A novel approach to the in egration of posterior knowledge into condition monitoring systems: theory and practice.

S. Marriott and R. F. Harrison

Research Report Number 691

Department of Automatic Control and Systems Engineering
The University of Sheffield
Mappin Street
Sheffield, S1 3JD, U.K.

Contact author: Dr. S. Marriott
E-Mail s.marriott@sheffield.ac.uk

The financial support of Rolls-Royce plc and the EPSRC (GR/L16651) is gratefully acknowledged

A novel approach to the integration of posterior knowledge into condition monitoring systems: theory and practice.

S. Marriott and R. F. Harrison

Department of Automatic Control and Systems Engineering

The University of Sheffield

Mappin Street

Sheffield, S1 3JD, U.K.

Contact author: Dr. S. Marriott

E-Mail s.marriott@sheffield.ac.uk

The financial support of Rolls-Royce plc and the EPSRC (GR/L16651) is gratefully acknowledged

Abstract

This paper considers the problem of the integration of posterior knowledge into condition monitoring systems from both the theoretical and practical points of view. The problem is framed in the context of elementary probability theory where the task of posterior knowledge representation is examined. A methodology for updating posterior probabilities is proposed for cases where fault conditions are rejected on the basis of external knowledge supplied by an end-user. Cases of exclusive, conditionally independent and dependent condition-classes are considered. A possible condition-class ranking is generated following the estimation of condition- class probability functions. It is shown that a simple renormalisation of existing probabilities does not apply in the dependent condition-class case and can lead to erroneous results; the condition-class ranking may change following the exclusion of condition-classes known *not* to have occurred. An artificial example is used to illustrate the theoretical principles. Simulated fault data are then used to explore the posterior probability estimation problem Through the use of radial basis function networks. A validated aircraft jet engine model is used which allows for the injection of faults (conditions). A simple, model-based range-checking methodology is applied to the data to provide a quick method of generating verified condition data for condition-class prediction and probability estimation. It is shown that the maximum possible accuracy can be achieved when the most probable fault is chosen in each case.

Key Words: Fault Diagnosis, Condition Monitoring, Posterior Knowledge Inclusion, Radial Basis Function Networks, Regularisation, Jet Aircraft Engines.

1. Introduction

The main aim of this work is to devise a mechanism for the inclusion of knowledge into predictive systems to allow the update of predicted probabilities generated without that knowledge. Posterior knowledge inclusion has potential applications in the field of condition monitoring or fault diagnosis and isolation as explored in this paper. Incorporation of knowledge about a monitored plant, not available to the condition monitoring system, will facilitate a more informed choice of maintenance strategy.

There is a growing interest in automated condition monitoring systems as the number and complexity of monitored plants increases to keep pace with the demands of modern technology. This interest is reflected in the number of fault detection and isolation methods appearing in the literature (e.g. Rodd, 1997; Patton and Clark, 1989). Such methods usually entail the monitoring of key system parameters—with or without a reference model—for pre-defined anomalies.

The so-called "classical" methods are based upon limit checking (Isermann, 1997) and involve the monitoring of measurable variables to detect pre-defined range violations. The monitoring system may initiate appropriate control actions immediately and alert the operator. Other systems may alert the operator only. Such systems are often simple and reliable (Isermann,1997) but may only be suitable for detecting relatively large changes. Furthermore, the detection is not dynamic in that changes in operating profile over time may indicate possible faults at a much earlier on prior to failure.

Condition monitoring systems may be model-based where a reference model is used in comparison with the real plant behaviour (e.g. Trave-Massuyes and Milne, 1997; Karsai and DeCaria,1997; Milne et al, 1996; Gomm, 1994). Other systems may involve the use of rule-bases and expert systems (e.g. Wang, Lu, and McGreavy, 1997; Bogunovic, and Mesic,1996; Keravnou and Johnson,1986; Liu, Singonahalli, and Iyer, 1996; McDonald, Burt, and Moyes, 1996; Wang Xue and Yang Shuzi, 1996). Novelty detection provides another way of detecting anomalous conditions by training an artificial neural network (or other adaptive system) to recognise normal operating modes; anomalous conditions are those which deviate from the learned regions of parameter space (Tarassenko, 1996;1997). Various types of artificial neural network have been applied to condition monitoring (e.g. Dimla, Lister and Leighton, 1997; Wilson, Irwin, and Lightbody, 1997; Boudoud and Masson, 1996; Li, Wong, and Nee, 1996; Patel, *et al*,1996; Perrott and Perryman, 1995; Zhang; Ganesan, and Sankar, 1995).

Other condition monitoring methods include chaos, (e.g. Logan and Matthew, 1996), statistical methods (e.g. Weighell, Martin, and Morris, 1997; Korbicz, and Kus, 1996; Ma Yizhong, 1996; Zhang,1996), Fourier Transforms and Wavelets (e.g. Pan, Sas. and van Brussel, 1996; MacIntyre and O'Brien, 1995), nonlinear observers (e.g. Preston, Shields, and Daley, 1996; Yang and Saif,1996; Krishnaswami and Rizzoni 1994.), Hybrid Approaches (e.g. Hines, Miller, and Hajek,1995; Eryurek, and Upadhyaya, (1995) Lianhui Chen and Ho 1994; Ding, and Wach,1994; Isermann, 1994) analytical redundancy (e.g. Dorr, et al, 1997) and evolutionary methods (e.g. Bilchev and Parmee, 1996; Korbicz, and Kus,1995)

In this paper, the classical range-checking detection method is used in the simulations for simplicity because this paper is concerned with the stage *after* condition-class data has been processed by whatever detection method. The range checking method provides adequate data for the demonstration of post-detection methods of the type explored here.

1.1 Posterior Knowledge

The emphasis of most condition monitoring systems is invariably confined to the actual tasks of detecting and isolating faults and alerting an end-user to their possible existence. These systems may or may not give probabilistic estimates of condition-class likelihoods to allow the end-user to decide a course of action. It is clear that such a methodology is "open-loop" in that the end-user is given a final analysis, upon which to base operational decisions, without having the opportunity to feed his or her objective information back into the system.

A "closed-loop" scenario is desirable because an end-user may have external information (not available to the condition monitoring system) which would alter the fault diagnosis for specific instances of the condition monitoring process. For example, the end-user may say, "The condition monitoring system indicates the possibility of faults x,y and z. I have just checked y and can discount the possibility of a fault there. How does this affect the possibility of faults x and z?".

The checking of y is not included in the monitored plant parameters and occurs after the condition monitoring system has made its predictions. This external knowledge is given the name "posterior knowledge" to distinguish it from any other knowledge about the monitored plant. Posterior knowledge is knowledge about the outcome supplied by an operator, or some other source, and which is not available to the predictive system at the time of prediction. It is new evidence about the posterior probabilities which have been predicted for the current classification in the form of an updated output classification and differs from the new evidence about the state of the system which is typically encountered in sequential decision theory (e.g. Melsa and Cohn, 1978). Posterior knowledge is deterministic; it is about known outcomes. The incorporation of this knowledge into the feedback loop of condition monitoring allows fault analysis to be adjusted towards a more accurate picture of the current plant status (Figure 1).

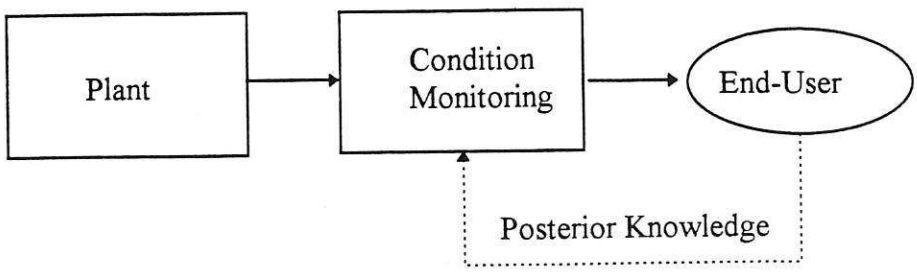


Figure 1. The condition monitoring feedback-loop. Posterior knowledge supplied by an end-user may be integrated into the condition monitoring process to improve condition-class isolation.

Where more than a single fault can occur at any one time, a ranking of possible fault scenarios can arise. This ranking is tentative because it is based upon the current values of the key plant parameters alone. The inclusion of posterior knowledge supplied by the end user may alter the fault probabilities sufficiently to alter the hierarchical structure. For example, fault x may be far more likely to be the case compared to fault z until posterior knowledge about fault y (perhaps highly correlated with x) alters the ranking by making fault z more probable. A synthetic example of this was included in (Marriott and Harrison,1997;1998) to illustrate the principle; this example is included in Section 5 for completeness.

1.2 Posterior Knowledge applied to Jet Aircraft Engine Condition Monitoring.

Two questions naturally arise from the foregoing discussion: how is posterior knowledge to be quantified and how is it to be integrated with the information contained within the condition monitoring system based upon the key plant parameters? This paper explores these two questions and considers the implicit problem of estimating and updating the probabilities associated with possible fault conditions. The key objective is to develop a method of automating the knowledge inclusion and updating process which follows logically from the fault detection and isolation tasks. The work detailed here is an exploration of these issues from first principles.

One particular application area for fault diagnosis and isolation methods is that of aircraft jet engines (e.g. Patton and Chen, 1997; Tarassenko, 1996;1997). These are complex systems comprised of distinct interacting sub-units which include electronic feedback control and monitoring devices (Rolls-Royce, 1986). The posterior knowledge inclusion problem, as considered in this paper, is discussed in the context of aircraft jet engine monitoring. The inclusion of posterior knowledge into condition monitoring systems applied to jet engines is motivated by a need to reduce costly no fault found (NFF) conditions . NFF conditions occur when one or more faults are flagged and subsequent tests of sub-units fail to locate a problem. The generation of fault rankings—capable of being updated by posterior knowledge—will allow better-informed decisions about which sub-units and/or components to remove and test.

The present work is based upon the Trent 700 engine model developed by Patel (Patel *et al*, 1996). This SIMULINK^{tm} model consists of the engine and accessories and is used to generate fault data. The accessories include the electronic engine control (EEC) to monitor engine performance and make necessary adjustments.

1.3 Paper Overview

Section 2 considers the theory underlying the representation of posterior knowledge. This leads into Section 3 where the posterior probability update equation is introduced. This equation governs the changes in posterior probabilities for a given fault determined by posterior knowledge. Section 4 looks at the resulting fault rankings and the effects of condition classes being excluded. Section 5 includes a discussion of probability estimation and the use of radial basis function networks.

Section 6 introduces the turbo-jet engine. Simulation of fault conditions and the subsequent fault detection method are both discussed in Section 7 along with some of the issues involved in fault diagnosis. Simulation results are introduced and discussed in Section 8. Finally, conclusions and further work are covered in Section 9.

1.4 Posterior Knowledge Inclusion From an Engineer's Point of View

A condition monitoring system will typically provide an end-user with a set of predictions indicating one or more possible condition-classes. Merely choosing a single condition-class, on the basis of its associated probabilities, may be too simplistic. Furthermore, the end-user's knowledge may come to bear on the problem, as posterior knowledge, and be used to modify the original condition monitoring system diagnosis. A simple example will illustrate this (Marriott and Harrison, 1998):

A gas turbine vibration monitoring system has detected several features that correspond to one of three conditions: "Bearing wear in IP shaft" with probability 0.65, "Out-of-balance in LP compressor" with probability 0.20, and "Out-of-balance in HP compressor" with probability 0.15. However, the user knows from additional knowledge that a recent change of bearing rules out condition "A". Is the most likely diagnosis now "Out of balance in LP compressor"?[1]

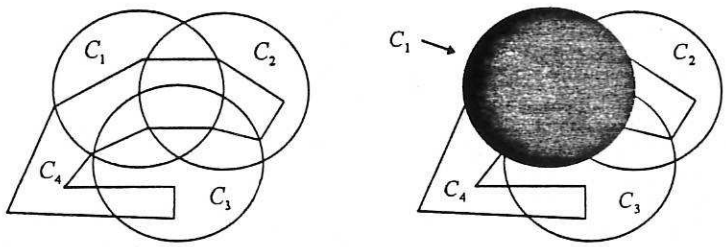
If the above conclusions are based on dependent probability distributions then it may not be sufficient simply to redistribute the probabilities between conditions "B" and "C"; this issue will be discussed further in Section 4. Indeed it is possible that the suggestion *"Out-of-balance in LP compressor"* is based on vibration phenomena attributed to bearing wear that also produces the out-of-balance as a side-effect. Eliminating bearing wear as a possible diagnosis could remove the possibility of the LP out-of-balance. The engineer may, therefore conclude, that the correct diagnosis is *"Out-of-balance in the HP compressor"*. This example illustrates some of the issues concerning the manner in which this posterior knowledge can be incorporated by the system for re-evaluation and future reference.

---

[1] Suggested by Dr. Steven King of Rolls-Royce plc

2. Posterior Knowledge Representation: Condition-Class Exclusion

As mentioned in the introduction, the first thing to be considered is how to represent the knowledge integration problem in such a way that posterior knowledge of possible system states and associated fault conditions may be incorporated. A useful framework is provided by elementary probability theory. (e.g. Durrett 1994; Grimmet and Stirzaker, 1992.). The set of possible monitored parameter values may be divided into N, possibly overlapping, condition-classes given by $U = C_1 \cup C_2,\ldots,\cup C_N$ where $C_i, i = 1,\ldots,N$ represents condition-class (fault) i. This space is assumed to be exhaustive. A four condition-class example is represented by a Venn diagram in Figure 2.



(a) (b)

Figure 2. (a) Abstract representation of a four class problem showing the maximum possible number of overlapped regions. Note that some of the possible regions of overlap may contain no members and, thus, would not exist. (b) The shaded circle represents the total probability of class 1 occurring.

Venn diagrams provide a useful way of representing the probabilities involved in updating condition-class predictions. Figure 2 (b) represents schematically the probability of class 1 faults occurring in a four class problem. The representation of probabilities by Venn diagrams can be justified by appealing to the frequentist interpretation of probability (e.g. Kneale, 1949). On analysis it is observed that condition-classes may be: *independent, dependent and exclusive*, (Appendix A1) *dependent and non-exclusive* These three distinct cases will be examined within this paper. Condition classes of the latter type will be dealt with first, being the most general—the former two are special cases. Independence in this context specifically is taken to be *conditional independence* (Bernardo and Smith, 1994; Grimmet and Stirzaker, 1992) See Appendix A7.

There are many possible ways of representing posterior knowledge. In a probabilistic context, it is useful to represent it as a set of revised condition-class probabilities, that is, a revised probability for each condition-class influenced by observations of the current situation consisting of external information; this information is then used to update the condition ranking via the updated condition-class probabilities. For example, a set of condition-class posterior probabilities will be predicted for a single input datum. If it is then possible to exclude one or more condition-classes on the basis of knowledge or reasoning not available to the predictive system, then the current list of condition-class probabilities must be revised to give a more accurate estimate of new condition-class *posterior* probabilities in the form of a ranking. Thus, the posterior knowledge is used to update the posterior probabilities of the condition-class occurrence.

Exclusion of condition-classes by posterior knowledge is the simplest case and will be dealt with here. Formally, the revised posterior probabilities require that the inclusion of external knowledge (evidence) be explicitly included in the notation e.g. $P(C_1|\varepsilon)$, where the symbol 'ε' denotes the external knowledge or evidence. Here, probabilities are required of the form $P(C_i|C_j^c)$, $P(C_i|C_j^c \cap C_k^c)$, and $P(C_i|C_j^c \cap C_k^c \cap C_l^c)$ with the general form given by $P\left(C_i|\bigcap_{k\in\Delta_\varepsilon} C_k^c\right)$ where Δ_ε denotes the set of indices of the excluded condition-classes; the exclusion being based upon external evidence. The external evidence is of the general form: $\varepsilon = \bigcap_{k\in\Delta_\varepsilon} C_k^c$ where the superscripted c indicates the complement operation with respect to the *universal set*, (Grimmet and Stirzaker, 1992) thus C_2^c and C_4^c signify that condition-classes two and four respectively have been excluded; this constitutes the new knowledge that those condition-classes are now known not to have occurred.. Note that the inclusion of posterior knowledge is given in terms of condition-classes which are known *not to have occurred* as indicated by the external knowledge. It is convenient to represent the updated posterior probabilities in terms of probabilities estimated from previous observations of system conditions, i.e. condition-classes which have occurred; these probabilities we call *probabilities of occurrence* and they can be estimated from empirical data. A more general form of ε is possible, were posterior knowledge to be represented as revised subjective or objective probabilities, but this paper is confined to the specific case of posterior knowledge as condition-class exclusions.

For a three condition-class problem, the probability of condition-class 1 occurring given the *posterior* information that condition-class 2 has not occurred is denoted by $P(C_1|C_2^c) = \frac{P(C_1 \cap C_2^c)}{P(C_2^c)} = \frac{P(C_1 \cup C_2) - P(C_2)}{P(C_1 \cup C_2 \cup C_3) - P(C_2)}$, using the definition of conditional probability (Appendix A2) for discrete events (e.g. Durrett, 1994, Grimmet and Stirzaker, 1992),. This situation is shown schematically in Figure 3. For the general case, where a set of dependent condition-classes is excluded, the following notation is introduced:, Δ_r and Δ_ε denote the index sets of included and excluded sets respectively where $\Delta_r = \{\delta_1, \delta_2, \ldots, \delta_{Nr}\}$ and $\Delta_\varepsilon = \{\delta_{Nr+1}, \delta_{Nr+2}, \ldots, \delta_N\}$ N_r is the number of included condition-classes, N is the total number of possible condition-classes. The delta notation is used to denote that the condition-class indices are not necessarily selected on the basis of ordering e.g. it could be that for a five condition-class problem, $\Delta_r = \{1,3,5\}$ and $\Delta_\varepsilon = \{2,4\}$ in which case $\delta_1 = 1$, $\delta_2 = 3$, $\delta_3 = 5$, $\delta_4 = 2$ and $\delta_5 = 4$ where condition-classes two and four have been excluded.

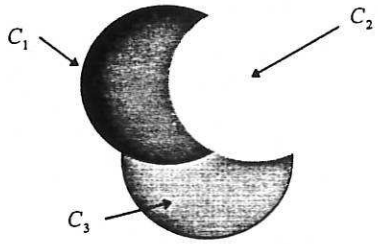


Figure 3. The diagrammatic representation of $P(C_1|C_2^c)$ for three dependent condition-classes. $P(C_1|C_2^c)$ is the probability of the remainder of C_1 (without C_2) divided by the probability of C_1 and C_3 combined (without C_2).

3. Posterior Probability Update Equation

Now that posterior knowledge has been quantified within a probabilistic framework, the second main question raised in the introduction can be explored. How are the posterior probabilities of the occurrence of condition-classes generated by a condition monitoring system to be updated using posterior knowledge? A general method of converting knowledge about excluded condition-classes in terms of estimable probabilities of occurrence is required. This general method can then be automated to provide an end-user with revised posterior probabilities given the posterior knowledge.

3.1 The General Update Equation for Excluded condition-Class Posterior Knowledge

In general, to calculate the updated probabilities, given *posterior* knowledge,

$$P\left(C_{\delta_i}|C^c_{\delta_{N+1}}\cap C^c_{\delta_{N+2}}\cap\ldots\cap C^c_{\delta_N}\right)=\frac{P\left(C_{\delta_i}\cap C^c_{\delta_{N+1}}\cap C^c_{\delta_{N+2}}\cap\ldots\cap C^c_{\delta_N}\right)}{P\left(C^c_{\delta_{N+1}}\cap C^c_{\delta_{N+2}}\cap\ldots\cap C^c_{\delta_N}\right)}=\frac{P\left(\bigcup_j C_{\delta_j}\right)-P\left(\bigcup_k C_{\delta_k}\right)}{P\left(\bigcup_l C_{\delta_l}\right)-P\left(\bigcup_k C_{\delta_k}\right)} \qquad (1)$$

where δ_i is the i th index, $\delta_i \in \{1,2,\ldots,N\}$, $j \in \{\delta_i\}\cup\Delta_\varepsilon$, $k\in\Delta_\varepsilon$, and $l\in\Delta_r\cup\Delta_\varepsilon$. This expression does not yet include the conditional dependence of the condition-class probabilities upon the monitored parameter vector. Introducing this definition, eqn (1) is replaced by

$$P\left(C_{\delta_i}\Big|\bigcap_k C^c_{\delta_k}\cap\mathbf{x}\right)=\frac{P\left(\left(\bigcup_j C_{\delta_j}\right)\Big|\mathbf{x}\right)-P\left(\left(\bigcup_k C_{\delta_k}\right)\Big|\mathbf{x}\right)}{P\left(\left(\bigcup_l C_{\delta_l}\right)\Big|\mathbf{x}\right)-P\left(\left(\bigcup_k C_{\delta_k}\right)\Big|\mathbf{x}\right)} \qquad (2)$$

This compact expression for the update of condition-class probabilities following the inclusion of posterior knowledge is derived in Appendix A8. Equation (2) (Marriott and Harrison,1997,1998) is a generalisation of the ideas illustrated in Figure 3. The union of the posterior probabilities of excluded condition-classes is subtracted from the remaining (non-excluded) condition-classes. Both the numerator and denominator of equation (2) represent what remains when any possibility of the excluded condition-classes is removed. Thus, the new probability of $P\left(C_{\delta_i}|\mathbf{x}\right)$ given by Equation (2) is based upon the reduced set of possible condition-classes. Note that $\bigcup_I C_{\delta_i} = U$ (the universal set) under the assumption of exaustivity.

Equation (2) represents and formalises the intuitive notion that the subsequent probabilities are conditioned on the known non-occurrence of a given fault. For example, if condition-class 2 has definitely not occurred by observation then joint events involving condition-class 2 (for example, the joint event condition-class1 and condition-class 2) can not have occurred.

3.2 An Example

An example of the application of Equation 2 to a four condition-class problem is shown in Figure 5.

$$P\left(C_1|C_2^c \cap C_4^c \cap \mathbf{x}\right) = \frac{P\left(C_1 \cap C_2^c \cap C_4^c \cap \mathbf{x}\right)}{P\left(C_2^c \cap C_4^c\right)}$$

$$= \frac{P\left(C_1 \cup C_2 \cup C_4|\mathbf{x}\right) - P\left(C_2 \cup C_4|\mathbf{x}\right)}{P\left(C_1 \cup C_2 \cup C_3 \cup C_4|\mathbf{x}\right) - P\left(C_2 \cup C_4|\mathbf{x}\right)}$$

$$= \frac{\begin{array}{l} P(C_1|\mathbf{x}) - P(C_1 \cap C_2|\mathbf{x}) \\ -P(C_1 \cap C_4|\mathbf{x}) + P(C_1 \cap C_2 \cap C_4|\mathbf{x}) \end{array}}{\begin{array}{l} P(C_1|\mathbf{x}) + P(C_3|\mathbf{x}) \\ -P(C_1 \cap C_2|\mathbf{x}) - P(C_1 \cap C_3|\mathbf{x}) - P(C_1 \cap C_4|\mathbf{x}) \\ -P(C_2 \cap C_3|\mathbf{x}) - P(C_3 \cap C_4|\mathbf{x}) \\ +P(C_1 \cap C_2 \cap C_3|\mathbf{x}) + P(C_1 \cap C_2 \cap C_4|\mathbf{x}) \\ +P(C_1 \cap C_3 \cap C_4|\mathbf{x}) + P(C_2 \cap C_3 \cap C_4|\mathbf{x}) \\ -P(C_1 \cap C_2 \cap C_3 \cap C_4|\mathbf{x}) \end{array}}$$

Figure 4. A representation of $P\left(C_1 | C_2^c \cap C_4^c\right)$ where condition-classes 2 and 4 have been excluded by posterior knowledge. The probability of condition-class 1 occurring, either alone or in conjunction with condition-class 3 is conditioned upon the remaining possible events with known probabilities.

3.3 A Taxonomy of Condition-Classes Excluded by Posterior Knowledge.

Three specific exclusion cases may be isolated from equation (2); these are, in ascending order of difficulty: Exclusive Class, Conditionally Independent Class (Appendix, A7)., and Dependent Class exclusions which reflect the previous division of condition-classes.

Three Theorems corresponding to the three cases are:

3.1 **Theorem***: Exclusive Class Renormalisation (ECR) Theorem*

For a set of exclusive condition-classes, the updated posterior probabilities of the remaining condition-classes, following the exclusion of the set, will be given by a renormalisation of the remaining probabilities.

3.2 **Theorem:** ***Independent Class Renormalisation (ICR) Theorem***

For a set of independent condition-classes, the updated posterior probabilities of the remaining condition-classes, following the exclusion of this set, will be given by a renormalisation of the remaining probabilities.

3.3 **Theorem***: The Dependent Class (DC) Theorem*

For non-exclusive and dependent condition-classes, neither the ECR Theorem nor the ICR Theorem applies. The full probability update equation (2) must be used.

All three cases have been dealt with separately by the theorems. Proofs can be found in Appendix A (9-11) It can be shown (Marriott and Harrison, 1997) that if all three cases occur in any one condition monitoring problem they can be decoupled and treated separately. Thus, a simple renormalisation is only valid in two of the three cases. This accords with intuition in that exclusion of condition-classes with dependent probability distributions will alter the position of condition-classes in the ranking. This idea will be illustrated in the next section.

The exclusive condition-class situation is shown schematically in Figure 5.

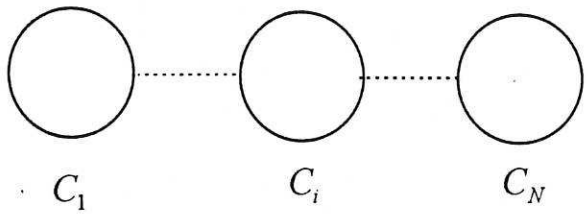


Figure 5. A set of exclusive condition-classes.

Because the condition-classes are exclusive, $P\left(C_i \cap C_j|\mathbf{x}\right)=0 \quad \forall i,j$ i.e. all probabilities of joint condition-classes are zero, only the single condition-class probabilities $P\left(C_i|\mathbf{x}\right)$ are required to calculate the condition-class union probabilities in Equation (2). This fact leads to the ECR theorem. The update equation gives rise to the following special form of Equation (2) under the conditions of exclusivity:

$$P\left(C_{\delta_i}|\bigcap_k C_{\delta_k}^c \cap \mathbf{x}\right)=\frac{P\left(C_{\delta_i}|\mathbf{x}\right)}{\sum_r P\left(C_{\delta_r}|\mathbf{x}\right)} \qquad (3)$$

for conditionally independent sets, the ICR Theorem ensures that a renormalisation of the remaining probabilities, following exclusion, is a valid operation. Thus, the update equation is of the special form

$$P\left(C_{\delta_i}\left|\bigcap_k C_{\delta_k}^c \cap \mathbf{x}\right.\right)=\frac{P\left(C_{\delta_i}\left|\mathbf{x}\right.\right)}{P\left(\bigcup_r C_{\delta_r}\left|\mathbf{x}\right.\right)}$$

where the denominator of Equation (3) has been replaced by the union which indicates that the condition-classes are not exclusive, i.e. that joint probabilities occur.

4. Condition-Class Probabilities

We assume that a statistical model of the condition-class probability distributions is available via some estimation process (e.g. neural networks, mixture models etc.). The resulting model is fixed and does not give any information about how posterior knowledge is to be incorporated. This can lead to problems in complex situations where posterior knowledge may change the relative ranking of possible condition-classes. For example, the interrogation of a fixed classifier will provide a ranking of possible condition-classes based upon the computed posterior probabilities. If the indicated conditions are *exclusive* or *conditionally independent* of all other possible conditions, then a simple renormalisation of the probabilities of the remaining condition-classes—following the exclusion of one or more condition-class on the basis of external information—is the obvious solution. Excluded *dependent* condition-classes may, however, affect the condition-class ranking owing to interactions between classes. This is illustrated in the following example (Marriott and Harrison, 1998):

4.1 The Condition-Class Ranking.

A Gaussian three class problem was specified with the posterior probabilities as shown in Figure 6 (a). The classes in this synthetic problem might represent anomalous conditions such as *"Out-of-balance in LP compressor"*. Gaussian likelihoods are specified for the occurrences of condition-classes 1,2, and 3 alone, that is where a condition-class does not occur in conjunction with any other. Gaussian Likelihoods are also specified for the joint events of classes 1 and 2 and classes 2 and 3. Priors are also specified for the classes. Using Baye's theorem gives the posterior probabilities shown in Figure 6.

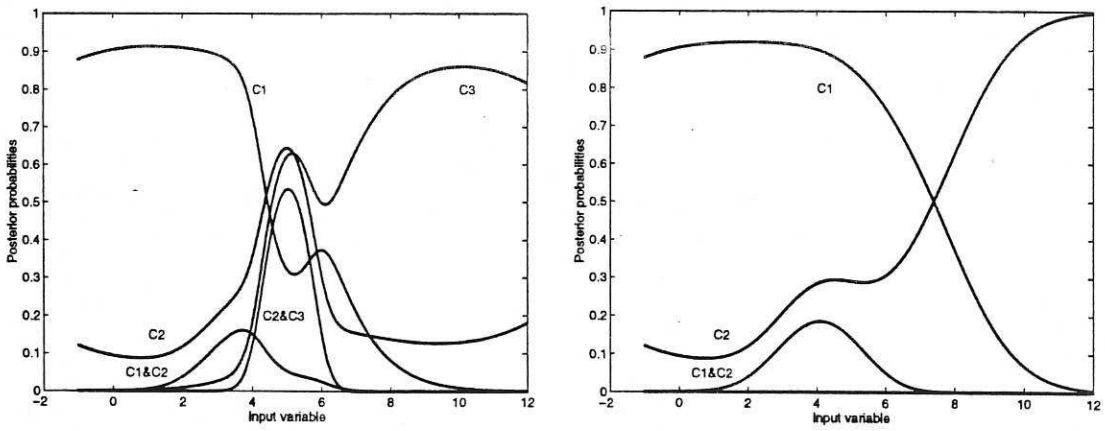


Figure 6. (a) The posterior probabilities for the three class example before exclusion of class 3. (b) the posterior probabilities following the exclusion of class 3 on the basis of posterior knowledge.

At the point $x = 5$ the posterior probabilities of fault occurrence prior to posterior knowledge are given by column 2 of Table 1 which shows the effect of external knowledge on the ranking of condition-classes. Given the *posterior* knowledge that class 3 is excluded, in this case, the updated posterior probabilities are given in column 4 of Table 1 and shown in Figure 6 (b). The exclusion of class 3 entails the removal of the likelihoods of class 3 alone and class 2 and 3. These revised probabilities have been calculated using Equation (2). Note that class 1 has risen to the top of the condition -class ranking following the inclusion of posterior knowledge into the probability adjustment process. A simple renormalisation would have placed class 2 at the top of the ranking which would have been incorrect.

The reason for the change in classification ranking following posterior knowledge is that faults C2 and C3 are very highly coupled as shown by the posterior probability of 0.5343 for the two classes occurring together. At the point $x = 5$ the exclusion of C3 reduces the probability of C2 occurring by an amount significant enough to alter the class ranking. The joint probability distribution of C2&C3 accounts for a significant proportion of class C2 occurring at $x = 5$.

Prior to External Evidence	Probability	Following External Evidence	Probability
C_2	0.6444	C_1	0.8518
C_3	0.6210	C_2	0.2907
$C_2 \cap C_3$	0.5343	$C_1 \cap C_2$	0.2014
C_1	0.3229	—	
$C_1 \cap C_2$	0.0540	—	

Table 1. The fault class ranking before and after the inclusion of external evidence.

4.2 Probability Update Procedure

A theoretical analysis of condition-class types leads naturally to a practical methodology for updating the posterior probabilities. The probability update procedure may be broken down into a series of discrete steps. From the theorems of Section 3, the following procedure for updating the posterior probabilities, may be specified.

i) Using the estimated priors, determine the *exclusive* condition-classes to be excluded on the basis of external knowledge and renormalise the remaining condition-class probabilities,

ii) From the estimated posterior probabilities, determine the *independent* condition-classes to be excluded and renormalise the remaining class probabilities,

iii) finally, use the probability update equation (equation (2)) to exclude the *non-independent* condition-classes.

Step i) is fairly straightforward because exclusive condition-classes will not occur in conjunction with any other classes. Exclusivity will be apparent from a pre-processing of the data. For example, if condition-class 1 does not occur in conjunction with any other condition-class, then it is assumed to be exclusive. Step ii) may be difficult in practice because some measure of independence across regions of input space will have to be developed.

Finding exclusive condition-classes means that the joint probabilities involving a given exclusive condition-class do not have to be estimated. At worst, there are no exclusive condition-classes and it is difficult to discern any independent classes. This means that the maximum number of probability functions have to be estimated, including all those of overlapping class regions.

Forming the set of all condition-classes $U = \{C_1,\ldots C_N\}$, denoting the number of elements in a set by $|\,.\,|$ and denoting the power set of U by

$pow(U) = \{\phi,\{C_1\},\ldots,\{C_N\},\{C_1,C_2\},\ldots,\{C_{N-1},C_N\},\ldots,\{C_1,\ldots,C_N\}\}$, the number of terms involved in calculating $P\left(\bigcup_{r=1}^{N} C_r\right)$ is now given by $|pow(U)| = 2^N - 1$. This follows, because each member of the power set of U determines uniquely a corresponding probability term in equation 2.

In the worst case, $2^N - 1$ probability distributions must be calculated where N is the number of condition-classes giving complete coverage of all class combinations. This is discussed further in Section 5.

It will be shown in Section 5.3 that it is the condition-class ranking which is important when the estimation problem is transformed into an exclusive condition-class problem. Accurate probability estimation is desirable to quantify the likelihood of one or more faults occurring but it is the ordering of condition-classes which is maintained following posterior knowledge. The correct ordering which reflects the actual circumstances must be preserved by estimation. This is only true in the case of the *exclusive* representation of condition-classes. For the overlapping representation, the ordering may change when equation 2 is applied. Furthermore, certain probabilistic constraints must not be violated if the outcomes of applying equation 2 is not to be meaningless.

5. The Estimation Problem

The unprocessed condition monitoring data will consist monitored parameter vectors with attached fault labels derived from a fault detection method. The inclusion of posterior information requires posterior probabilities to be estimated either directly, or indirectly from this data.

A common method of estimating posterior probabilities is to use an artificial neural network (e.g. Bishop, 1995; Richard and Lippmann, 1991). Where the condition-classes are exclusive, given N classes, there arises the 1 from N estimation problem, that is, for each input, one condition-class will be chosen on the basis of the posterior probabilities. Where the classes are non-exclusive, more than one condition can occur simultaneously giving rise to an m from n estimation problem. It has been shown (e.g. Bishop, 1995; Richard and Lippmann, 1991) that for both the mean squared error (MSE) and cross entropy (CE) measures, the neural networks will estimate the total Bayesian posterior probabilities of the form $P(C_i|\mathbf{x})$ only. Thus, although joint class information (m from n) is available in the training vectors, a neural network will not be able to estimate the joint probability function unless the output space is expanded to give an equivalent 1 from n problem. To capture class combination information in general, an augmented output vector consisting of $2^N - 1$ outputs is required. It is shown in Marriott and Harrison (1997) that the expansion is valid by treating the output space as a collection of disjoint sets.

5.1 Expanding the Output Space

The motivation for seeking a partition of the output space is that we need to expand the space to estimate all of the probabilities required for the update equation. In other words, the condition-class dependencies indicated by more than one 'on bit' in the output vector.

A partition of classified input space may be achieved by specifying that the class intersections are pairwise disjoint, for example C_i' only contains data points that belong to C_i and not $C_i \cap C_j$ etc. Similarly, $\left(C_i \cap C_j\right)'$ only contains data points that belong to $C_i \cap C_j$ and not $C_i \cap C_j \cap C_k$ etc. This will ensure a partition of the space with disjoint sets as required (e.g. $C_i' \cap \left(C_i \cap C_j\right)' = \phi$). The 'dash' notation is used throughout to indicate partition members which compose the entire sample space.

Now, the original formula for the union of sets in terms of set intersections can be specified in purely additive terms:

$$P\left(\bigcup_{r=1}^{N} C_r|\mathbf{x}\right) = \sum_{i=1}^{N} P(C_i'|\mathbf{x}) + \sum_{\substack{i=1,j=2 \\ j\neq k}}^{N} P\left(\left(C_i \cap C_j\right)'|\mathbf{x}\right) + \sum_{\substack{i=1,j=2,k=3 \\ i\neq j\neq k}}^{N} P\left(\left(C_i \cap C_j \cap C_k\right)'|\mathbf{x}\right) \vdots + P\left(\left(C_1 \cap C_2 \cap \ldots \cap C_N\right)'|\mathbf{x}\right)$$

That this partitioned representation is formally equivalent to the overlapping representation can be proved by representing the partitioning sets in terms of their overlapped counterparts. This way, expressions involving the condition-classes can be shown to be equivalent (Marriott and Harrison, 1997).

5.2. Using Radial Basis Function Networks to Estimate the Posterior Probabilities

One way of estimating posterior probabilities is to use a radial basis function network (RBFN) (e.g. Powell, 1987; Broomhead and Lowe, 1988; Moody and Darken, 1989; Bishop,1993 1995; Haykin, 1994; Wasserman, 1993).

This paper deals with classification problems which necessitates the use of the *softmax function* (e.g Bishop, 1995) to ensure that the total probability is equal to one. To prevent over-learning of the training data, *regularisation* (Bishop, 1991, 1993,1995) may be used. The total cost function for any error-driven neural network using regularisation will be given by

$$C = E + \nu\Omega$$

where E is the original error function, ν is the regularisation constant and Ω is the regularisation function. For the simulations of this sub-section and those of the jet engine fault data, the second-order differential regularisation function is given by

$$\Omega = \sum_{i=1}^{N}\sum_{l=1}^{L}\left(\frac{\partial^2 y_i}{\partial x_l^2}\right)^2 .$$

Details of the implementation of an RBFN network with second-order differential with a cross-entropy cost function regularisation applied to a standard network configuration with a

softmax layer will be found in Appendix B. Second-order differential regularisation penalises large changes in the curvature of the output function thus smoothing the resultant function.

The following dependent condition-classes were generated using Gaussian distributions for the likelihoods of: C_1, C_2, C_3, $C_1 \cap C_2$, and $C_2 \cap C_3$. The RBFN is expected to approximate the posterior probabilities $P(C_1|\mathbf{x}), P(C_2|\mathbf{x}), P(C_3|\mathbf{x})$, $P(C_1 \cap C_2|\mathbf{x})$, and $P(C_2 \cap C_3|\mathbf{x})$. The RBFN used had an expanded output set consisting of 5 outputs, each output signifying that case alone e.g. $P(C_1|\mathbf{x})$ gives the posterior probability of condition-class 1 occurring alone. To be consistent with earlier notation: $P(C_i|\mathbf{x}) = P(C_i'|\mathbf{x})$ and $P(C_i \cap C_j|\mathbf{x}) = P\left((C_i \cap C_j)'\middle|\mathbf{x}\right)$.

Figure 7 shows the estimated posterior probabilities without regularisation.

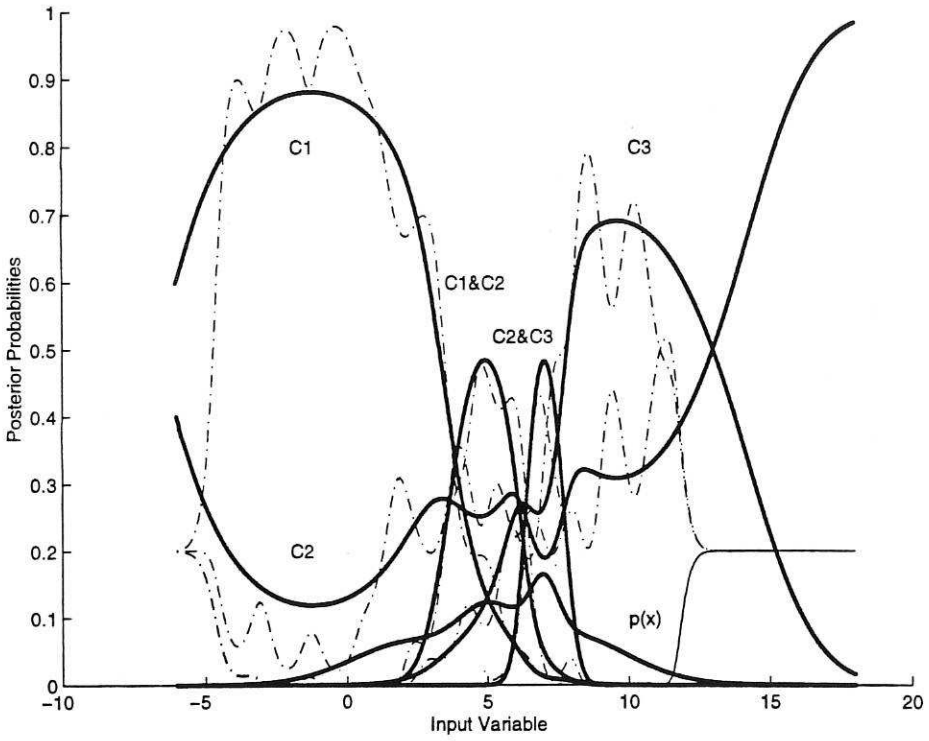


Figure 7. A graph showing the estimation of posterior probabilities by a radial basis function network.

The data density outside of the range $[-3,+12]$ is low giving inaccurate predictions of the posterior probability functions as expected. The lack of regularisation allows over-learning of the data and is indicated by the considerable curvature of the estimated probability functions.

Figure 8 shows the estimated posterior probabilities with second-order differential regularisation.

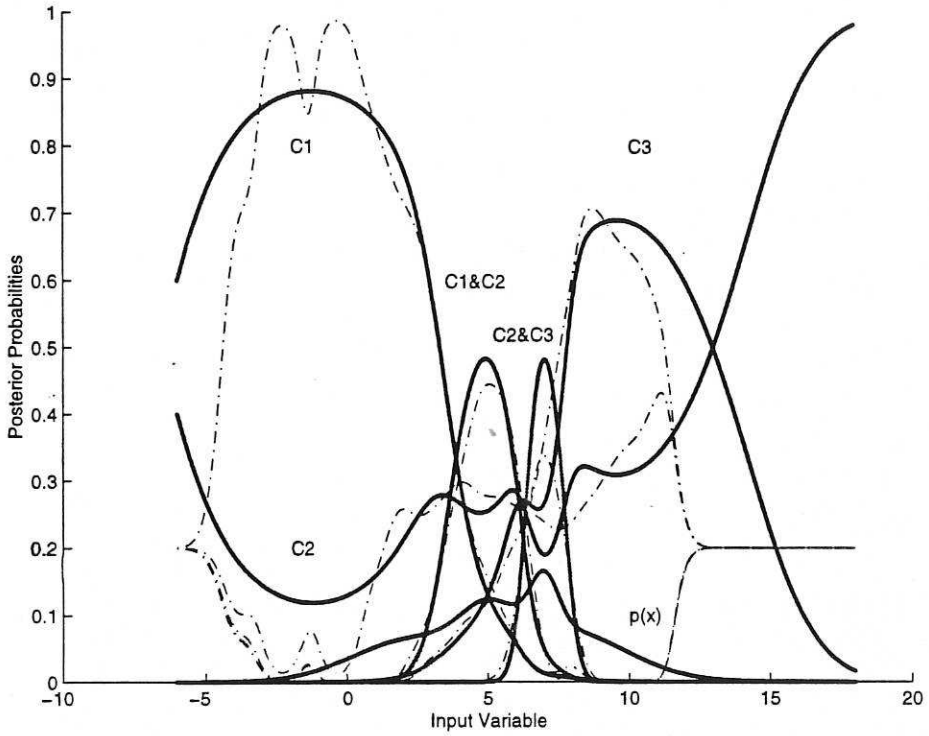


Figure 8. A graph showing the estimation of posterior probabilities by a radial basis function network using second-order differential regularisation as explained in the text.

Note that the approximated functions are considerably smoother in the region of higher data density.

5.3 Exclusive Set Probability Estimation and the Ranking

Reformulating the overlapping set representation in terms of exclusive sets increases the number of neural network outputs but renders the update problem easier by virtue of the ECR theorem. That is, the update problem is always an exclusive set problem which merely requires that the remaining sets are renormalised after the exclusion of sets using posterior knowledge. This being the case, it is easy to see that rank ordering is preserved following posterior knowledge; it follows that the rank order may only be important and that accurate probability estimation is not necessary providing the estimated probabilities have the same rank order as the actual probabilities.

This relaxed condition of rank ordering is not the case with the overlapping sets; it does not follow that the rank ordering will be preserved when posterior knowledge is included because the ranked probabilities are *combinations* of the exclusive probabilities. Furthermore, where the overlapping sets are reconstructed from the exclusive sets, differences in probability, although not disrupting the exclusive class rank ordering, may combine to change the overlapping class rank ordering. So, accurate probability estimation is still required in the overlapping class case.

6. Jet Engines: a Brief Tutorial

The turbo-jet engine is divided into two parts: the engine proper and the engine accessories. The accessories provide power for various aircraft systems and also include the engine control system. The basic mechanism of jet engine operation is the intake of air which is then burnt with fuel to produce exhaust gases (Rolls-Royce, 1986). The exhaust gases provide direct thrust and turn a turbine which compresses the air prior to combustion. The engine working cycle consists of four main phases: air intake, air compression, combustion and exhaust. A schematic block-diagram of the engine is shown in Figure 9.

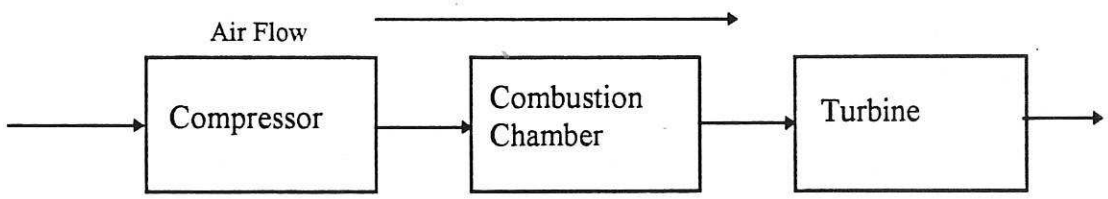


Figure 9. A schematic diagram of the main jet engine components. The compressor compresses air prior to being mixed with fuel in the combustion chamber. The hot air is used to drive a turbine which powers the compressor.

6.1 The Engine

A triple-spool axial flow compressor is shown in Figure 10. Each spool is driven by its own turbine and consists of multiple stages, each of which increases the air pressure by a small amount prior to combustion. Each spool is connected to its respective turbine by a shaft. The shaft speeds are monitored by the control system. Unusual vibrations may occur in one or more of the shafts indicating a problem (Tarassenko, 1996, 1997). Bearing faults can show up as shaft vibrations. Various safety features are built-in to detect and deal with overspeed or shaft breakage.

The electronic engine control (EEC) has two main functions: the monitoring of engine shaft speeds and the monitoring of the exhaust gas temperature (EGT) (Rolls-Royce, 1986). The EEC attempts to control the engine and maintain the set operating points requested by the operator. Fault diagnosis is complicated by the corrective action taken by the EEC. A fault may occur which is then compensated for by automatic adjustment of specific control parameters.

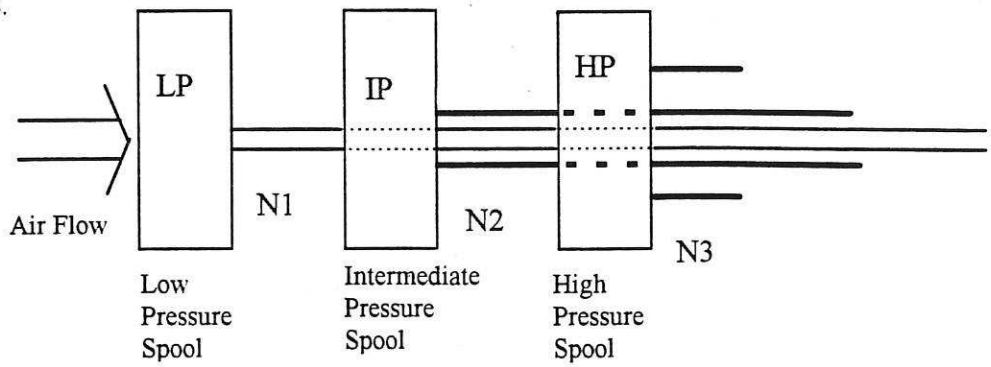


Figure 10. A triple spool compressor. N1,N2,N3 indicate low, intermediate and high pressure shaft speeds respectively

More details about the engine structure and operation will be introduced as required in later sections.

7. Simulation Methodology

This section refers to the Rolls-Royce Trent 700 engine model. Further details will be given which build upon the those of Section 6. To reiterate the point of the introduction, the methods of fault detection used here are simple classical range-check methods. This is to provide data for post-detection and feedback processing. The posterior knowledge integration methods are general and can be applied to any fault detection system which provides posterior probability (or density) estimates for each condition-class or conjunction of condition-classes where condition-classes overlap.

Here, fault detection is based upon model reference where faults are detected by checking parameter ranges with their computed values. The Trent 700 model is run at a different nominal operating point (NOP) for each run. Each NOP is determined by the Mach number and altitude (MA) settings. The NOP at a given MA determines the key parameters used by the condition monitoring system. Faults may or may not be induced at each run depending upon pre-determined probabilities. This provides a set of raw fault data for the fault detection methods used here, as a precursor to the posterior knowledge inclusion method.

7.1 Dependent Parameters and Induced Faults

The Mach and altitude settings are chosen at random from a uniform distribution in the ranges [0.3,0.8] and [20000,80000] ft respectively. This method is used for simplicity. A more realistic flight envelope could be chosen and is a possible extension.

The MA settings give rise to a series of dependent parameters which include the sub-set:

1. WF: Fuel flow,
2. TGT: Turbine gas temperature sensed by thermocouple, at entry to the LP turbine
3. N3: HP shaft speed,
4. P30: HP compressor delivery pressure (total),
5. T30: HP compressor delivery temperature (total).

This sub-set has been chosen specifically as a first approximation. This the minimal set of useful parameters chosen in conjunction with engineers from Rolls-Royce plc. An extended, richer set of parameters is a subject for further investigation.

Induced fault cases are chosen to be:

1. Fuel decay
2. TGT increase
3. N3 decay
4. Normal (N) Fault not induced ie running at NOP
5. No fault found (NFF): fault not induced but flagged in parameters

This set has been chosen as a first approximation to the problem. The fault severities are fixed but may be varied in an extended model. Fuel decay is realistic in that partial blockages may occur in fuel pipes or filters. TGT increase and N3 decay may reflect a decrease in engine efficiency. No fault found reflects the condition in which one or more faults are flagged by the system but no apparent cause can be found. Faults 1, 2 and 3 may also occur in binary combination giving eight fault conditions in total. These are f1, f2, f3, f1&f2, f1&f3, f2&f3, N, and NFF. The condition f1&f2&f3 has been omitted for simplicity.

Priors are chosen for each of the eight cases to reflect the mix of actual faults, no fault found and normal conditions. For case 5, no fault found, a single fault is chosen at random from faults 1 to 3 depending upon the priors.

7.2 Fault Diagnosis Issues

The engine and accessories comprise a dynamical system and fault detection / diagnosis is not a straightforward task. Fault diagnosis is a research area in itself to determine both what constitutes a fault and how to detect faults. When faults occur, the EEC attempts to compensate for the problem and maintain the desired set-point further complicating the problem. Many design choices have had to be made to allow the generation of data of sufficient complexity and realism to investigate methods of posterior knowledge inclusion which is the primary remit of this work.

Even seemingly straightforward fault diagnosis methods such as range checking pose several problems. How are the ranges to be determined across the operating envelope? What constitutes a fault? How are fault labels to be associated with sets of parameters? Will fault conditions in steady-state mode following EEC intervention be mistaken for NOPs? In the latter case, will range checking be of any use? How are static faults to be detected and represented? How are dynamical faults to be detected and represented?

The steady-state reached following the injection of a fault may give a parameter vector commensurate with normal operation in a different region of the operating envelope. The labelling of the final parameter vector as indicating a fault may provide misinformation to a FD system. It may be possible to prevent this by including the Mach number and altitude information in the parameter vector.

The N3 decay fault is a case in point. Whilst operating at a NOP during an experiment, (fixed Mach number and altitude) a decay was introduced into the HP shaft (which may indicate bearing faults for example). An expected increase in fuel flow (WF) occurred as the EEC attempted to compensate for the problem. There was an initial surge in fuel followed by reversion to a steady-state value of fuel flow not significantly above the original. The final parameter vector may be indicative of a normal (N) state. In this case, the FD problem is one of detecting a fuel surge not accounted for by normal operation or allowed transients. This involves the detection of dynamical anomalies—research topic in itself. Such considerations belong to the domain of fault diagnosis proper. Here, the concern is with post-processing of fault occurrence probabilities, hence our crude simulation and detection methods.

7.3 Fault Protocol

The protocol used for generating the fault data used in exploring the probability estimation and update problems is:

for each choice of Mach-Altitude co-ordinates,

i) run the model at the NOP without any faults,

ii) run the model with faults induced based upon the priors

iii) compare NOP run with the possible fault run, then

iv) find the maximum absolute percentage deviation across the time-trace of the variable for each indicator; if this exceeds the limit set for that particular variable, flag a fault.

For the experiments detailed here a data set of 800 training patterns and 300 test patterns was used. The faults actually induced were:

i) Fuel decay (increase), C1

ii) TGT increase, C2

iii) N3 decay, C3

iv) Fuel decay (increase) and TGT increase, C1 & C2

v) Fuel decay (increase) and N3 decay, C1 & C3

vi) TGT increase and N3 decay, C2 & C3

Faults are not induced for a number of cases which are thus considered normal. From these normal cases, a fraction is assigned a false alarm or no fault found. The faults are induced according to the prior probabilities. The following list of priors was used in the preliminary experiments featured in this paper. These could be adjusted to give a more realistic spread of fault/normal conditions.

Condition-Class	Prior Probabilities
C1	0.15
C2	0.15
C3	0.15
C1 & C2	0.05
C1 & C3	0.05
C2 & C3	0.05
No fault / normal(N):	0.3
No fault found (NFF)	0.1

Table 2. The induced condition-class priors

As mentioned in step iv) of the protocol, faults are assigned on the basis of range checking compared with the fault-free model for the five parameters detailed above. The fault ranges of all five variables are all set at (±10%) for simplicity; they can all be set at separate limits if required.

7.4 The fault vector coding scheme

The fault vectors are coded using a 5 bit input string which indicates the occurrence or non-occurrence of a limit-trip on each of the monitored variables. A seven bit output is used to indicate the following fault conditions: C1, C2, C3, C1&C2, C1&C3, C2&C3, N. The case C1&C2&C3 is prevented from happening by not allowing all three faults to be induced at any one time, i.e the condition is ignored. This is using one from many coding. Thus, the three condition-classes can be indicated separately or in pairs or the plant can be operating normally. The output vector represents verified fault / no fault occurrences. This form of input/output data indicates a binary heteroassociative problem.

An example data vector is (10111, 0000100). The first set of five bits is the alarm indicator set which signifies that there are trip deviations in fuel flow,N3, P30, and T30 but TGT is within range. It forms the input pattern to the neural network The second set of seven bits indicates the verified status. It represents the training input and represents a verified fault in C1 & C3, i.e, there is a verified fuel problem and a problem with the HP turbine (N3). The network has to develop a mapping between input alarms and verified faults.

7.5 The RBFN Network

A Radial Basis Function network of the sort discussed in sub-section 5.2 was used in the simulations of this paper. The network had a cross-entropy cost-function and incorporated a softmax layer to reflect the output probabilities. Second-order differential regularisation was used to reduce the rate of curvature of the output to prevent over-fitting to the data. Details are given in Appendix B.

8. Results

The empirical training set and test set probabilities were computed from the data files. These were found by computing the relative frequencies for each input vector.

8.1 The Theoretical Maximum Accuracy

A linear network, incorporating a softmax output layer to allow for the representation of probabilities, was trained and tested with the 800/300 set using a variety of initial weights. The network was used to predict the most likely fault(s). The best performance was a prediction accuracy of 64% and this varied very little for different choices of the initial weight set; this indicated that the minimum mean-squared error for such a system had been achieved. Thus, comparing with the theoretical maximum prediction accuracy of 87.7% (calculated directly), it is clear that this is not a trivial problem solvable using a simple network. As the complexity of the fault data increases, it is likely that the linear system will have an even poorer performance. Measurement noise and quantisation of inputs possibly will reduce the accuracy further.

A single run of the regularised RBFN was carried out with the 800/300 data set to assess the network's accuracy in probability estimation. The prediction accuracy for the test set for the most likely fault scenario was 87.7%, the maximum possible; that is, if faults were chosen on the basis of probability magnitudes. How can this be? This is because the RBFN models the probability distribution and only the maximum probability for each of the binary input vectors is required. There may be a large error in the estimates of the probabilities which does not affect the MAP decision as long as the probability of the most probable prediction exceeds the others by a small margin. In other words, the winning probability only has to be largest. Thus, the probability density function may not be very accurate or representative of the underlying distribution of fault vectors but still allow the maximum achievable accuracy.

The test set results are shown in Table 3. Only 11 out of a possible 32 states occurred with this run; NB this would change for different values of the fault detection thresholds. For each input, the actual probabilities (relative frequencies) of occurrence are shown together with those predicted by the network. The column labelled p(x) shows the data distribution (relative

frequency) of the patterns. This is included to illustrate the variation of accuracy with data density.

The RBFN network was then trained and tested with an 800/300 data set based upon the 11/32 binary input vectors encountered in the above fault-induction experiments. This set was devised for use as a calibration check. This time, a single *unambiguous* input was assigned to each input on the basis of the maximum probabilities encountered in the previous experiments. The input vectors were distributed approximately according to the frequency of occurrence encountered above. Thus, the max theoretical accuracy of correct diagnoses was 100% The RBFN achieved this 100% target.

		C1	C2	C3	C1&C2	C1&C3	C2&C3	N	p(x)
input	actual	0	0	0	0	0	0	1	0.3100
00000	predicted	0.005029	0.012104	0	0	0	0	0.982867	
input	actual	0	0	0	0	0	0	1	0.0433
00100	predicted	0.164490	0.202020	0.008506	0.007712	0.006416	0.006438	0.604418	
input	actual	0	0.7910	0	0	0	0	0.2090	0.2233
01000	predicted	0.000097	0.883493	0.000068	0.000187	0.000001	0.000009	0.116146	
input	actual	0	0	0.6667	0.3333	0	0	0	0.0100
01011	predicted	0.010755	0.014644	0.850851	0.008370	0.011507	0.092996	0.010877	
input	actual	0	0	0.8750	0	0.1250	0	0	0.0267
01111	predicted	0.000516	0.000375	0.922690	0.000667	0.001522	0.073864	0.000366	
input	actual	0.8750	0	0	0	0	0	0.1250	0.1333
10000	predicted	0.941444	0.000090	0.000002	0.000078	0.000001	0.000001	0.058384	
input	actual	1	0	0	0	0	0	0	0.0133
10010	predicted	0.558011	0.053072	0.025421	0.051029	0.029564	0.015743	0.267161	
input	actual	0	0	0	0	1	0	0	0.0500
10111	predicted	0.000875	0.000359	0.005837	0.000367	0.991319	0.000811	0.000432	
input	actual	0	0	0	1	0	0	0	0.0533
11000	predicted	0.000629	0.000737	0.000058	0.998410	0.000010	0.000011	0.000144	
input	actual	0	0	0	1	0	0	0	0.0033
11010	predicted	0.087575	0.085436	0.094690	0.576361	0.042954	0.054367	0.058617	
input	actual	0	0	0.6000	0	0.0500	0.3500	0	0.1333
11111	predicted	0.000094	0.000036	0.719065	0.000333	0.021754	0.258647	0.000070	

Table 3. Test set results. The actual probabilities are the relative frequencies of the classes in the data set. The predicted results are those of the RBFN run. P(x) indicates the relative data frequency.

Note that not all of the condition-class rankings are correct; where they are incorrect (e.g. alarm condition 01111) the relative data frequency is low. Some type of error measure will have to be developed which allows comparison of rankings between experiments.

8.2 Discussion

As expected, the underlying statistics of the training set population are estimated by the RBFN system; where the estimated probabilities become inaccurate, the relative data frequency is low. For both training and test sets, the same decision as predicted by the actual probabilities will be taken in all cases using the predicted probabilities. Some states may not occur in the fault diagnosis procedure, e.g. 00001 which signifies that T30 has changed without any concomitant changes in other monitored variables.

The eleven observed states were as follows:

State 00000 indicates that no fault has occurred.

State 00100 indicates that a fault has occurred in N3 only. The probability estimations point to the fact that a NFF condition is actually the case because an actual N3 decay fault usually has an effect on other parameters such as fuel flow and TGT. Note the low data density.

State 01000 indicates that a fault has occurred with TGT only. It is highly likely that the fault lies with the TGT thermocouple because no other parameter changes have been noted. There is also the possibility that no fault has occurred.

State 01011 indicates that, as well as the thermocouple changes, there are concomitant changes in P30 and T30 indicating that the N3 shaft may be involved (N3 or N3&TGT). In both cases, the N3 activity does not show up as a fault. Note that the data density is low in this case.

State 01111 indicates that all faults are triggered except for the fuel flow. This is highly indicative of an N3 fault but the data density is low indicating that a fuel flow problem usually occurs as well. This is supported by the higher data density associated with state 11111.

State 10000 indicates that there is a fuel flow problem. When this occurs alone it is rarely a consequence of any other actual fault. However, there is a possibility that a NFF condition has occurred.

State 10010 indicates both a fuel flow and P30 problem. According to the test set statistics, it is always a fuel flow problem.

State 10111 indicates that all faults are triggered except for TGT. A low data density indicates that TGT is usually associated with N3 (alarm pattern 11111). Where TGT is omitted, WF and N3 together are expected according to the training or test data.

State 11000 indicates that WF and TGT have occurred together. The actual faults are WF and TGT because the occurrence of N3 usually has a 'knock-on' effect.

State 11010 indicates that it is again a conjunction of WF and TGT but the P30 fault is anomalous as shown by the data density.

State 11111 is either indicative of N3 alone or N3 and TGT. Note the ratio of occurrences of approximately 3:1 is commensurate with the ratio of prior probabilities of 0.15:0.05 or 3:1.

This preliminary empirical investigation indicates that fault induction and detection using the aircraft engine model will provide data suitable for testing and extending the posterior knowledge inclusion model. The fault induction and detection process is to be refined so that meaningful posterior probability hierarchies will be generated.

9. Conclusions

It has been stated that, in general, condition monitoring involves the detection of anomalous conditions which arise during the operation of some plant or process. The indication of the most likely fault and its estimated probability by a fixed pattern recognition system is not necessarily the end-point. Condition monitoring is a continuous, closed-loop process involving an end-user. The end-user ultimately decides how to use the information generated by the condition monitoring system. The end-user may, in turn, require a mechanism of incorporating his or her observations into the condition monitoring system for a more accurate diagnosis. The incorporation and utilisation of posterior knowledge presents a difficult problem. This paper has attempted both to articulate the problem and to provide a framework for its solution. It is clear that more work is required in this area. The contrived example illustrates some of the issues involved in the integration of posterior knowledge within the human / machine diagnostic cycle as fault evidence is accumulated. Three phases of the fault diagnosis cycle have been identified:

(i) fault diagnosis and isolation to provide fault data,

(ii) probability estimation to provide the fault hierarchy, and

(iii) posterior knowledge inclusion to provide a revised fault hierarchy.

Phases (i) and (ii) are covered by many condition monitoring schemes. Phase (iii) has been explored in this paper.

The problem of posterior knowledge representation is a difficult one and further work is needed to increase the scope beyond just excluding classes on the basis of external observations; work is being done to investigate this. Both the method of knowledge representation and the posterior probability update problem are independent of the method used for probability estimation; this is because of the general framework based upon set theory.

Using the specified probabilistic framework, the posterior knowledge inclusion problem has been reduced to an m from n estimation problem. Furthermore, The m from n estimation problem has been reduced to a 1 from n problem by expansion of the output space. Pre-processing may be required to reduce the combinatorial explosion. It is clear that probability

distribution estimation methods must give sufficiently accurate estimates to maintain condition class hierarchies; the estimation problem has been explored further using an established neural network technology. The radial basis function network has the combined features of cross-entropy, a softmax layer and second order regularisation. On-going work using the simulated aircraft engine model is being carried out to explore these issues further. Now that it is feasible to generate fault data using the aircraft engine model and process this fault data using a RBF network, the use of the model will be refined and extended.

The authors would like to acknowledge the support of both the Engineering and Physical Sciences Research Council of the UK and Rolls-Royce PLC in the production of this work.

References

Applebaum, D (1996) *Probability and Information: An Integrated Approach* CUP Cambridge

Bernardo, J. M. and Smith, A. F. M. (1994) *Bayesian Theory* John Wiley and Sons Ltd, Chichester.

Bilchev, G.; Parmee, I.(1996) Constraint handling for the fault coverage code generation problem: an inductive evolutionary approach, Proc. International Conference on Evolutionary Computation -The 4th International Conference on Parallel Problem Solving from Nature. p. 880-9

Bishop, C. M. (1991) Improving the Generalisation Properties of Radial Basis Function Neural Networks, *Neural Computation* **3**, 4, 579-588

Bishop, C. M. (1993) Curvature-Driven Smoothing: A Learning Algorithm For Feedforward Networks *IEEE Transactions on Neural Networks* **4**(5) 882-884

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

Bogunovic, N. Mesic, T. (1996) Adaptive uncertainty management for a class of diagnostic expert systems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 10,5, 421-9

Boudoud, A. N. and Masson, M. H. (1996) The diagnosis of a technological system: on-line fuzzy clustering using a gradual confirmation of prototypes. *Proc. CESA '96 IMACS Multiconference. Computational Engineering in Systems Applications.* Vol. 1. 110-115

Broomhead, D. S. and Lowe, D. (1988) Multivariable Function Interpolation and Adaptive Networks, *Complex Systems* **2** 321-355

Dimla, D.E, Lister, P. M. and Leighton, N. J. (1997) Tool condition monitoring in metal cutting through application of MLP neural networks *IEE Colloquium on Fault Diagnosis in Process Systems (Digest No. 1997/174)* 9/1-3

Ding, Y.; Wach, D. (1994) A rule- and case-based hybrid system for rotating machinery diagnosis, *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes* - SAFEPROCESS '94. Preprints vol.2. 475-8

Dorr, R.; Kratz, F.; Ragot, J.; Loisy, F.; Germain, J.-L. (1997) *IEEE Transactions on Control Systems Technology* Vol: 5 Iss: 1 p. 42-60

Duda and Hart (1973) *Pattern Classification and Scene Analysis*

Durrett, R. (1994). *The Essentials of Probability* The Duxbury Press. Belmont California.

Eryurek, E.; Upadhyaya, B.R. (1995) An integrated fault-tolerant control and diagnostics system for nuclear power plants, *Proceedings of the Topical Meeting on Computer-Based Human Support Systems: Technology, Methods, and Future*. 267-74

Haykin, S. (1994) *Neural networks a Comprehensive Foundation* Macmillan

Hines, J.W.; Miller, D.W.; Hajek, B.K.(1995) Fault detection and isolation: a hybrid approach *Proceedings of the Topical Meeting on Computer-Based Human Support Systems: Technology, Methods, and Future*, 363-70

Gomm, J.B. (1994) Fault detection in a multivariable chemical process by monitoring process dynamics, *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS '94.* Preprints. Vol. 1, 177-82

Grimmet, G. R. and Stirzaker, D. R. (1992) *Probability and Random Processes*, Oxford Science Publications, Oxford.

Isermann, R. (1997). Supevision, Fault-Detection and Fault-Diagnosis Methods-an Introduction, *Control Engineering Practice*: **5** (5) 639-651

Isermann, R. (1994) Integration of fault detection and diagnosis methods. *Proc. IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS '94. Preprints* . 597-612 vol.2

Karsai, G.; DeCaria, F. (1997) Model-integrated on-line problem-solving in chemical engineering, *Control Engineering Practice*, Vol: 5 Iss: 1 1-9

Keravnou, E. T. and Johnson, L. (1986) *Competent Expert Systems: A Case Study in Fault Diagnosis* McGraw Hill NY

Krause, P. and Clarke, P. (1993) *Representing Uncertain Knowledge: An Artificial Intelligence Approach.* Intellect Books Oxford

Kneale, W. (1949) *Probability and Induction* Oxford at the Clarendon Press, Oxford

Korbicz, J and Kus, J (1996) Structural-parametrical identification method in fault detection and diagnosis systems. *Proc. CESA '96 IMACS Multiconference. Computational Engineering in Systems Applications.*Vol. 1. 696-700

Korbicz, J.; Kus, J. (1995) Knowledge-based fault detection system using genetic observer approach, *Proceedings of the 12th International Conference on Systems Science* 262-70 vol.3

Krishnaswami, V.; Rizzoni, G. (1994) A survey of observer based residual generation for FDI, *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS '94. Preprints.* 34-9 vol.1

Li, X.Q.; Wong, Y.S.; Nee, A.Y.C. (1996) Tool condition monitoring using acoustic emission sensing and an integrated multi-ART 2 neural network. *Proc. Advanced Manufacturing Processes, Systems, and Technologies (AMPST 96)* 193-200

Lianhui Chen; Ho, E. (1994) Improving fault diagnosis performance with contextual knowledge, Proc. *The Third International Conference on Automation, Robotics and Computer Vision.* vol.2,1338-42

Liu, T.I.; Singonahalli, J.H.; Iyer, N.R. (1996) Detection of roller bearing defects using expert system and fuzzy logic *Mechanical Systems and Signal Processing* Vol: 10 Iss: 5 p. 595-614

Logan, D.; Mathew, J. (1996) Using the correlation dimension for vibration fault diagnosis of rolling element bearings .I. *Basic concepts. Mechanical Systems and Signal Processing* Vol: 10 Iss: 3 p. 241-50

Ma Yizhong (1996), Diagnosis for signals in multiple correlated processes *Computers & Industrial Engineering* Vol: 31 Iss: 3-4 p. 817-20

McDonald, J. R., Burt, G. and Moyes, A. (1996) Knowledge based systems for condition monitoring, *Proc. UKACC International Conference on Control '96*, vol.2, 1424-9

MacIntyre, J.; O'Brien, J.C. (1995) Investigations into the use of wavelet transformations as input into neural networks for condition monitoring, *Artificial Neural Nets and Genetic Algorithms. Proceedings of the International Conference*. 116-19

Marriott, S, and Harrison, R. F. (1997). *The use of posterior knowledge in statistical pattern recognition with particular application to fault diagnosis.* Research Report No. 676 May 1997 The University of Sheffield, U.K.

Marriott, S. and Harrison, R. F. (1998) *The integration of posterior knowledge into statistical pattern recognition systems with particular application to fault diagnosis*, Proceedings of EIS'98, In Press

Melsa, J. L. and Cohn, D. L. (1978) *Decision and Estimation Theory.* McGraw-Hill

Milne, R.; Nicol, C.; Trave-Massuyes, L.; Quevedo, J. (1996) TIGER: knowledge based gas turbine condition monitoring, *AI Communications* Vol: 9 Iss: 3 p. 92-108

Moody, J. and Darken, C. J. (1989) Fast Learning in Networks of Locally-Tuned Processing Units, *Neural computation* **1** (2) 281-294

Pan, M. C., Sas, P. and van Brussel, H. (1996) Non-stationary time-frequency analysis for machine condition monioring. *Proc. IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*. 477-480.

Patel, V.C.; Kadirkamanathan, V.; Kulikov, G.G.; Arkov,V.Y.; Breikin, T.V. (1996) Gas turbine engine condition monitoring using statistical and neural network methods, *Proc. IEE Colloquium on Modelling and Signal Processing for Fault Diagnosis* (Ref. No.1996/260), 1/1-6

Patel, V. C., Kadirkamanathan, V., Thompson, H. A. and Flemming, P. J.(1996) Development of a gas turbine engine model for fault diagnosis. *Proc. IASTED International Conference. Artificial Intelligence, Expert Systems and Neural Networks*. 379-382.

Patton, R. J. and Chen, J. (1997) Observer-Based Fault Detection and Isolation: Robustness and Applications, *Control Engineering Practice*, **5** (5) 671-682

Patton, R., J., Frank, P. M. and Clark, R. N. (eds.) (1989) *Fault Diagnosis in Dynamic Systems, Theory and Application* Control Engineering Series. Prentice hall, London

Perrott, S.N.; Perryman, R.(1995) Adaptive resonance theory applied to condition monitoring of combined heat and power systems, *UPEC '95. 30th Universities Power Engineering Conference 1995. Conference Proceedings.* Vol.1 45-8

Preston, G. J. Shields, D. N. and Daley, S. (1996) Application of a robust nonlinear fault detection observer to an hydraulic system. *Proc. UKACC International Conference on Control '96*. vol. 2 1484-9.

Richard, M. D. and Lippmann, R. P. (1991) Neural Network Classifiers Estimate Bayesian *a posteriori* Probabilities. *Neural Computation*, **3** 461-483

Rodd, M. G. (ed. in chief) (1997) *Control Engineering Practice*: Special section on fault detection and diagnosis. **5** (5)

Rolls-Royce, (1986) *The Jet Engine*, Rolls-Royce plc.

Tarassenko, L (1996) *Novelty Detection* Neural Computing Applications Forum digest supplement Edinburgh

Tarassenko, L (1997) *Novelty Detection*: *Key questions*, Neural Computing Applications Forum digest supplement, Bath

Trave-Massuyes, L and Milne, R (1997) Gas-turbine condition monitoring using qualitative model-based diagnosis *IEEE Expert* 12,3 22-31

Walpole, R. E. and Myers, R. H. (1989) *Probability and Statistics for Engineers and Scientists* Macmillan Publishing Company New York

Wang Xue; Yang Shuzi (1996) A parallel distributed knowledge-based system for turbine generator fault diagnosis, *Artificial Intelligence in Engineering* Vol: 10 Iss: 4 p. 335-41

Wang, X.Z.; Lu, M.L.; McGreavy, C. (1997) Learning dynamic fault models based on a fuzzy set covering method, *Computers & Chemical Engineering* Vol: 21 Iss: 6 p. 621-30

Wasserman, P. D. (1993) *Advanced Methods in Neural Computing* VNR New York

Weighell, M. Martin, E. B. and Morris, A. J. (1997) Fault diagnosis in industrial process manafacturing using MSPC. *IEE Colloquium on Fault Diagnosis in Process Systems* (Digest No. 1997/174) 4/1-3

Wilson, D.J.H, Irwin, G. W. and Lightbody, G. (1997) Neural networks and multivariate SPC *IEE Colloquium on Fault Diagnosis in Process Systems* (Digest No. 1997/174) 5/1-5

Yang, H.; Saif, M. (1996) Monitoring and diagnostics of a class of nonlinear systems using a nonlinear unknown input observer. *Proceedings of the 1996 IEEE International Conference on Control Applications*. 1006-11

Siyu Zhang; Ganesan, R.; Sankar, T.S. (1995) Self-organizing neural networks for automated machinery monitoring systems *Computers in Engineering*, p. 1001-9

Zhang, Q (1996) Using non-linear black-box models in fault detection. *Proc. 35th IEEE conference on Decision and Control*. Vol. 1 636-7

Appendix A

A1. Exclusive sets:

Two classes A and B are *mutually exclusive* or disjoint if $A \cap B = \phi$, that is, if A and B have no elements in common.

A2. Conditional Probability

The *conditional probability* of B, given A, denoted by $P(B|A) = \frac{P(A \cap B)}{P(A)}$ if $P(A) > 0$

A3. Identity

Proof of the result

$$P(A \cap B^c) = P(A \cup B) - P(B)$$

used in the proof of equation (2).

$$\begin{aligned} P(A \cap B^c) &= P(A) - P(A \cap B) \\ &= P(A) + P(B) - P(B) - P(A \cap B) \\ &= P(A) + P(B) - P(A \cap B) - P(B) \\ &= P(A \cup B) - P(B) \end{aligned}$$

A4. Set Union

This can be proved by induction on n (e.g. Grimmet and Stirzaker, 1992).

$$\begin{aligned} P\left(\bigcup_{s=1}^{K} C_s\right) = &\sum_{i=1}^{K} P(C_i) \\ &- \sum_{i<j}^{K} P(C_i \cap C_j) \\ &+ \sum_{i<j<k}^{K} P(C_i \cap C_j \cap C_k) \\ &\vdots \\ &+ (-1)^{K+1} P(C_1 \cap C_2 \cap \ldots \cap C_K) \end{aligned}$$

A5 Total Probability

Lemma (Grimmet and Stirzaker, 1992):

For any events A and B

$$P(A) = P(A|B)P(B) + P(A|B^c)P(B^c)$$

More generally, let $B_1, B_2, \ldots, B_N$ be a partition of U. Then,

$$P(A) = \sum_{i=1}^{N} P(A|B_i)P(B_i)$$

A6 The Use of Bayes Theorem.

Posterior probabilities can be estimated directly if certain techniques are used. In some cases, however, it may be more appropriate to use Bayesian decision theory, and compute the posterior probabilities indirectly rather than estimating them directly. Bayesian decision theory is a framework for calculating the required conditional probabilities from other empirically derivable probabilities (e.g. Duda and Hart, 1973; Gelman *et al*, 1995). Bayes' theorem for real valued data variables is of the form

$$P(C_i|\mathbf{x}) = \frac{p(\mathbf{x}|\mathrm{C}_i)P(\mathrm{C}_i)}{p(\mathbf{x})} \qquad \text{(A1)}$$

where $P(C_i|\mathbf{x})$, is the *posterior probability*, $p(\mathbf{x}|\mathrm{C}_i)$ is the *likelihood*, $P(\mathrm{C}_i)$, is the *prior probability* of class *i* occurring and $p(\mathbf{x})$ is the *unconditional density function*. These probabilities are estimated from the data.

For a set of *exclusive* classes, the form of $p(\mathbf{x})$ is given by

$$p(\mathbf{x}) = \sum_{i}^{N} p(\mathbf{x} \cap C_i) = \sum_{i}^{N} p(\mathbf{x}|C_i)P(C_i) \qquad \text{(A2)}$$

as $\boldsymbol{x}$ belongs to a single class only. Equation (A3) ensures that the posterior probabilities sum to unity, i.e.,

$$\sum_{i=1}^{N} P(C_i|\mathbf{x}) = 1 \qquad \text{(A4)}$$

Equation (A3) is a special case of the more general case involving non-exclusive classes given by

$$\begin{aligned}
p(\mathbf{x}) &= p(\mathbf{x} \cap U) = \\
p\left(\mathbf{x} \cap \left(\bigcup_{r=1}^{N} C_i\right)\right) &= \sum_{i=1}^{N} p(\mathbf{x} \cap C_i) \\
&\quad - \sum_{i<j}^{N} p(\mathbf{x} \cap C_i \cap C_j) \\
&\quad + \sum_{i<j<k}^{N} p(\mathbf{x} \cap C_i \cap C_j \cap C_k) \\
&\quad \vdots \\
&\quad + (-1)^{N+1} p(\mathbf{x} \cap C_1 \cap C_2 \cap \ldots \cap C_N)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{i=1}^{N} p(\mathbf{x}|C_i) P(C_i) \\
&\quad - \sum_{i<j}^{N} p(\mathbf{x}|C_i \cap C_j) P(C_i \cap C_j) \\
&\quad + \sum_{i<j<k}^{N} p(\mathbf{x}|C_i \cap C_j \cap C_k) P(C_i \cap C_j \cap C_k) \qquad \text{(A5)} \\
&\quad \vdots \\
&\quad + (-1)^{N+1} p(\mathbf{x}|C_1 \cap C_2 \cap \ldots \cap C_N) P(C_1 \cap C_2 \cap \ldots \cap C_N)
\end{aligned}$$

where equation (A5) ensures that the probability of the union of the classes conditional upon $\boldsymbol{x}$ is unity, i.e every input is classified.

$$\begin{aligned}
P(U|\mathbf{x}) &= \\
P\left(\left(\bigcup_{r=1}^{N} C_i\right)|\mathbf{x}\right) &= \sum_{i=1}^{N} P(C_i|\mathbf{x}) \\
&\quad - \sum_{i<j}^{N} P(C_i \cap C_j|\mathbf{x}) \\
&\quad + \sum_{i<j<k}^{N} P(C_i \cap C_j \cap C_k|\mathbf{x}) \qquad \text{(A6)} \\
&\quad \vdots \\
&\quad + (-1)^{N+1} P(C_1 \cap C_2 \cap \ldots \cap C_N|\mathbf{x}) \\
&= 1
\end{aligned}$$

where $P(C_i \cap C_j|\mathbf{x}) = \dfrac{p(\mathbf{x}|C_i \cap C_j)P(C_i \cap C_j)}{p(\mathbf{x})}$ etc.

Equation (A6) reduces to Equation (A4) when $C_i \cap C_j = \phi$, i.e. the classes are exclusive giving rise to the usual definition of Bayes' theorem (e.g. Walpole and Myers, 1989):

Given a partition of the event space, $\{B_1,\ldots,B_N\}$ that is $B_i \cap B_j = \phi, \quad \forall i \neq j$, and a set A such that $A \subseteq \bigcup_{k=1}^{N} B_k$, the conditional probability, $P(B_i|A)$ can be written as

$$P(B_i|A) = \frac{P(B_i|A)}{P(A)} = \frac{P(B_i)P(A|B_i)}{\sum_{j}^{N} P(B_j)P(A|B_j)}$$

Note that the condition that $B_i \cap B_j = \phi, \quad \forall i \neq j$ is required.

A7. Conditional Independence

(Bernardo and Smith, 1994; Grimmet and Stirzaker, 1992)

Definition:

Two events A and B are called *conditionally independent* given C if

$P(A \cap B|C) = P(A|C)P(B|C)$.

In general, a family of events $\{C_i\}, \quad i = 1\cdots N$ is conditionally independent if

$$P\left(\bigcap_i C_i \middle| \mathbf{x}\right) = \prod_i P(C_i|\mathbf{x})$$

A8. The Posterior Probability Update Equation

Equation (2) can be proved formally as follows:

$$P\left(C_{\delta_i}\middle|\left(\bigcap_k C_{\delta_k}^c\right)\cap\mathbf{x}\cap\varepsilon\right)=\frac{P\left(C_{\delta_i}\cap\left(\bigcap_k C_{\delta_k}^c\right)\cap\mathbf{x}\right)}{P\left(\left(\bigcap_k C_{\delta_k}^c\right)\cap\mathbf{x}\right)}$$ by definition of conditional probability

$$=\frac{P\left(C_{\delta_i}\cap\left(\bigcap_k C_{\delta_k}^c\right)\middle|\mathbf{x}\right)p(\mathbf{x})}{P\left(\left(\bigcap_k C_{\delta_k}^c\right)\middle|\mathbf{x}\right)p(\mathbf{x})}$$ by $P(A\cap B)=P(A|B)P(B)$

$$=\frac{P\left(C_{\delta_i}\cap\left(\bigcap_k C_{\delta_k}^c\right)\middle|\mathbf{x}\right)}{P\left(\left(\bigcap_k C_{\delta_k}^c\right)\middle|\mathbf{x}\right)}$$ by cancellation of $P(\mathbf{x})$

$$=\frac{P\left({}^cC_{\delta_i}\cap\left[\left(\bigcap_k C_{\delta_k}^c\right)^c\right]\middle|\mathbf{x}\right)}{P\left(U\cap\left[\left(\bigcap_k C_{\delta_k}^c\right)^c\right]^c\middle|\mathbf{x}\right)}$$ by $\left(A^c\right)^c=A$ and $U\cap A=A$

$$=\frac{P\left(C_{\delta_i}\cup\left(\bigcap_k C_{\delta_k}^c\right)^c\middle|\mathbf{x}\right)-P\left(\left(\bigcap_k C_{\delta_k}^c\right)^c\middle|\mathbf{x}\right)}{P\left(U\cup\left(\bigcap_k C_{\delta_k}^c\right)^c\middle|\mathbf{x}\right)-P\left({}^c\left(\bigcap_k C_{\delta_k}^c\right)\middle|\mathbf{x}\right)}$$ by $P\left(A\cap B^c\right)=P(A\cup B)-P(B)$

$$=\frac{P\left(C_{\delta_i}\cup\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)-P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}{P\left(U\cup\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)-P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}$$ by de Morgan's law (e.g. Applebaum, 1996)

$$=\frac{P\left(C_{\delta_i}\cup\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)-P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}{P(U|\mathbf{x})-P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}$$

$$= \frac{P\left(\left(\bigcup_j C_{\delta_j}\right)\middle|\mathbf{x}\right) - P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}{P\left(\left(\bigcup_l C_{\delta_l}\right)\middle|\mathbf{x}\right) - P\left(\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}$$

where the fact that the union of the classes is exhaustive has been used.

After including the dependency upon $\boldsymbol{x}$, equation (2) is now written in terms of probabilities conditional upon the input

$$\begin{aligned} P\left(\bigcup_{s=1}^{K} C_{\delta_s}\middle|\mathbf{x}\right) &= \sum_{i=1}^{K} P\left(C_{\delta_i}\middle|\mathbf{x}\right) \\ &\quad - \sum_{i<j}^{K} P\left(C_{\delta_i} \cap C_{\delta_j}\middle|\mathbf{x}\right) \\ &\quad + \sum_{i<j<k}^{K} P\left(C_{\delta_i} \cap C_{\delta_j} \cap C_{\delta_k}\middle|\mathbf{x}\right) \qquad \text{(A7)} \\ &\quad \vdots \\ &\quad + (-1)^{K+1} P\left(C_{\delta_1} \cap C_{\delta_2} \cap \ldots \cap C_{\delta_K}\middle|\mathbf{x}\right) \end{aligned}$$

to include the conditional probabilities of equation (2). Equation (A7) can be proved easily by using the distributivity of set relations and substituting $C_{\delta_i} \cap \mathbf{x}$ for C_{δ_i} in the general form of $P\left(\bigcup_{s=1}^{K} C_{\delta_s}\right)$ (e.g Durrett, 1994, Grimmet and Stirzaker, 1992) where K is the number of sets involved in the union:

$$\begin{aligned} P\left(\bigcup_{s=1}^{K} C_{\delta_s}\right) &= \sum_{i=1}^{K} P\left(C_{\delta_i}\right) \\ &\quad - \sum_{i<j}^{K} P\left(C_{\delta_i} \cap C_{\delta_j}\right) \\ &\quad + \sum_{i<j<k}^{K} P\left(C_{\delta_i} \cap C_{\delta_j} \cap C_{\delta_k}\right) \\ &\quad \vdots \\ &\quad + (-1)^{K+1} P\left(C_{\delta_1} \cap C_{\delta_2} \cap \ldots \cap C_{\delta_K}\right) \end{aligned}$$

in terms of probabilities of occurrence.

A9. *Theorem*: Exclusive Class Renormalisation (ECR) Theorem

For a set of exclusive classes, the updated posterior probabilities, following the exclusion of the set, will be given by a renormalisation of the remaining probabilities.

Proof:

From equation (2)

$$P\left(C_{\delta_i}|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(\bigcup_j C_{\delta_j}|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}|\mathbf{x}\right)}{P\left(\bigcup_l C_{\delta_l}|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}|\mathbf{x}\right)}$$

where $j \in \{\delta_i\} \cup \Delta_\varepsilon$, $k \in \Delta_\varepsilon$, and $l \in \Delta_r \cup \Delta_\varepsilon$.

For the set of exclusive classes, the following equations hold:

$$P\left(\bigcup_j C_{\delta_j}|\mathbf{x}\right) = \sum_j P\left(C_{\delta_j}|\mathbf{x}\right) = P\left(C_{\delta_i}|\mathbf{x}\right) + \sum_k P\left(C_{\delta_k}|\mathbf{x}\right) \qquad \text{(A8)}$$

$$P\left(\bigcup_k C_{\delta_k}|\mathbf{x}\right) = \sum_k P\left(C_{\delta_k}|\mathbf{x}\right) \qquad \text{(A9)}$$

and

$$P\left(\bigcup_l C_{\delta_l}|\mathbf{x}\right) = \sum_r P\left(C_{\delta_r}|\mathbf{x}\right) + \sum_k P\left(C_{\delta_k}|\mathbf{x}\right) \qquad \text{(A10)}$$

where $r \in \Delta_r$

Substituting (A8), (A9) and (A10) into (2) gives

$$P\left(C_{\delta_i}|\bigcap_k C_{\delta_k}^c \cap \mathbf{x}\right) = \frac{P\left(C_{\delta_i}|\mathbf{x}\right)}{\sum_r P\left(C_{\delta_r}|\mathbf{x}\right)} \qquad \text{(A11)}$$

Equation (A11) ensures that

$$\sum_i P\left(C_{\delta_i} | \bigcap_k C^c_{\delta_k} \cap \mathbf{x}\right) = \sum_i \frac{P\left(C_{\delta_i} | \mathbf{x}\right)}{\sum_r P\left(C_{\delta_r} | \mathbf{x}\right)} = \frac{1}{\sum_r P\left(C_{\delta_r} | \mathbf{x}\right)} \sum_i P\left(C_{\delta_i} | \mathbf{x}\right) = 1$$

where $i \in \Delta_r$ ■

A10. ***Theorem:*** **Independent Class Renormalisation (ICR) Theorem**

For a set of independent classes, the updated posterior probabilities, following the exclusion of this set, will be given by a renormalisation of the remaining probabilities.

Proof:

From equation (2)

$$P\left(C_{\delta_i} | \bigcap_k C^c_{\delta_k} \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(\bigcup_j C_{\delta_j} | \mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k} | \mathbf{x}\right)}{P\left(\bigcup_l C_{\delta_l} | \mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k} | \mathbf{x}\right)}$$

where $j \in \{\delta_i\} \cup \Delta_\varepsilon$, $k \in \Delta_\varepsilon$, and $l \in \Delta_r \cup \Delta_\varepsilon$.

Now,

$$P\left(\bigcup_j C_{\delta_j} | \mathbf{x}\right) = P\left(\left(\bigcup_k C_{\delta_k}\right) \cup C_{\delta_i} \middle| \mathbf{x}\right)$$

Further expansion gives

$$P\left(\bigcup_j C_{\delta_j} | \mathbf{x}\right) = P\left(C_{\delta_i} \middle| \mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k} \middle| \mathbf{x}\right) - P\left(\left(\bigcup_k C_{\delta_k}\right) \cap C_{\delta_i} \middle| \mathbf{x}\right) \quad \text{(A12)}$$

and

$$P\left(\bigcup_l C_{\delta_l}|\mathbf{x}\right) = P\left(\left(\bigcup_k C_{\delta_k}\right)\cap\left(\bigcup_r C_{\delta_r}\right)\middle|\mathbf{x}\right)$$
$$= P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) - P\left(\left(\bigcup_r C_{\delta_r}\right)\cap\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) \quad \text{(A13)}$$

where $r \in \Delta_r$

For independent sets,

$$P\left(\left(\bigcup_k C_{\delta_k}\right)\cup C_{\delta_i}\middle|\mathbf{x}\right) = P\left(C_{\delta_i}|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) \quad \text{(A14)}$$

and

$$P\left(\left(\bigcup_r C_{\delta_r}\right)\cap\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) = P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\mathbf{x}\right) \quad \text{(A15)}$$

substituting (A14) and (A15) into (A12) and (A13) respectively gives

$$P\left(\bigcup_j C_{\delta_j}|\mathbf{x}\right) = P\left(C_{\delta_i}|\mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) - P\left(C_{\delta_i}|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) \quad \text{(A16)}$$

and

$$P\left(\bigcup_l C_{\delta_l}|\mathbf{x}\right) = P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) - P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) \quad \text{(A17)}$$

Finally, substituting (A16), and (A17) into (2) gives

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(C_{\delta_i}\middle|\mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) - P\left(C_{\delta_i}\middle|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}{P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right) + P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) - P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}$$

giving

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(C_{\delta_i}\middle|\mathbf{x}\right)}{P\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)} \qquad \text{(A18)}$$

Where the fact that $P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right) = 0$ has been used to indicate that these classes have been excluded in this particular case.

Equation (A18) ensures that the union of adjusted posterior probabilities is equal to 1.

For the specific case were the remaining sets are exclusive

$$P\left(\bigcup_i C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \sum_i P\left(C_{\delta_i}|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \sum_i \frac{P\left(C_{\delta_i}|\mathbf{x}\right)}{\sum_r P\left(C_{\delta_r}|\mathbf{x}\right)} = \frac{1}{\sum_r P\left(C_{\delta_r}|\mathbf{x}\right)} \sum_i P\left(C_{\delta_i}|\mathbf{x}\right) = 1$$

where $i \in \Delta_r$ ■

Where excluded classes are independent, the remaining probabilities are renormalised as the excluded classes have no effect on the outcomes.

Theorem: The Dependent Class (DC) Theorem

For non-exclusive and dependent classes, neither the ECR Theorem nor the ICR Theorem applies.

Proof;

From (2)

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(\bigcup_j C_{\delta_j}\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}{P\left(\bigcup_l C_{\delta_l}\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}$$

where $j \in \{\delta_i\} \cup \Delta_\varepsilon$, $k \in \Delta_\varepsilon$, and $l \in \Delta_r \cup \Delta_\varepsilon$. This expression may be expanded to give

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(C_{\delta_i} \cup \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}{P\left(\left(\bigcup_r C_{\delta_r}\right) \cup \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}$$

$$= \frac{P\left(C_{\delta_i}\middle|\mathbf{x}\right) + P\left(\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)\right) - P\left(C_{\delta_i} \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}{P\left(\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)\right) + P\left(\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)\right) - P\left(\left(\bigcup_r C_{\delta_r}\right) \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) - P\left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)}$$

$$= \frac{P\left(C_{\delta_i}\middle|\mathbf{x}\right) - P\left(C_{\delta_i} \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}{P\left(\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)\right) - P\left(\left(\bigcup_r C_{\delta_r}\right) \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}$$

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c \cap \mathbf{x} \cap \varepsilon\right) = \frac{P\left(C_{\delta_i}\middle|\mathbf{x}\right) - P\left(C_{\delta_i} \cap \left(\bigcup_k C_{\delta_k}\middle|\mathbf{x}\right)\right)}{P\left(\left(\bigcup_r C_{\delta_r}\middle|\mathbf{x}\right)\right) - P\left(\left(\bigcup_r C_{\delta_r}\right) \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)}$$

There are intersecting terms in both the numerator and denominator which are non-zero. This precludes using a simple renormalisation to give the revised probabilities. These are only zero for exclusive and independent classes.

Now, for exclusive and independent classes

$$P\left(C_{\delta_i} \cap \left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right) = 0$$

and

$$P\left(\left(\bigcup_r C_{\delta_r}\right)\cap\left(\bigcup_k C_{\delta_k}\right)\middle|\mathbf{x}\right)=0$$

which gives

$$P\left(C_{\delta_i}\middle|\bigcap_k C_{\delta_k}^c\cap\mathbf{x}\right)=\frac{P\left(C_{\delta_i}|\mathbf{x}\right)}{P\left(\left(\bigcup_r C_{\delta_r}|\mathbf{x}\right)\right)}$$

as stated by the ECR and ICR formulae ■.

Appendix B.

Radial-Basis Function Network (RBFN) with a Softmax layer using Cross-Entropy and Second-Order Regularisation

The following analysis is similar to the one carried out for the Multilayer Perceptron in Bishop, 1993.

1. The Error Function

For a training set of P patterns classified into N classes of fault conditions, the combined error term consisting of cross-entropy and regularisation components is given by

$$E = \sum_{p=1}^{P} \left\{ E_p^{CE} + \upsilon E_p^R \right\} \qquad \text{(B1)}$$

where the cross-entropy term per pattern is defined as

$$E_p^{CE} = \sum_{n=1}^{N} t_n^p \ln\left(\frac{t_n^p}{y_n^p}\right) \qquad \text{(B2)}$$

and the regularisation term per pattern is given by

$$E_p^R = \frac{1}{2} \sum_{n=1}^{N} \sum_{l=1}^{L} \left(\frac{\partial y_n^p}{\partial \left(x_l^p\right)^2} \right)^2 \qquad \text{(B3)}$$

The RBFN consists of a layer of L input nodes feeding into a layer of J basis function nodes. The layer of J basis function nodes feeds forward into a layer of N output nodes; the N outputs are then fed to a softmax function which provides the final outputs.

The final outputs are given by

$$y_i = f\left(a_i\right) \qquad \text{(B4)}$$

where

$$f(a_i) = \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} \qquad \text{(B5)}$$

is the softmax function,

$$a_i = \sum_{j=1}^{J} w_{ij} z_j \qquad \text{(B6)}$$

is the net output feeding into the ith output node, and

$$z_j = \phi_j(\mathbf{x}) \qquad \text{(B7)}$$

is the output from the j th basis function.

Gradient descent methods require the calculation of the gradient, $\frac{\partial E}{\partial w_{ij}}$

The gradient can be decomposed to give

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}}\left\{\sum_{p=1}^{P}\left[E_p^{CE} + E_p^R\right]\right\}$$

$$= \sum_{p=1}^{P}\left[\frac{\partial E_p^{CE}}{\partial w_{ij}} + \frac{\partial E_p^R}{\partial w_{ij}}\right]$$

Now, the gradients $\frac{\partial E_p^{CE}}{\partial w_{ij}}$ and $\frac{\partial E_p^R}{\partial w_{ij}}$ defined per pattern are required.

To reduce notational complexity, the superscript p is be dropped.

2. The Cross-Entropy Gradient Component

Applying the chain rule of differentiation gives

$$\frac{\partial E^{CE}}{\partial w_{ij}} = \frac{\partial E^{CE}}{\partial a_i}\frac{\partial a_i}{\partial w_{ij}} \qquad \text{(B8)}$$

where

$$\frac{\partial E^{CE}}{\partial a_i} = \sum_{i'=1}^{N}\frac{\partial E^{CE}}{\partial y_{i'}}\frac{\partial y_{i'}}{\partial a_i} \qquad \text{(B9)}$$

by applying the chain rule once again.

From Equation (B

$$\frac{\partial E^{CE}}{\partial y_{i'}} = \frac{\partial}{\partial y_{i'}}\left\{\sum_{n=1}^{N} t_n \ln\left(\frac{t_n}{y_n}\right)\right\}$$

$$= t_{i'}\left(\frac{t_{i'}}{y_{i'}}\right)^{-1}(-1)(y_{i'})^{-2}t_{i'}$$

giving

$$\frac{\partial E^{CE}}{\partial y_{i'}} = -\frac{t_{i'}}{y_{i'}} \qquad \text{(B10)}$$

$$\frac{\partial y_{i'}}{\partial a_i} = \frac{\partial}{\partial a_i}\left\{\frac{e^{a_{i'}}}{\sum_k^N e^{a_k}}\right\}$$

$$= \frac{\left(\sum_k^N e^{a_k}\right)\frac{\partial}{\partial a_i} e^{a_{i'}} - e^{a_{i'}} e^{a_i}}{\left(\sum_k^N e^{a_k}\right)^2}$$

$$= \frac{e^{a_{i'}}\delta_{ii'}}{\sum_k^N e^{a_k}} - \frac{e^{a_{i'}}}{\sum_k^N e^{a_k}} \cdot \frac{e^{a_i}}{\sum_k^N e^{a_k}}$$

giving

$$\frac{\partial y_{i'}}{\partial a_i} = (\delta_{ii'} - y_i) y_{i'} \qquad \text{(B11)}$$

Where δ_{ij} is the Kronecker delta function and,

$$\frac{\partial a_i}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}}\left\{\sum_{j=1}^J w_{ij} z_j\right\} = z_j \qquad \text{(B12)}$$

Substitute equations (B10), (B11) and (B12) into equation (B8)

$$\frac{\partial E^{CE}}{\partial w_{ij}} = \left(\sum_{i'=1}^N \frac{\partial E^{CE}}{\partial y_{i'}} \frac{\partial y_{i'}}{\partial a_i}\right) \frac{\partial a_i}{w_{ij}}$$

$$= \sum_{i'=1}^N \left(-\frac{t_{i'}}{y_{i'}}\right)(y_{i'}\delta_{ii'} - y_{i'} y_i) z_j$$

$$= \sum_{i'=1}^N (-t_{i'}\delta_{ii'} + t_{i'} y_i) z_j$$

$$= \left[\left(-\sum_{i'=1}^N t_{i'}\delta_{ii'}\right) + \left(\sum_{i'=1}^N t_{i'}\right) y_i\right] z_j$$

$$= (-t_i + y_i) z_j$$

because $\sum_{i'=1}^N t_{i'} = 1$

giving,

$$\frac{\partial E^{CE}}{\partial w_{ij}} = (y_i - t_i) z_j \qquad \text{(B13)}$$

i.e. the Widrow-Hoff rule.

3. The Regularisation Gradient Component

$$\frac{\partial E^R}{\partial w_{ij}} = \sum_{l=1}^{L} \frac{\partial E_l^R}{\partial w_{ij}} \qquad \text{(B14)}$$

{STEP}

$$\begin{aligned}
\frac{\partial E_l^R}{\partial w_{ij}} &= \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \cdot \frac{\partial^2}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} z_j \right) \\
&= \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \cdot \left[\frac{\partial}{\partial x_l} \left(\frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} z_j \right) \right) \right] \\
&= \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \cdot \left[\frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \frac{\partial z_j}{\partial x_l} + z_j \frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \right) \right) \right] \\
&= \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \left\{ \left[\frac{\partial y_n}{\partial a_i} \frac{\partial}{\partial \mathbf{x}_l} \left(\frac{\partial z_j}{\partial x_l} \right) + \frac{\partial z_j}{\partial x_l} \frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \right) \right] + \left[\frac{\partial z_j}{\partial x_l} \frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \right) + z_j \frac{\partial^2}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} \right) \right] \right\} \\
&= \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} \right) \frac{\partial^2 z_j}{\partial x_l^2} + 2 \sum_n^N \frac{\partial y_n}{\partial x_l^2} \frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \right) \frac{\partial z_j}{\partial x_l} + z_j \sum_n^N \frac{\partial y_n}{\partial x_l^2} \frac{\partial^2}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} \right)
\end{aligned}$$

Following Bishop (1993), this expression may be rewritten in the form

$$\frac{\partial E_l^R}{\partial w_{ij}} = \sigma_{li} \frac{\partial^2 z_j}{\partial x_l^2} + 2\hat{\sigma}_{li} \frac{\partial z_j}{\partial x_l} + \hat{\hat{\sigma}}_{li} z_j \qquad \text{(B15)}$$

where the following quantities have been defined (Bishop, 1993)

$$\sigma_{li} = \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} \right) \qquad \text{(B16)}$$

$$\hat{\sigma}_{li} = \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \frac{\partial}{\partial x_l} \left(\frac{\partial y_n}{\partial a_i} \right) \qquad \text{(B17)}$$

$$\hat{\hat{\sigma}}_{li} = \sum_n^N \frac{\partial^2 y_n}{\partial x_l^2} \frac{\partial^2}{\partial x_l^2} \left(\frac{\partial y_n}{\partial a_i} \right) \qquad \text{(B18)}$$

$$\frac{\partial y_n}{\partial a_i} = \left(\delta_{in} - y_i \right) y_n \qquad \text{(B19)}$$

Now the component derivatives are required in order to evaluate (B15).

The first derivative is

$$\frac{\partial}{\partial x_l}\left(\frac{\partial y_n}{\partial a_i}\right)=\frac{\partial}{\partial x_l}\left(\delta_{in}y_n-y_iy_n\right)$$

$$=\frac{\partial}{\partial x_l}\left(\delta_{in}y_n\right)-\frac{\partial}{\partial x_l}\left(y_iy_n\right)$$

giving

$$\frac{\partial}{\partial x_l}\left(\frac{\partial y_n}{\partial a_i}\right)=\delta_{in}\frac{\partial y_n}{\partial x_l}-y_i\frac{\partial y_n}{\partial x_l}-y_n\frac{\partial y_i}{\partial x_l} \qquad \text{(B20)}$$

which forms a component of (B17). Equation (B20) is differentiated again

$$\frac{\partial^2}{\partial x_l^2}\left(\frac{\partial y_n}{\partial a_i}\right)=\frac{\partial}{\partial x_l}\left(\delta_{in}\frac{\partial y_n}{\partial x_l}-y_i\frac{\partial y_n}{\partial x_l}-y_n\frac{\partial y_i}{\partial x_l}\right)$$

$$=\delta_{in}\frac{\partial^2 y_n}{\partial x_l^2}-y_i\frac{\partial^2 y_n}{\partial x_l^2}-\frac{\partial y_i}{\partial x_l}\frac{\partial y_n}{\partial x_l}-y_n\frac{\partial^2 y_i}{\partial x_l^2}-\frac{\partial y_n}{\partial x_l}\frac{\partial y_i}{\partial x_l}$$

giving

$$\frac{\partial^2}{\partial x_l^2}\left(\frac{\partial y_n}{\partial a_i}\right)=\delta_{in}\frac{\partial^2 y_n}{\partial x_l^2}-y_i\frac{\partial^2 y_n}{\partial x_l^2}-2\frac{\partial y_i}{\partial x_l}\frac{\partial y_n}{\partial x_l}-y_n\frac{\partial^2 y_i}{\partial x_l^2} \qquad \text{(B21)}$$

which is substituted into (B18).

To evaluate (B20) and (B21) the derivatives $\frac{\partial y_n}{\partial x_l}$ and $\frac{\partial^2 y_n}{\partial x_l^2}$ are required.

Now,

$$\frac{\partial y_n}{\partial x_l}=\sum_{n'=1}^{N}\frac{\partial y_n}{\partial a_{n'}}\frac{\partial a_{n'}}{\partial x_l}$$

$$=\sum_{n'=1}^{N}\left(\delta_{n'n}-y_{n'}\right)y_n\left(\sum_{j'=1}^{J}w_{n'j'}\frac{\partial z_{j'}}{\partial x_l}\right)$$

giving

$$\frac{\partial y_n}{\partial x_l}=\sum_{n'=1}^{N}\sum_{j'=1}^{J}w_{n'j'}\left(\delta_{n'n}-y_{n'}\right)y_n\frac{\partial z_{j'}}{\partial x_l} \qquad \text{(B22)}$$

and,

$$\frac{\partial^2 y_n}{\partial x_l^2}=\frac{\partial}{\partial x_l}\left\{\sum_{n'=1}^{N}\sum_{j'=1}^{J}w_{n'j'}\left(\delta_{n'n}-y_{n'}\right)y_n\frac{\partial z_{j'}}{\partial x_l}\right\}$$

$$=\sum_{n'=1}^{N}\sum_{j'=1}^{J}w_{n'j'}\left\{\frac{\partial}{\partial x_l}\left(\delta_{n'n}-y_{n'}\right).y_n\frac{\partial z_{j'}}{\partial x_l}+\left(\delta_{n'n}-y_{n'}\right)\frac{\partial y_n}{\partial x_l}\frac{\partial z_{j'}}{\partial x_l}+\left(\delta_{n'n}-y_{n'}\right)y_n\frac{\partial^2 z_{j'}}{\partial x_l^2}\right\}$$

giving

$$\frac{\partial^2 y_n}{\partial x_l^2} = \sum_{n'=1}^{N}\sum_{j'=1}^{J} w_{n'j'}\left\{-\frac{\partial y_{n'}}{\partial x_l} y_n \frac{\partial z_{j'}}{\partial x_l} + \left(\delta_{n'n} - y_{n'}\right)\frac{\partial y_n}{\partial x_l}\frac{\partial z_{j'}}{\partial x_l} + \left(\delta_{n'n} - y_{n'}\right) y_n \frac{\partial^2 z_{j'}}{\partial x_l^2}\right\} \quad \text{(B23)}$$

$$\frac{\partial^2 y_n}{\partial x_l^2} = \sum_{n'=1}^{N}\sum_{j'=1}^{J} w_{n'j'}\left\{\left[\left(\delta_{n'n} - y_{n'}\right)\frac{\partial y_n}{\partial x_l} - \frac{\partial y_{n'}}{\partial x_l} y_n\right]\frac{\partial z_{j'}}{\partial x_l} + \left(\delta_{n'n} - y_{n'}\right) y_n \frac{\partial^2 z_{j'}}{\partial x_l^2}\right\}$$

The evaluation of (B22) and (B23) require the derivatives $\frac{\partial z_j}{\partial x_l}$ and $\frac{\partial^2 z_j}{\partial x_l^2}$.

For a specific radial basis function used in this work:

$$z_j = \exp\left(-\frac{\left\|\mathbf{x} - \mathbf{x}_j\right\|^2}{2\sigma^2}\right) = \exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right)$$

$$\frac{\partial z_j}{\partial x_l} = -\frac{\left(x_l - x_{lj}\right)}{\sigma^2}\exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right) \quad \text{(B24)}$$

$$\frac{\partial^2 z_j}{\partial x_l^2} = \frac{\partial}{\partial x_l}\left(\frac{\partial z_j}{\partial x_l}\right) = \frac{\partial}{\partial x_l}\left\{-\frac{\left(x_l - x_{lj}\right)}{\sigma^2}\exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right)\right\}$$

$$= -\frac{\left(x_l - x_{lj}\right)}{\sigma^2}\left[-\frac{\left(x_l - x_{lj}\right)}{\sigma^2}\exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right)\right] + \left[-\frac{1}{\sigma^2}\right]\left(\exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right)\right)$$

giving

$$\frac{\partial^2 z_j}{\partial x_l^2} = \left[\frac{\left(x_l - x_{lj}\right)^2}{\sigma^4} - \frac{1}{\sigma^2}\right]\exp\left(-\frac{\sum_{l=1}^{L}\left(x_l - x_{lj}\right)^2}{2\sigma^2}\right)$$

The learning rule used to update the network weights was chosen to be a simple gradient descent rule of the form

$$w_{ij}(t+1) = w_{ij}(t) - \eta\frac{\partial E}{\partial w_{ij}}$$

where η is the learning rate. This rule was found to be sufficient to learn the desired probability distributions.