.3$ eliminates the instances of under-smoothing in the 1D slices.

In an independent study, Hann [21] attempted to optimise the Parzen Windows widths using a different error metric, and ended up with a value of $h = 0.1$, a very similar value to that presented here. In his scheme, the squared difference between marginalised 1D models from the Parzen windows model and from the validation data model were used to calculate the error. A marginalised 1D model was produced by integrating out three of the four vital signs. This has the advantage of dealing with the p.d.f. in a more direct manner but also neglects the

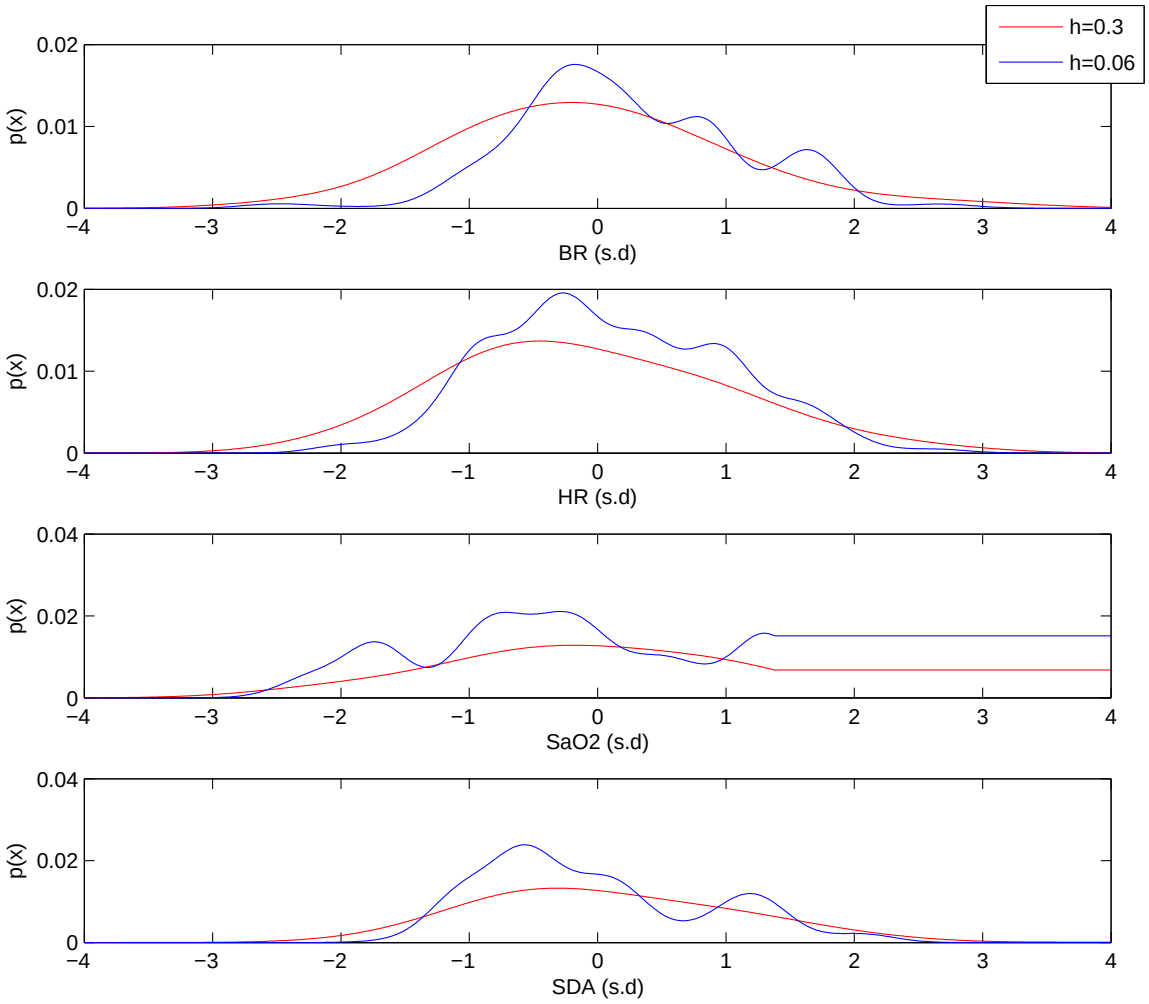

Figure 16: 1D p.d.f.s for the "optimised" Parzen Windows model ($h = 0.06$) where the parameter of interest is varied from -4 to +4, and all other parameters are kept at the mean value.

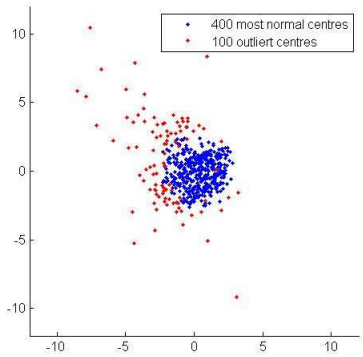

Figure 17: Neuroscale plot showing the remaining 400 'most normal' centres and the 100 outlying centres for the training (JR) data set.

effects of under-smoothing.

To calculate a better estimate of h, one could add a regularisation term to the error function to avoid selecting an overfitted solution. Alternatively, the model could be optimised by using information about critical events (C") in an error metric, so that the optimal width wass directly related to the model's ability to detect instances of patient deterioration. So far, this has not been possible as the validation set is currently not labelled with C" events.

5.2 Optimising the number of Parzen Windows Kernels

In Chapter 4 it was shown that the 500 prototype cluster centres included some points that appeared outside the main group of centres after a Neuroscale transformation. It is possible that a better model of normality can be created by removing such outliers, so that only the 400 centres closest to normality (i.e. the point [0 0 0 0]) as defined by Euclidean distance are selected [61]. The output from this procedure was processed through the Neuroscale algorithm and the result is shown in Figure 17. The 400 'most normal' centres appear to be tightly clustered, and the 100 most outlying centres have been removed from the original 500 centres. This suggests that the reduced set may provide a better model of normality.

To test this, the 400 'normal' prototype centres were used to generate a new Parzen Windows model of normality for the training (JR) data. 10^5 points were sampled from the model and compared to the validation data using the method described previously. The resulting error was $\epsilon = 1.10 \times 10^6$, which is 9×10^4 less than the error for 500 centre model, indicating that it is a slightly better model. However, we again note that alternative error metrics should be

undertaken to confirm this analysis.

In conclusion, the 400 point Parzen windows is an improvement on the original model. However, the choice of 400 centres is arbitrary and a more rigorous approach to removing outliers may yield a better model. One improvement that should be investigated is to define an outlier in terms of its local surroundings instead of using the Euclidean distance from the mean. That is, a centre should be regarded as an outlier if its nearest neighbours are 'far away'. Such a scheme would also allow outliers to be correctly identified for a multimodal distribution.

5.3 Assessment of the Model of Normality

The effectiveness of using a probabilistic model of normality to detect changes in patient vital signs has so far been justified primarily by clinical results. These results show that the model is able to detect specific instances of patient deterioration earlier than nursing staff. For example, Hravnak et al. state that in their trial *"All MET activation events were detected by BSI in advance (mean, 6.3 hours)"* [26]. Despite these excellent clinical results, the training methodology, and in particular the K-means and Parzen windows procedures that were introduced in Chapter 3, have not been analysed critically.

In a simplistic sense, the Parzen Windows method creates a smoothed and continuous high dimensional histogram of the training data. Therefore, to create a valid Parzen Windows model, data must be drawn from the underlying training set distribution. However, during the training procedure described in Chapter 3, the training set was too large to apply Parzen windows directly and instead a downsampled 'prototype' subset was generated using the K-means algorithm. This subset was assumed to be a good representation of the training set probability distribution. We now consider whether this was a valid assuption.

To test the assumption, 10,000 samples were selected from a 2D Gaussian distribution and from these, 50 cluster centres were selected using the K-means algorithm. The cluster centres were saved, and the process was then repeated 80 times, leading to the result in Figure 18.

The cluster centres are shown in red, and plotted alongside samples taken directly from the underlying distribution in blue. The figure shows that the K-means centres do not represent the underlying data distribution well, firstly appearing to produce a 'annular pattern' effect, and secondly and most importantly, showing a larger number of clusters centres in the distribution

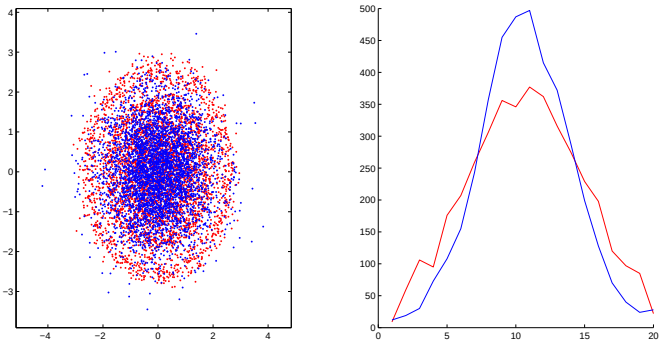

Figure 18: The result of a K-means clustering on a 2D equivariant Gaussian with $k = 50$, repeated 80 times, coloured in red. The points in blue show 500 points selected at random from the 10,000 points in the Gaussian test set.

tails than would be expected if they were selected directly from the distribution.

The reason for the 'annular pattern' effect can be established if one considers some simple cases where there are very few clusters. Consider an example where data is selected from a uniform circular distribution. Applying K-means when $k = 2$, it is clear that the optimal solution to the clustering problem will result in the two cluster centres being equidistant from the centroid. When the trial is repeated many times, a set of points that lie roughly in an annular region will be formed because of the rotational symmetry in the underlying distribution. Similarly, when $k > 2$, it can also be shown that there are a limited number of optimal cluster centre configurations that also give rise to annular patterns for repeated trials.

The cluster centre bias towards less populated areas of a data distribution is of greater importance and can also be confirmed as a genuine effect by considering a 1D case with many samples centred around $x = 0$, and a few outlying points at $x = 10$. Then the K-means representation with $k = 2$ gives the outlying data a large influence on the final cluster positions (see figure 19). It should be emphasised that this is not a problem with K-means per se (in fact, the outlying cluster may be significant), but rather shows that the K-means algorithm is not appropriate for selecting the representative subset used for the Parzen windows model.

Because the prototype subset used for the Parzen windows method is weighted towards outliers, the original training method does not accurately model the underlying data distribution, this has been partly corrected for by removing the most outlying centres (Section 5.2). However it has already been noted the method of choosing 400 centres was not entirely rigourous, and so an alternative solution is analysed in the following section.

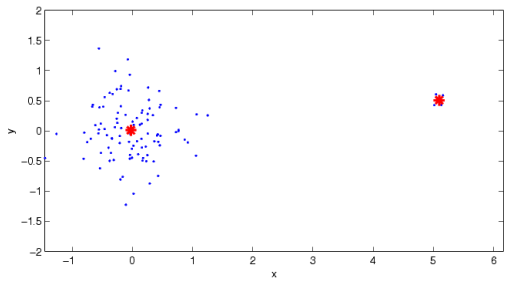

Figure 19: The result of K-means on a synthetic data set, demonstrating that outlying data can have a strong effect on the final positions of cluster centres. The final positions of the cluster centres are denoted by red stars, and the data points are in blue.

5.4 Weighted Parzen Windows

Weighted Parzen Windows (wPw) was introduced by Babich and Camps[3] to deal with the problem of the significant processing time and data storage needed to compute a kernel density estimate as a training data set gets large. Their approach is similar to the training process described so far, in that training data is clustered to reduce the size of the set. However, Babich and Camps extend this idea in an intuitive way, using the population of each cluster to bias the Parzen windows estimator.

Mathematically, the weighted Parzen Windows technique approximates Parzen Windows using a set of m prototype patterns, and is given by:

$$p_m(x) = \sum_{i=1}^{m} \frac{w_i}{h} K\left(\frac{x - x_i}{h}\right) \tag{11}$$

where w_i is the i^{th} cluster weighting and is equal to the number of training samples in the i^{th} cluster, divided by the number of points in the whole training set. All other variables take the same meanings as before. In the original paper, a hierarchical clustering algorithm was used to generate the m points, but clustering techniques such as K-means may also be used.

The result of this is that low weightings are assigned to the kernels that lie in low density regions of data space, reducing that kernel's influence on the overall shape of the estimated p.d.f. and providing a more accurate estimate of the underlying data distribution function.

5.4.1 2D Tests

A simple 2D example was first constructed in order to test the suitability of the wPw method. 10,000 samples were selected from a 2D Gaussian distribution with unit equivariance and mean

at [0,0]. They were downsampled using the K-means algorithm to a subset of 50 cluster centres. The cluster membership was also stored and was then used to compute a kernel density estimate using wPw as shown in Figure 20(c). The other plots in the figure show the underlying distribution (Fig. 20(d)), the result of the standard Parzen Windows on the 50 cluster centres (Fig. 20(a)), and the result of applying Parzen Windows on 50 points selected directly from the underlying distribution (Fig. 20(b)). The Parzen width parameter was chosen emprically at *h=0.1*, the width for which the 2D p.d.f. estimate begins to fail for a standard Parzen Windows.

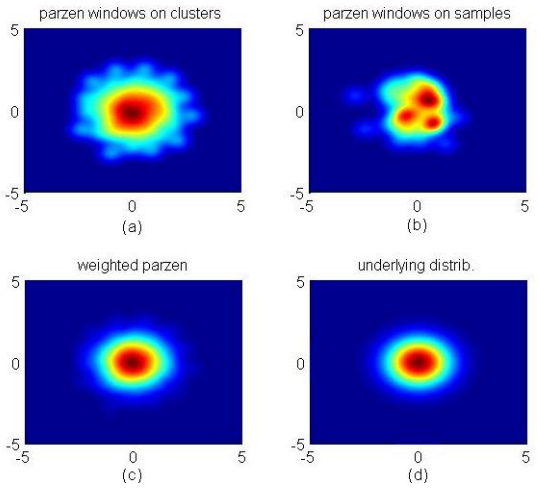

Figure 20: Kernel density estimates using 50 kernels for data derived from a 2D Gaussian distribution. (a) shows the Parzen Windows result using the Visensia training procedure (b) shows the Parzen windows result using data selected directly from the underlying distribution. (c) shows the result using wPw, and (d) shows the original distribution

The wPw technique estimates the underlying distribution well and in particular, the variances of the wPw distribution and the underlying distribution match well. By visual inspection, the wPw method and appears more accurate than the standard Parzen estimator. Furthermore, with wPw, the lower weightings towards the extremes of the distribution help to reduce the spurious pattern that can be seen in the standard technique.

In addition to this, we note that a Parzen Windows estimate produced directly by sampling the training set (Figure 20(b)), does not estimate the underlying distribution well, as the data space is too sparsely sampled. The error of the three Parzen windows models can be quantified by calculating the integrated squared error (from equation 10). The errors are shown in Table 6. It is interesting to see that the error from the sampled data model is smaller than for the clustered data model, despite the overall shape of the sampled data model being inferior.

Method	Weighted Parzen (clustered data)	Parzen (clustered data)	Parzen (sampled data)
Error	7.56×10^{-6}	1.23×10^{-4}	1.03×10^{-4}

Table 6: The mean integrated squared error for a Parzen Windows model of a 2D Gaussian with kernels generated using (a)K-means, (b) Random Sampling, and for (c) a wPw model using K-means generated kernels.

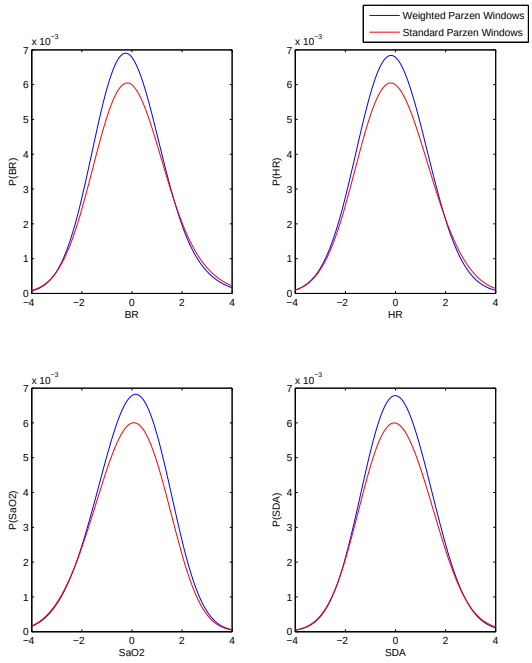

Figure 21: Unconditional probability for the case when each vital sign parameter is ramped from -4 s.d. to +4 s.d. in turn, while fixing the remaining three vital signs at their normalised mean value of zero. The weighted Parzen Windows estimate is shown in blue, and standard Parzen Windows is shown in red.

This matches with observations from earlier in this Chapter, where the minimum error did not correspond to the best fitting model.

5.4.2 Training Data tests

Having shown that wPw outperforms the standard Parzen windows estimator in the 2D case, we apply the wPw technique to the 4D JR training data set. The original parameters of Parzen width of *h=1,* and 500 prototype centres were chosen, so that results can be compared to the original model. The figure presented below shows 1D slices from the wPw model in blue, compared to the original Parzen model in red. As before, one variable is ramped between -4 and +4 s.d. with all other variables set to the mean value of zero.

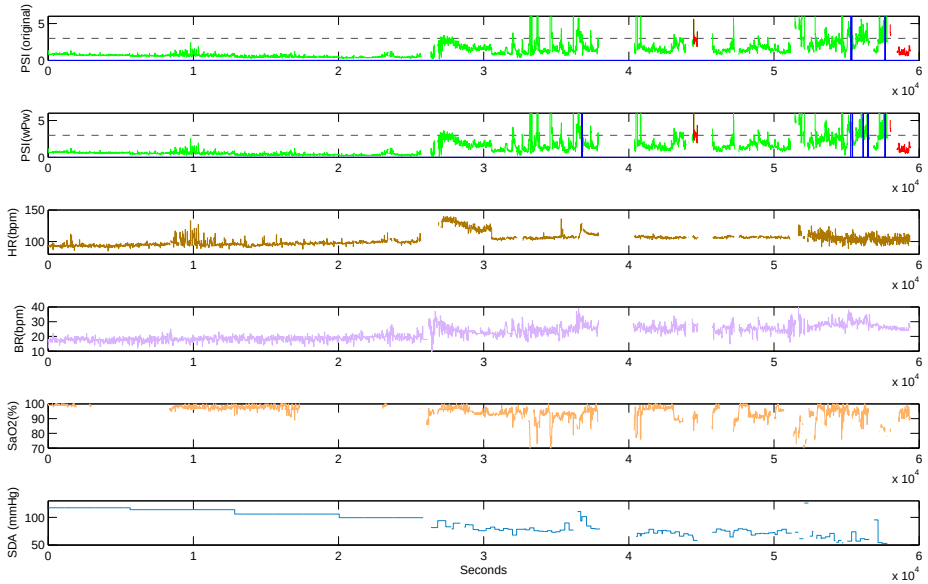

Figure 22: Summary of novelty and vital signs for Patient 328, 16 hours before MET activation. The 3.0 threshold is plotted in blue, and C" are highlighted in red on the PSI plots. Instances that would have produced a Visensia alarm are marked by vertical dashed lines.

From these, one first notes that wPw emphasises probabilities near the centre of the distribution, as this is where the most populated clusters are situated, corresponding to 'normal' vital signs. Using the error metric established in the previous chapter, it can be confirmed that the wPw-generated kernel density function provides a better estimate than standard Parzen windows, with a sum-of-squared error of 8.8×10^5 (compared to 1.1×10^6 in Figure 15). Despite this, the model's capability to detect patient deterioration may not improve much. This is because critical clinical events only correspond to the low probability regions in the model, where both the wPw and standard Parzen Windows estimates are very similar.

5.4.3 UPMC Phase I example

The wPw method was assessed using the Phase I data for a number of patient records with clinically adverse C" events. For brevity, only one record, Patient 328, is analysed here.

Figure 22 shows vital sign recordings for Patient 328 during the last 16 hours before MET (Medical Emergency Team) intervention. The index of patient normality was calculated using standard Parzen Windows and wPw are also depicted, and the two C" events that occured are highlighted in red. In the run-up to these events, long-term deterioration is readily observable. The heart rate and breathing rate slowly increase as the blood pressure falls. This gradual

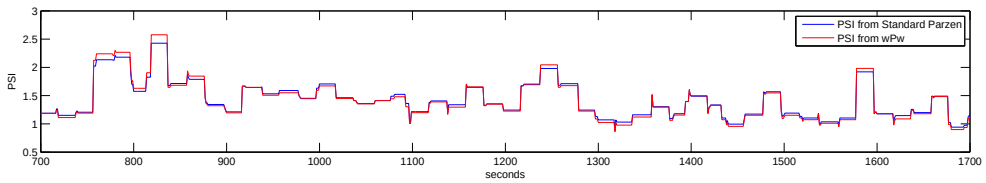

Figure 23: PSI for 1000 seconds of data from patient 328, using the weighted and standard Parzen windows

deterioration is reflected in the patient normality scores generated from the original model and the wPw model, which increase and become more erratic as time progresses.

The initial C" event was caused by a drop in diastolic blood pressure, and the event lasted for four minutes. Prior to this, the Patient Status Index (PSI) crosses the 3.0 alert threshold numerous times. However, in many of these cases the peaks in PSI are transitory and in this instance the original novelty detection system does not alarm. The wPw model would fare slightly better, alerting a clinician two hours prior to the initial event.

Following this, the blood pressure stabilises and the second C" event does not occur for another three hours. During this time, the breathing rate continues to increase, causing an elevation in PSI such that in the final hour of the patient record, the PSI for both models remains greater than the alert threshold for a significant proportion of time. In this case, both models would raise an alarm over an hour before the second C" event was recognised.

Both the original and wPw models give very similar results, and a more detailed inspection is required to reveal the differences between the two models. Figure 23 shows a much shorter section (1000 seconds) of PSI for Patient 328. The output from both models are plotted on the same axis. When the vital signs are 'normal', the wPw model generally gives a lower PSI than the original model, and gives a higher PSI than the original model when vital signs are unusual. In most cases this is unlikely to affect the generation of alerts. However, it has been shown that a borderline case (the first C" event in Figure 22) may benefit from the wPw model. It is still not clear whether using the wPw will improve the system's overall performance, and thorough testing using ROC analysis may provide a clearer assessment.

6 Conclusion and Future Work

The work described in this report provides much of the background required for future investigations. Firstly, the Visensia model of normality has been trained using the JR data. Methods to help visualise the vital sign data in 2D were then studied in Chapter 4, and a novel technique based on Sammon Maps has been proposed that allows visualisation of very large data sets. In Chapter 5 possible improvements to the probabilistic model of normality were highlighted and in response, the width and variance parameters in the Parzen Windows model were optimized using a validation data set from Phase III of the UPMC trial according to a least squares error calculated in a 2D visualization space.

Finally, an alternative method for modelling the vital sign data, weighted Parzen Windows, was implemented and was visually assessed for synthetic 2D data. The method was also applied to the JR training data, where distinct changes in the shape of the 4D vital sign p.d.f were apparent. The effectiveness of each modification to the original Visensia system was briefly tested using an example from Phase I of the UPMC trial.

In the future, three important extensions to the current system will be pursued. Firstly, the feasibility of patient-specific vital sign models will be investigated. Secondly, the continuous monitoring of a patient's state of consciousness will be developed, with a view to adding an extra variable to the current model of normality to improve its effectiveness as an early warning device. Finally, the system will be directly compared to current MET (including MEWS based) calling criteria.

As well as the work on probabilistic models of normality, the SASS visualisation technique will be extended by improving the method for visualising outlying data, and this will be tested on even larger data sets. A fuller description of these projects is provided below.

6.1 Patient-specific models of normality

Despite the means and variances of the UPMC Phase III vital signs matching very closely to the corresponding results in the JR data set, it is not unusual to observe large patient-to-patient variation. This was discussed in Chapter 4, and Figure 13 provided compelling evidence to support this notion. Furthermore, clinical trials have shown that the most effective Early Warning scores for detecting patient deterioration incorporated patient-specific information

such as age [55]. Because of this, it is highly desirable to create a personalised model where the alert threshold is not fixed, but is dependent on an individual's condition and medical history upon admission to hospital.

Initially, this will involve refining the model of normality and extending the work introduced in Chapter 5.3. In particular, a number of alternative probabilistic models will be considered, including Gaussian Mixture Models (GMM) and Support Vector Machines (SVM). GMMs are a density estimation tool that use a sum of Gaussians to estimate the underlying p.d.f from which the data has been drawn. Parzen Windows are in fact a particular form of GMM where each data point is modelled as an individual Gaussian. SVMs are a supervised learning method originally used to create an optimum classification boundary between the two data sets, but they have since been adapted for outlier detection. The method offers a different modelling paradigm that does not require a density estimate, considering only data at the edges of normality.

Numerous approaches of training a patient-specific model in real time will also be investigated, including neural networks as proposed by Zhang [66]. Another possible option is to use Bayesian Extreme Value Theory, which would allow the alert threshold on the model of normality to be set automatically, and was first introduced by Clifton et al. for monijet engines [11]. In such schemes, there are a number of obvious difficulties that must be considered. For instance, one must ensure that any patient-specific model does not adapt to long term trends that may indicate slow deterioration. The effectiveness of such a model will thus be highly dependent on the length of the training period. Zhang suggests a span of 8 hours is sufficient, but our investigations will consider whether a shorter training period is acceptable.

6.2 Investigating the relationship between EEG and GCS

In the Modified Early Warning Score described in Chapter 2, the scores are based on six vital signs, including a score of mental state such as the Glasgow Coma Scale (GCS). The GCS is a scale that ranges from a score of 3 to 15 and qualitatively describes the level of consciousness of a patient by testing the patient's visual, verbal and motor responses (see table 7). However, the score can only be assessed when a healthcare professional is present to conduct the tests.

The importance of the GCS was highlighted during the UPMC Phase I trials, where all ten critical events were correctly assessed by the early warning system, apart from one instance

	1	2	3	4	5	6
Eyes	Does not open eyes	Opens eyes in response to painful stimuli	Opens eyes in response to voice	Opens eyes sponta-neously	N/A	N/A
Verbal	Makes no sounds	Incompre-hensible sounds	Utter inap-propriate words	Confused, disorien-tated	Oriented, converses normally	N/A
Motor	Makes no movements	Extension to painful stimuli	Abnormal flexion to painful stimuli	Flexion / withdrawal to painful stimuli	Localizes painful stimuli	Obeys commands

Table 7: Criteria for assessing the Glasgow Coma Score of a patient

of patient self diagnosis, and two instances where an MET was called due to abnormalities in patient mental function. In light of this, it is highly desirable to find a method of automatically assessing patient consciousness.

The electroencephalogram (EEG) may provide a possible solution. The EEG is the measurement of electrical activity produced by the brain as recorded from electrodes placed on the scalp. It is well known that EEG signals are correlated with the states of consciousness. In particular, the frequency content of an EEG changes during the different phases of sleep [59].

A trial is currently being conducted at the John Radcliffe hospital, Oxford, that will collect continuous measurements of vital signs and EEG as well as occasional GCS scores. The study will enrol 1000 patients in Majors, Minors, Resuscitation, and the Clinical Decision Unit areas of the JR Emergency Department (ED). Electroencephalogram (EEG) data for each patient will be recorded continuously between the left and right mastoid, while the five other vital signs will be automatically collected using an industry-standard Philips bedside monitor.

During the trial, patients will continue to receive standard care. As part of this, the values of the vital signs (including GCS) will be recorded on paper at regular intervals by the nursing staff. The physical records will then be converted into an electronic format for comparing with the automatically collected data. By doing this, it is hoped that the relationship between GCS and EEG can be discovered explicitly,to determine whether segment-based analysis of the EEG data can be used as a proxy for GCS.

Work in a similar field has been undertaken by Gill et al. [19], who attempt to find a correlation between the Bispectral index (BIS) and GCS. The BIS is a proprietary algorithm that

uses the bispectrum of the EEG to assess the depth of consciousness (DOC) under anaesthesia on a scale of 0 to 100. The bispectrum is used to highlight phase relations between a signal's frequency components and is typically calculated on a time series split into L epochs as:

$$B(f_1, f_2) = \left\| \sum_{i=1}^{L} X_i(f_1) X_i(f_2) X_i^*(f_1 + f_2) \right\| \tag{12}$$

where $*$ indicates the conjugate operator, and $X_i(f)$ is the Fourier transform of the i^{th} epoch. The work suggested here differs because the EEG data will be processed directly. In doing this, we do not assume that DOC due to anaesthesia has the same effect on EEG as DOC due to other causes (such as head trauma). Furthermore, we are mainly interested in the *change* in DOC and whether deterioration can be detected (rather than creating an abolute scale of DOC.) We will investigate whether changes in the EEG in the frequency, time-frequency, or bi-spectral domain can allow changes in consciousness to be automatically assessed.

Providing that there is a positive outcome to the EEG investigation, the EEG will then be included as an extra parameter in the model of normality with the aim of increasing the sensitivity and specificity of the vital sign early warning system in predicting care escalation (i.e. prompting clinical review, referral to the Intensive Therapy Unit, cardiac arrest, or death, etc.) of ED patients with head trauma.

6.3 Timeline for Completion of Research

The projects listed above will incorporated into a structured scheme of research. Firstly, the model of normality will be retrained using the vital sign data (excluding EEG and GCS) from the 1000 patients taking part in the new ED study. This will allow us to compare results with the current model trained on the original JR training set. In particular, we will investigate whether the temperature records are suitably robust for re-inclusion to the model. The method used to create the probabilistic model will also be reconsidered as described in Section 6.1.

During the retraining process, the SASS algorithm will be used for visualising the new vital sign data set, which is likely to be over three times as large as the UPMC Phase I data set. In Chapter 4, we noted that the selection of the SASS subset is critically important for good visualisation, and a successful method to select the subset was described. However, other methods may be useful in specific instances. For instance, one may hypothesise that by biasing

the subset so that outliers are highly connected, outlying data will be visualised more accurately at the expense of slightly poorer visualisation for 'normal' data. Furthermore, it may also be possible to extend SASS to arbitrarily large data sets. An idea that will be investigated is to use a set of 'skeleton' points that are fixed in the 2D visualisation space. The 'skeleton' points contribute towards the SASS error metric, so it is hoped that a large data set can be split into a number of smaller batches that can be visualised with respect to the fixed 'skeleton'.

Once the model has been retrained, we will investigate how the alarm threshold is set. Currently, an alarm is sounded when the novelty score exceeds a threshold of 3.0 for 240 out of 300 seconds. This procedure was discovered heuristically, and has been shown to be adequate in practice. However, we believe that a more rigourous approach to this problem may improve detection of critical clinical events. One promising technique we shall investigate is called Extreme Value Theory (EVT)[14]. EVT is a branch of statistics dealing with extreme deviations from the median of a probability distribution, and is commonly used to predict the probability of unlikely events.

The natural extension of this is to use Bayesian EVT for personalised vital sign models, as described in Section 6.1. During this stage, a sixth parameter based on EEG will be introduced, that will have been derived using data from the the ED project as described in Section 6.2. It is appropriate to include EEG information at this stage in the project, as the EEG signal varies between patients depending on factors such as age, and previous medical conditions.

Finally, the new novelty detection system will be evaluated. In related work, the Visensia novelty system was assessed using ROC analysis [21]. A true positive was recorded when a critical clinical (C") event was correctly detected within a 5 minute window, and a true negative was recorded when a patient did not suffer a C" event AND no Visensia alerts were generated during the patient's entire stay on the ward. The inconsistency between how positives are recorded (on an event-basis), and how negatives are recorded (on a per-patient basis) undermines the true effectiveness of the system, and will be reconsidered when the new model is evaluated. Successful completeion of these tasks will allow comparison to current systems (Visensia), and data from the current JR trial will also provide opportunities to compare probabilistic models against current Critical Care Outreach (CCO) calling criteria. The proposed research is summarised in the Gantt chart in Figure 24.

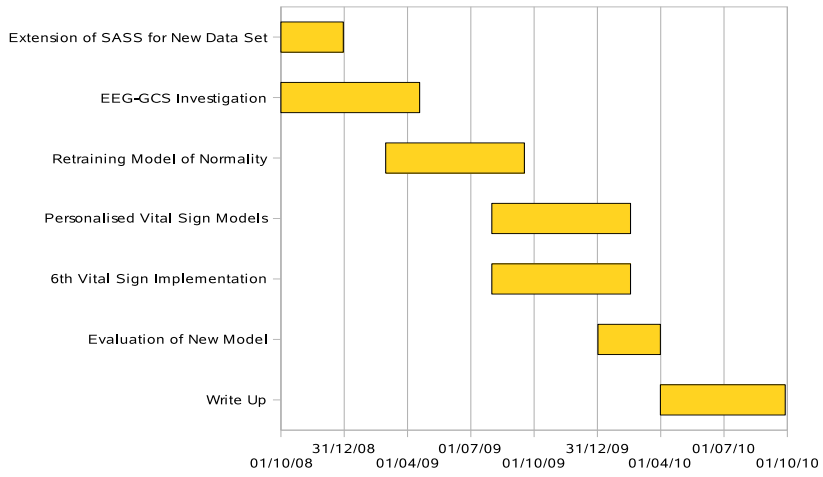

Figure 24: 24-month timeline for completion of research and write-up

A Glossary

Randomised Control Trial : A Randomised Control Trial is a methodology commonly used it testing the efficacy of health technologies and is considered a reliable way of eliminating spurious causality and bias. During the trial, different interventions are randomly administered so that known and unknown extraneous variables are distributed between the groups. Typically, one group will be the *control*, who do not receive the treatment under study and give researchers a useful comparison.

Sensitivity and Specificity : Sensitivity and specificity are statistical measures of the performance of a binary classification test. The sensitivity measures the proportion of actual positives which are correctly identified (i.e. the percentage of patient alarms that correspond to real critical events); and the specificity measures the proportion of negatives which are correctly identified (i.e. the percentage of well people who are identified as not having the condition). Formally, they are defined by:

$$sensitivity = \frac{number\ of\ true\ positives}{number\ of\ true\ positives + number\ of\ false\ negatives}$$

$$specificity = \frac{number\ of\ true\ negatives}{number\ of\ true\ negatives + number\ of\ false\ positives}$$

The positive predictive value is the proportion of patients with positive test results who actually suffer from the condition being tested for. However, one must be aware that it is proportional to the prevalence of the condition.

$$PPV = \frac{number\ of\ true\ positives}{number\ of\ true\ positives + number\ of\ false\ positives}$$

B Conferences Attended

NCAF Summer Meeting (18-19 June 2008, University of Oxford) Theme: Signal Processing and Biomedical Applications. NCAF stands for Natural Computing Algorithms Forum

ICML Workshop on Machine Learning in Healthcare Applications (July 9 2008, Helsinki, Finland). ICML stands for International Conference of machine learning. The article 'High Dimensional Visualisation for Very Large Data Sets' that demonstrated the SASS algorithm was accepted for a poster presentation

References

[1] Comprehensive Critical Care: a Review of Adult Critical Services (2000). *Department of Health. London: Department of Health.*

[2] Norm Aleks, Stuart Russell, Michael Madden, Kristan Staudenmayer, Mitchell Cohen, Diane Morabito, and Geoffrey Manley. Probabilistic modeling of sensor artifacts in critical care. In *ICML workshop on machine learning for health-care applications*, 2008.

[3] G.A. Babich and O.I. Camps. Weighted Parzen Windows for Pattern Classification. *IEEE trans. Pattern Analysis and Machine Intelligence*, pages 567–570, 1996.

[4] S. E. Bedell, D. C. Deitz, D. Leeman, and T. L. Delbanco. Incidence and characteristics of preventable iatrogenic cardiac arrests. *JAMA : the journal of the American Medical Association*, 265(21):2815–2820, June 1991.

[5] R. Bellomo, D. Goldsmith, S. Uchino, J. Buckmaster, G. K. Hart, H. Opdam, W. Silvester, L. Doolan, and G. Gutteridge. A prospective before-and-after trial of a medical emergency team. *Med J Aust*, 179(6):283–287, September 2003.

[6] D. Bennett and J. Bion. Abc of intensive care: organisation of intensive care. *BMJ (Clinical research ed.)*, 318(7196):1468–1470, May 1999.

[7] CM Bishop. Novelty detection and neural network validation. *Vision, Image and Signal Processing, IEE Proceedings-*, 141(4):217–222, 1994.

[8] O. Blatchford and S. Capewell. Emergency medical admissions: taking stock and planning for winter. *BMJ*, 315(7119):1322–3, 1997.

[9] M. D. Buist, E. Jarmolowski, P. R. Burton, S. A. Bernard, B. P. Waxman, and J. Anderson. Recognising clinical instability in hospital patients before cardiac arrest or unplanned admission to intensive care. a pilot study in a tertiary-care hospital. *Med J Aust*, 171(1):22–25, July 1999.

[10] S. Charbonnier and S. Gentil. A trend-based alarm system to improve patient monitoring in intensive care units. *Control Engineering Practice*, 15(9):1039–1050, September 2007.

[11] D.A. Clifton, L. Tarassenko, N. McGrogan, D. King, S. King, and P. Anuzis. Bayesian Extreme Value Statistics for Novelty Detection in Gas-Turbine Engines. In *Aerospace Conference, 2008 IEEE*, pages 1–11, 2008.

[12] S. Coad and S. Haines. Supporting staff caring for critically ill patients in acute care areas. *Nursing in Crit Care*, 4(5):245–248, 1999.

[13] K. Daffurn, A. Lee, KM Hillman, GF Bishop, and A. Bauman. Do nurses know when to summon emergency assistance? *Intensive Crit Care Nurs*, 10(2):115–20, 1994.

[14] L. de Haan and A. Ferreira. *Extreme Value Theory: An Introduction.* Springer Verlag, 2006.

[15] R. W. Duckitt, R. Buxton-Thomas, J. Walker, E. Cheek, V. Bewick, R. Venn, and L. G. Forni. Worthing physiological scoring system: derivation and validation of a physiological early-warning system for medical admissions. an observational, population-based single-centre study. *Br. J. Anaesth.*, 98(6):769–774, June 2007.

[16] S. Q. Duffy and D. E. Farley. Patterns of decline among inpatient procedures. *Public health reports (Washington, D.C. : 1974)*, 110(6):674–681, 1995.

[17] C. Franklin and J. Mathew. Developing strategies to prevent inhospital cardiac arrest: analyzing responses of physicians and nurses in the hours before the event. *Critical care medicine*, 22(2):244–247, February 1994.

[18] J. Gardner-Thorpe, N. Love, J. Wrightson, S. Walsh, and N. Keeling. The Value of Modified Early Warning Score (MEWS) in Surgical In-Patients: A Prospective Observational Study. *Annals of The R Coll of Surgeons England*, 88(6):571, 2006.

[19] M. Gill, S.M. Green, and B. Krauss. Can the Bispectral Index Monitor Quantify Altered Level of Consciousness in Emergency Department Patients? *Academic Emergency Medicine*, 10(2):175–179, 2003.

[20] D. R. Goldhill. The critically ill: following your mews. *QJM*, 94(10):507–510, October 2001.

[21] Alistair Hann. *Multi-Parameter Monitoring for Early Warning of Patient Deterioration.* PhD thesis, University of Oxford, 2008.

[22] K. Hillman, J. Chen, M. Cretikos, R. Bellomo, D. Brown, G. Doig, S. Finfer, and A. Flabouris. Introduction of the medical emergency team (MET) system: a cluster-randomised controlled trial. *Lancet*, 365(9477):2091–7, 2005.

[23] S. W. Hoare and P. C. W. Beatty. Automatic artifact identification in anaesthesia patient record keeping: a comparison of techniques. *Medical Engineering & Physics*, 22(8):547–553, October 2000.

[24] Timothy J. Hodgetts, Gary Kenward, Ioannis Vlackonikolis, Susan Payne, Nicolas Castle, Robert Crouch, Neil Ineson, and Loua Shaikh. Incidence, location and reasons for avoidable in-hospital cardiac arrest in a district general hospital. *Resuscitation*, 54(2):115–123, August 2002.

[25] F. Hourihan, G. Bishop, K. M. Hillman, and K. Daffurn. The medical emergency team: a new strategy to identify and intervene in high-risk patients. *Clinical Intensive Care*, page 269.

[26] M. Hravnak, L. Edwards, A. Clontz, C. Valenta, M. A. Devita, and M. R. Pinsky. Defining the incidence of cardiorespiratory instability in patients in step-down units using an electronic integrated monitoring system. *Archives of internal medicine*, 168(12):1300–1308, June 2008.

[27] T. Jacques, G.A. Harrison, M.L. McLaws, and G. Kilborn. Signs of critical conditions and emergency responses (SOCCER): A model for predicting adverse events in the inpatient setting. *Resuscitation*, 69(2):175–183, 2006.

[28] J. Kause, G. Smith, D. Prytherch, M. Parr, A. Flabouris, K. Hillman, and and. A comparison of antecedents to cardiac arrests, deaths and emergency intensive care admissions in australia and new zealand, and the united kingdom–the academia study. *Resuscitation*, 62(3):275–282, September 2004.

[29] T. Kohonen. *Self-Organizing Maps*. Springer, 2001.

[30] E.M.J. Koski, A. Mäkivirta, T. Sukuvaara, and A. Kari. Frequency and reliability of alarms in the monitoring of cardiac postoperative patients. *Journal of Clinical Monitoring and Computing*, 7(2):129–133, 1990.

[31] S. T. Lawless. Crying wolf: false alarms in a pediatric intensive care unit. *Critical care medicine*, 22(6):981–985, June 1994.

[32] RCT Lee, JR Slagle, and H. Blum. A Triangulation Method for the Sequential Mapping of Points from N-Space to Two-Space. *IEEE Trans. on Computers*, 100(26):288–292, 1977.

[33] D. Lowe and ME Tipping. NeuroScale: Novel Topographic Feature Extraction using RBF Networks. *Advances in Neural Information Processing Systems*, pages 543–549, 1997.

[34] J.B. MacQueen and Western Management Science Inst UCLA. some Methods for Classification and Analysis of Multivariate Observations, 1966.

[35] A. Mäkivirta, E. Koski, A. Kari, and T. Sukuvaara. The median filter as a preprocessor for a patient monitor limit alarm system in intensive care. *Computer methods and programs in biomedicine*, 34(2-3):139–144, 1991.

[36] H. McGloin, S. K. Adam, and M. Singer. Unexpected deaths and referrals to intensive care of patients on general wards. are some cases potentially avoidable? *J R Coll Physicians Lond*, 33(3):255–259, 1999.

[37] JWR McIntyre and LM Stanford. Ergonomics and anaesthesia: Auditory alarm signals in the operating room. *Anaesthesia: Innovation in Management. Droh R, Erdmann W, Spintge R (Eds). New York, Springer-Verlag*, pages 81–86, 1985.

[38] C. Meredith and J. Edworthy. Are there too many alarms in the intensive care unit? An overview of the problems. *J Adv Nurs*, 21(1):15–20, 1995.

[39] Ian T. Nabney. *NETLAB: Algorithms for pattern recognition*. Advances in pattern recognition. Springer, 2002.

[40] C. Oberli, J. Urzua, C. Saez, M. Guarini, A. Ciprianio, B. Garayar, G. Lema, R. Canessa, C. Sacco, and M. Irarrazaval. An expert system for monitor alarm integration. *J Clin Monit Comput*, 15(1):29–35, January 1999.

[41] T. M. O'Carroll. Survey of alarms in an intensive therapy unit. *Anaesthesia*, 41(7):742–744, July 1986.

[42] A. Otero, P. Felix, F. Palacios, C. Perez-Gandia, and C. O. S. Sorzano. Intelligent alarms for patient supervision. In *Intelligent Signal Processing, 2007. WISP 2007. IEEE International Symposium on*, pages 1–6, 2007.

[43] E. Pekalska, D. de Ridder, R.P.W. Duin, and M.A. Kraaijveld. A new method of generalizing Sammon mapping with application to algorithm speed-up. *ASCI*, 99:15–17, 1999.

[44] A. J. Pittard. Out of our reach? assessing the impact of introducing a critical care outreach service. *Anaesthesia*, 58(9):882–885, 2003.

[45] AJ Pittard. Out of our reach? Assessing the impact of introducing a critical care outreach service. *Anaesthesia*, 58(9):882, 2003.

[46] D.R. Prytherch, G.B. Smith, P. Schmidt, P.I. Featherstone, K. Stewart, D. Knight, and B. Higgins. Calculating early warning scores: A classroom comparison of pen and paper and hand-held computer methods. *Resuscitation*, 70(2):173–178, 2006.

[47] J.W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Trans. on Computers*, 18(5):401–409, 1969.

[48] R. M. Schein, N. Hazday, M. Pena, B. H. Ruben, and C. L. Sprung. Clinical antecedents to in-hospital cardiopulmonary arrest. *Chest*, 98(6):1388–1392, December 1990.

[49] B. Schoelkopf, A.J. Smola, and K.R. Mueller. Kernel Principal Component Analysis. *Lectures notes in computer Science*, pages 583–588, 1997.

[50] Roy Schoenberg, Daniel Z. Sands, and Charles Safran. Making ICU alarms meaningful: a comparison of traditional vs. trend-based algorithms. pages 379–83, 1999.

[51] J. T. Sharpley and J. C. Holden. Introducing an early warning scoring system in a district general hospital. *Nursing in critical care*, 9(3):98–103, 2004.

[52] Wiebke Sieben and Ursula Gather. Classifying alarms in intensive care - analogy to hypothesis testing. In *AIME*, pages 130–138, 2007.

[53] Andrew F. Smith and Jeremy Wood. Can some in-hospital cardio-respiratory arrests be prevented? a prospective survey. *Resuscitation*, 37(3):133–137, June 1998.

[54] G. B. Smith and J. Nolan. Medical emergency teams and cardiac arrests in hospital. results may have been due to education of ward staff. *BMJ (Clinical research ed.)*, 324(7347), May 2002.

[55] G.B. Smith, D.R. Prytherch, P.E. Schmidt, and P.I. Featherstone. Review and performance evaluation of aggregate weighted track and trigger systems. *Resuscitation*, 77(2):170–179, 2008.

[56] GB Smith, DR Prytherch, PE Schmidt, PI Featherstone, and B. Higgins. A review, and performance evaluation, of single-parameter track and trigger systems. *Resuscitation*, 2008.

[57] C. P. Subbe, M. Kruger, P. Rutherford, and L. Gemmel. Validation of a modified early warning score in medical admissions. *QJM*, 94(10):521–526, October 2001.

[58] CP Subbe, RG Davies, E. Williams, P. Rutherford, and L. Gemmell. Effect of introducing the Modified Early Warning score on clinical outcomes, cardio-pulmonary arrests and intensive care utilisation in acute medical admissions [white star]. *Anaesthesia*, 58(8):797, 2003.

[59] K. Susmáková. Human sleep and Sleep EEG. *Measurement Science Review*, 4, 2004.

[60] Tarassenko, L., Hann, A., Young, and D. Integrated monitoring and analysis for early warning of patient deterioration. *BJA: British Journal of Anaesthesia*, 97(1):64–68, July 2006.

[61] L. Tarassenko, A. Hann, A. Patterson, E. Braithwaite, K. Davidson, V. Barber, and D. Young. BIOSIGN: multi-parameter monitoring for early warning of patient deterioration. In *Medical Applications of Signal Processing, 2005. The 3rd IEE International Seminar on (Ref. No. 2005-1119)*, pages 71–76, 2005.

[62] L. Tarassenko, N. Townsend, G. Clifford, L. Mason, J. Burton, and J. Price. Medical signal processing using the Software Monitor. *Intelligent Sensor Processing (Ref. No. 2001/050), A DERA/IEE Workshop on*, page 3, 2001.

[63] C. L. Tsien and J. C. Fackler. Poor prognosis for existing monitors in the intensive care unit. *Critical care medicine*, 25(4):614–619, April 1997.

[64] C.K.I. Williams, J. Quinn, and N. McIntosh. Factorial Switching Kalman Filters for Condition Monitoring in Neonatal Intensive Care. *Advances in Neural Information Processing Systems*, 18, 2006.

[65] Michael P. Young, Valerie J. Gooder, Karen Mcbride, Brent James, and Elliott S. Fisher. Inpatient transfers to the intensive care unit. *Journal of General Internal Medicine*, 18(2):77–83, 2003.

[66] Y. Zhang. Real-time development of patient-specific alarm algorithms for critical care. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2007:4351–4354, 2007.