INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15-18, 2005

SENSITIVITY ANALYSIS: A PRIORI AND POST FACTO

M D Giess, S J Culley, B J Hicks

Abstract

The use of sensitivity analysis allows for parameters that influence system performance to be identified, and (in the case that such influence is undesirable) remedial action to be taken. An *a priori* sensitivity analysis requires that the system be modelled analytically, where the specific relationships between parameters must be understood and defined in order to construct this model. This is not always possible or practical. Data Mining (DM) analysis of manufacturing data allows sensitivity analysis to be performed *post facto* without recourse to analytical modelling, simply by recording and analysing manufacturing data generated as part of a standard manufacturing process. This paper considers whether it is possible to use a DM-based approach without recourse to the considerable overhead of *a priori* construction of an analytical model.

In this research a traditional analytical model was used to provide a benchmark sensitivity analysis ranking. This information had previously been used to good effect in industry, thus verifying its accuracy. This analytical model was used to generate 1000 instances of data representing manufacturing data, where variance within tolerance of the product parameters were mapped via the analytical model onto variance in product performance.

The next stage was to create DM models, in the form of Decision Tree Induction (DTI) and Artificial Neural Network (ANN)models, in order to model this data. Information was extracted from these models using sensitivity analysis for the ANN model and a bespoke significance metric approach for the DTI model, giving information in the form of a ranked list.

Correlation was seen to be good between information from the ANN and analytical models, where 3 of the 4 most influential parameters featured in both rankings, and the DTI model agreed with the analytical model ranking regarding the most influential parameter. The method of information extraction for the DTI models was suggested to require further refinement to improve the accuracy of ranking of the less significant parameters.

Keywords: Manufacturing data analysis, Data mining, Sensitivity analysis

1. Introduction

The research described in this paper is but one aspect of a project which aims to allow manufacturing data to be examined for the purposes of providing information to designers. This research project focuses exclusively upon data generated as part of a manufacturing operation, thus specifically excluding data generated during external experimentation. Many methods exist for the analysis of data generated during a manufacturing process, for example Taguchi [1] or other Design of Experiments methods [2], however these methods rely upon either an ability to control dependent variables or upon specifying an expected relationship within the data. In the case of manufacturing data it is suggested that neither situation can be guaranteed. Data Mining (DM) is a methodology that uses Machine Learning (ML) and associated modelling methods to analyse data free from the need to specify expected relationships, and where the modelling methods seek to extract relationships between data by considering the effects of inherent variation within the data.

DM, and the associated ML modelling methods, have previously been used within engineering, a notable example being the work of Reich and Barai [3]. DM is a complete methodology, extending from initial consideration of the nature of business and data through to modelling of data and evaluation and deployment of obtained knowledge. As a general observation, it has been noted that much of the literature describing DM applications within engineering focused upon the middle stages of analysis, the actual modelling of the data, at the expense of both the initial business and data understanding stages and the later evaluation and deployment stages.

In terms of addressing these neglected areas, this paper describes research into the later evaluation stage, where models created as part of the DM analysis are investigated and consideration is given to methods of extracting information from created models. Whilst establishing the accuracy of the created models is relatively trivial, achieved by evaluating model performance on specific validation datasets, a difficulty remains in verifying the accuracy of the extracted information.

This was addressed by using a previously completed, industrially validated study as a source of verified information against which information from the DM models could be compared. The previous study provided information in the form of a sensitivity analysis, indicating the effects of variation within each product characteristic upon product performance. Information from DM models was extracted in a similar form, allowing for direct comparison. Of great importance is the idea that the initial study is *a priori*, where the analysis is carried out before the product is manufactured, whereas the DM sensitivity analysis is *post facto*. The *a priori* analysis requires a means of physically modelling or mapping the performance of the product or system, whereas the DM analysis models the data generated during manufacturing. It is highly preferable to be able to model the system before the onset of manufacturing, as information may most efficiently be deployed at this point, but in many cases such modelling is either impractical or inaccurate. In such cases the DM analysis is useful as it requires no physical modelling of the system, and the data is generated via direct measurement of the actual product, which has been manufactured according to the designed process.

2. Rationale and Objectives

The research described in this paper is part of a wider project that seeks to provide means of extracting information from manufacturing data using DM methods. Issues related to the nature of manufacturing data are the topic of separate discussion, to be published in due course, and this paper focuses solely on the methods of creating and extracting information from DM models. It is intended to demonstrate the proposed approach via analysis of computationally derived data, generated from an analytical model. This analytical model will also be used as a basis for a sensitivity analysis, providing benchmark information against which the information from the DM models can be compared and contrasted.

The objectives of the research described in this paper are:

- To propose a methodology in which to create and interrogate DM models using engineering data
- To indicate the veracity of information extracted from ML models via direct comparison with an *a priori* sensitivity analysis.

3. Analytical Model

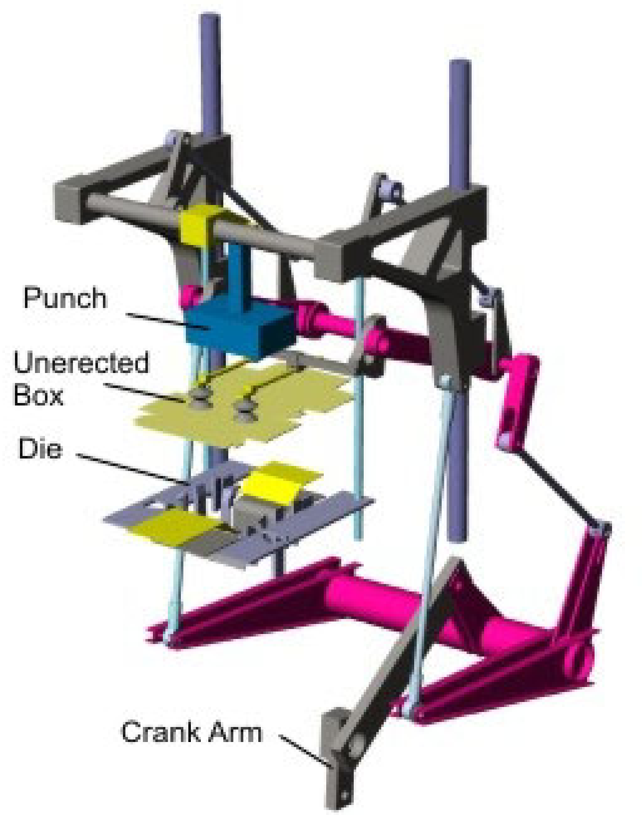

Figure 1 Packaging Machine Mechanism

The analytical model used in this research describes the mechanism of a packaging machine, whose construction can be seen in Figure 1. The machine takes flat-pack boxes from a stack and punches them into the required shape through a die. The mechanism was computationally modelled using the constraint-based modelling software environment SWORDS [4]. This software allows for mechanisms to be assembled and manipulated according to pre-defined constraints, and the motion of the mechanism can be reviewed to optimise both component paths and physical performance (such as velocity and acceleration).

As background, the mechanism under examination is based upon a machine used by an industrial collaborator. This mechanism was seen to be prone to malfunction in the event of excessive acceleration of the vacuum grips holding the flat pack boxes, where these vacuum attachments lost contact with the boxes. The mechanism is adjustable, allowing for different shaped boxes to be assembled, however this adjustability results in variance in the lengths of each linkage each time that the machine is adjusted for a specific box size. This variation in linkage length is considered to cause excessive acceleration and hence malfunction.

The analytical model was created for the purpose of deducing which specific linkage lengths contributed to this variance in acceleration. This was performed via the use of a sensitivity analysis. The results of this sensitivity analysis were used successfully in practice, where corrective efforts were focused upon this linkages whose variation in length were seen to have great impact upon the magnitude of acceleration. The issue under consideration is whether a non-analytical DM-based approach would be able to provide similar information and insights to the analytical model.

3.1 Sensitivity Analysis of Analytical Model

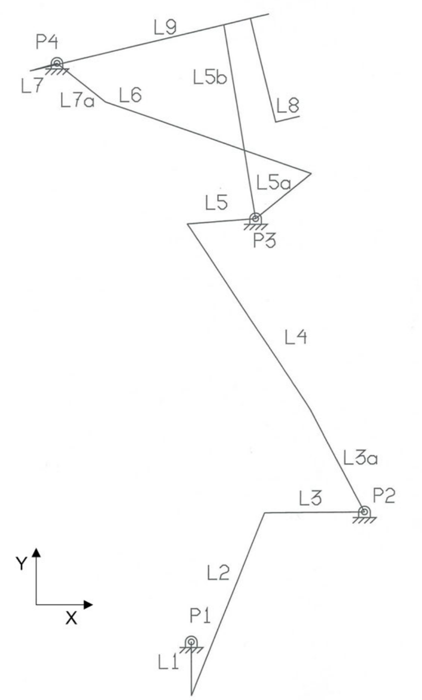

Figure 2 Mechanism Notation

Figure 2 shows a 2-dimensional representation of the mechanism and introduces the terminology that will be used to define each link in the mechanism. The geometry of this mechanism was entered into the constraint-based modeller as a series of linkages of appropriate length, and constraints were added to define the pivot points and the unions of each linkage.

Table 1 Form of Perturbation for Analytical Model Sensitivity Analysis

Parameter	Perturbation
Linkage Lengths	+/- 2% of linkage length
Pivot Points	+/-2% of overall mechanism height

The mechanism was cycled and the maximum acceleration seen over the entire cycle was recorded. This maximum acceleration, and the linkage lengths and pivot positions were used as a benchmark, representing the ideal arrangement and performance of the mechanism. The sensitivity analysis sought to identify parameters which exert a strong influence upon the acceleration of the working head, and this was performed via perturbation of each parameter, whilst maintaining the original settings for the remaining parameters, and recording the variation in maximum acceleration of the working head. Table 1 shows the form and magnitude of the perturbation for each different parameter. The pivot points were perturbed in both the X and Y directions as separate operations.

Table 2 Results of Analytical Model Sensitivity Analysis

Parameter	Change in Acceleration			Rank
	Positive Perturbation	Negative Perturbation	Summed Magnitude	
L5b	30.1	-22.8	**52.9**	**1**
L3	-10.5	-5.5	**16**	**2**
L4	-7.1	-7.6	**14.7**	**3**
L2	2.3	-6.9	**9.2**	**4**
L3a	1.5	-6.5	**8**	**5**
L7a	4	-3.9	**7.9**	**6**
P2_y	-3.6	-3.8	**7.4**	**7**
L1	2.4	-4.8	**7.2**	**8**
L6	0.6	-5.7	**6.3**	**9**
P4_x	2.7	-2.9	**5.6**	**10**

Table 2 shows the results of the analytical model sensitivity analysis, where the parameters listed in the first column are labelled in the manner defined in Figure 2. The pivot point information has a qualifier that describes variation in the Cartesian co-ordinate system (for example, p4_x indicates a variation in the position of pivot point number 4 in the x-direction). The magnitudes of the effect of positive and negative perturbation are summed for each parameter, and this summed metric is used to compile the ranking. These results will not be expanded upon here; they are intended simply to serve as a benchmark for the information from the DM models. It is emphasised again that these results were successfully deployed by the industrial collaborator for the purposes of system improvement, ensuring the veracity of this information.

3.2 Generation of Data for DM Modelling

The second task of the analytical model is to provide data for DM analysis, where such data will be taken to represent manufacturing data, or data that could feasibly be measured and recorded during a given manufacturing, assembly and testing process. In such a process each aspect of each component of a product will be assigned a manufacturing tolerance, and the specific value of each aspect will vary within this tolerance. It is suggested that the effects of these variations will manifest themselves as a variation of the performance of the product under test. The task of the DM analysis therefore is to relate the variation in performance (in this case maximum acceleration) to variation within each component, indicating where consistent trends are found across a number of products.

In order to generate data describing both variation within tolerance and the effects upon performance of such variation, it is necessary to define the tolerances and nature of variance. The tolerances used were the same as those listed in Table 1, where the linkages were assigned tolerances of between +/-2% for their lengths and the pivot points were free to move +/-2% of the total height of the mechanism in any direction. Random variance within these tolerances was specified, although it is possible that other, more skewed distributions might be seen in practice.

The input data, comprising the mechanism link lengths and pivot positions, were generated according to this variance within tolerance. This was achieved via the use of a simple macro implemented in the Microsoft Excel package [5]. 1000 such cases were generated, each one with random variations in each link length and pivot position, representing a batch of 1000 mechanisms produced within tolerance. The configurations of each of these 1000 exemplars were loaded in turn into SWORDS and the maximum acceleration seen in a cycle was then computed. The resultant dataset therefore contained information describing the link lengths, pivot positions and maximum accelerations for 1000 different mechanism configurations. This data could now be analysed using DM methods in order to deduce which input parameters (the link lengths and pivot positions) exerted the greatest influence upon the mechanism acceleration.

4. DM Modelling

The two algorithms used in the DM analysis were Decision Tree Induction (DTI) and Artificial Neural Networks (ANNs). Good introductory texts are given by Witten and Frank [6] and Fu [7] for the DTI and ANN algorithms respectively. The DTI algorithm used in this analysis is the C5.0 algorithm, a commercial algorithm developed from Quinlan's earlier (and freely distributed) C4.5 algorithm [8].

Whilst both algorithms provide a prediction of the likely value for an output metric given the values for a set of input variables, and both are created by analysis of previous exemplars with known inputs and outputs, the two algorithms are notably different in methods of creation, function and information extraction. The DTI algorithm assembles a tree structure which, via the use of logical conditional statements regarding the value of a given input parameter at each node or branch junction within the tree, leads to a given prediction within a leaf node. This prediction is give in the form of a class, where the output is considered to reside within one of a number of predefined, mutually exclusive classes or ranges. If, as is the case in this research, the output is continuous, it must be broken down into appropriate ranges.

The ANN algorithm assembles a network of interconnected nodes, where each interconnection is given a weighting which acts to multiply or attenuate the signal passed from each node along that specific connection. By iteratively adjusting the values of these weights, the network can be adapted to give a desired signal from the output nodes given specific signals to the input nodes. This iterative adjustment is carried out by attempting to minimise the error between actual signal and desired signal for each examplar within the dataset used for model creation, following a gradient descent. The back-propagation algorithm, as defined by Rumelhart *et al* [9] , is typically used for this task.

4.1 Creation of DTI Models

In the case of the DTI algorithm, the continuous output variable (the maximum acceleration) must be divided into appropriate ranges prior to the creation of the model. This was achieved by assigning instances (exemplars) to three ranges, A, B and C, where range A contained those instances with a low maximum velocity and range C those instances with a high maximum velocity. The boundaries of each range were assigned in such a way as to ensure equal density of population across all three ranges.

Each DTI model was created with a pruning severity of 90 and a minimum number of records per child branch of 2, where these specific values correspond to the implementation of C5.0 as seen in the Clementine proprietary DM package [10]. Both of these parameters assist in preventing the tree from overspecialising or overtraining, a phenomenon that occurs when the models extend to describe each specific instance as opposed to providing a concise description of general trends. In the event of overtraining these parameters may be adjusted to produce a more compact tree.

Table 3 Results of DTI Modelling

Model No	No. of training instances	No of Validation Instances	Training Accuracy (%)	Validation Accuracy (%)	Cross-validation Accuracy (%)
DTI-1	100	900	96	55.27	63
DTI-2	200	800	90	58.8	51
DTI-3	500	500	93.4	60.08	63.2

Table 3 shows the results of the DTI modelling. The three models all have training accuracies of 90% or greater, indicating that the models describe the functions within the training data well. The validation and Cross-validation accuracies are lower, however these are more representative measures of model accuracy and should be used in preference to training accuracy.

4.2 Accuracy Considerations

Accuracy is more faithfully evaluated via the use of external validation, and has been applied in two ways here. The validation accuracy depicts the percentage of correctly classified instances within a separate dataset, which was not used in the creation of the model. Whilst the training accuracy simply indicates how well the model has captured the relationship between input and output, it is possible that it has simply learnt the relationships by rote and has not captured the underlying process or relationships. By testing on unseen data, a more accurate impression of the success of modelling can be gained, as the model is forced to predict the output for inputs of which it has not previously seen, and as such it can only provide an accurate prediction if it has captured the relationship that projects input onto output. This method, however, has weaknesses in that the specific compositions of both the training and validation datasets have an effect upon modelling, and that data is held back from model creation, reducing the richness of data used in model creation. Cross-validation addresses both of these concerns by creating equally sized data samples that are used for both training and validation. A range of models is created, equal in number to the number of data sample sets, and for each model one data sample set in turn is excluded from training and used for validation. These validation accuracies are aggregated across each model in this range, and the entire dataset used to create the model. This aggregation attenuates the effect of validation dataset composition.

4.3 Creation of ANN Models

The ANN modelling was carried out using a standard feed-forward network with back-propagation, once again utilised via the Clementine DM software environment. The ANN algorithm requires significant initial user input in terms of specifying an architecture or topology for the network, effectively defining how many interconnected nodes will be used between the input and output nodes. This is an important decision, as the number of nodes controls the complexity of the network that can be fitted. Bishop (1995) gives a useful analogy with fitting a polynomial to data points; with too few coefficients the polynomial will be unable to capture the underlying function, and with too many the polynomial will start to measure the noise on the data and generalisation will be reduced. The topology can be manually prescribed or iteratively adjusted during training using methods such as pruning or the application of Simulated Evolution (of which Maniezzo provides a good example [11]). The data used in this research is computationally generated and is free from any significant noise, and hence overtraining is unlikely to be as prevalent as when applications map noisy data. In this respect manual definition of the topology was used in preference to the more computationally expensive methods of iterative adjustment.

As the topology is manually selected, it is unlikely to be optimal, and hence overtraining is likely to occur. To prevent this, a number of models were created using a separate test set, where model accuracy is evaluated using this test set of the training iteration and training terminated in the event that this test accuracy decreases over a number of cycles.

Table 4 Results of ANN Modelling

Model No	Train Data Amount	Test set %	Topology – input, hidden1, (hidden2,) output	Accuracy			
				Train		Valid	
				R-Squared	Pearson	R-Squared	Pearson
ANN-1	100	0	22,20,10,1	1.0000	1.0000	0.6670	0.8167
ANN-2	100	10	22,20,10,1	0.8494	0.9216	0.7832	0.8850
ANN-3	100	0	22,20,1	1.0000	1.0000	0.6625	0.8139
ANN-4	100	10	22,20,1	0.8625	0.9287	0.7883	0.8879
ANN-5	200	0	22,20,10,1	1.0000	1.0000	0.5333	0.7303
ANN-6	200	10	22,20,10,1	0.8124	0.9013	0.7986	0.8936
ANN-7	200	0	22,20,1	1.0000	1.0000	0.5875	0.7665
ANN-8	**200**	**10**	**22,20,1**	**0.8624**	**0.9286**	**0.7995**	**0.8941**
ANN-9	500	0	22,20,10,1	0.9972	0.9986	0.5847	0.7646
ANN-10	500	10	22,20,10,1	0.8443	0.9189	0.7962	0.8923
ANN-11	500	0	22,20,1	0.9959	0.9979	0.4262	0.6529
ANN-12	500	10	22,20,1	0.8299	0.9110	0.7778	0.8819

Table 4 shows the results of ANN modelling, indicating the amount of training data, the amount of data used as a test set, the topology of the network and the training and validation accuracies. The network is structured as a series of columns of nodes, with input into the left-hand column and output from the right-hand column, and hence the topology is defined by the numbers of nodes in each column or layer. In the mechanism under investigation here, there were 22 input nodes (link lengths and pivot positions) and 1 output node (maximum acceleration), and two arrangements were used for the intermediate 'hidden' layers, of either one layer of 20 nodes or 2 layers of 20 and 10 nodes respectively. The accuracies are quoted as both R-squared and Pearson correlation coefficients, where a score of 1 indicates perfect agreement between predicted and actual output and an inverse relationship is indicated by a score of 0 or -1 for the R-squared and Pearson respectively. The R-squared value is simply the square of the Pearson value, however both are commonly used and hence both are included here.

The accuracies of each model were evaluated using both the training dataset and a separate hold-out validation dataset. This method of validating model accuracy is sub-optimal, as discussed earlier, and suffers from high variance as highlighted by Reich [12]. The more suitable Cross-validation (as discussed in section 4.2) requires a range of models be created using differing data samples, and as training (period taken to fully complete model creation) can take a considerable time, in certain cases a period of days for an ANN model. Thus the decision was taken to forgo cross-validation of the purposes of his example study. It is suggested that cross-validation should be used where practicable in preference to hold-out validation.

There are signs of overtraining within the ANN models, where those models created without the use of a separate test set were seen to have greater disparity between training and validation accuracy. For example, models ANN-7 and ANN-8 were created using identical means except for the use of a test set comprising 10% of the available training data for model ANN-8. Despite this reduction in training data volume, ANN-8 has an R-squared value for validation of 0.7995 compared to 0.5875 for ANN-7, although ANN-7 has a training accuracy of 1 compared to 0.8628 for ANN-8. The variance in accuracy between models created with either 1 or 2 hidden layers is suggested to be inconclusive, as there are cases where models with 2 hidden layers have greater validation accuracies than those with 1 hidden layer (compare ANN-9 and ANN-10 against ANN-11 and ANN-12) whereas there are situations when the opposite is noted (ANN-5 against ANN-7).

The range of models present a number of options in terms of selecting a suitable information extraction candidate. The model with the highest validation accuracy was selected, in this case model ANN-8 as shown in Table 4. The following section discusses the methods of information extraction, and when applied to an ANN model in subsequent information comparison sections the information from the ANN model will be taken from model ANN-8.

4.4 Information from DM Models

As mentioned previously, the methods of information extraction are notably different for both DTI and ANN models. DTI models are constructed by consideration of a measure called an information gain, which is simply a metric describing how successfully the data is split by a given logical condition. Each possible logical condition is evaluated, and the most successful conditions in terms of dividing the data are credited with a high score for information gain, where successful division is considered to be that which most equally divides the data. By selecting the logical condition with the highest information gain (the one that most successfully divides the data), the algorithm has provided the greatest gain in information. The selected logical condition should (if binary) divide the data into two equally sized chunks, and should avoid the situation where only a small portion of the data is separated from the bulk by the logical condition. This infers that the logical condition chosen at the upper levels of a tree (the logical conditions selected early on) has the greatest influence upon system output, as it provides the greatest gain in information when seeking to predict the output.

The rationale that those logical conditions nearest to the top or root of the tree exert greatest influence upon tree (and hence system) output is used in the extraction of information. This method is simple in logic and implementation.

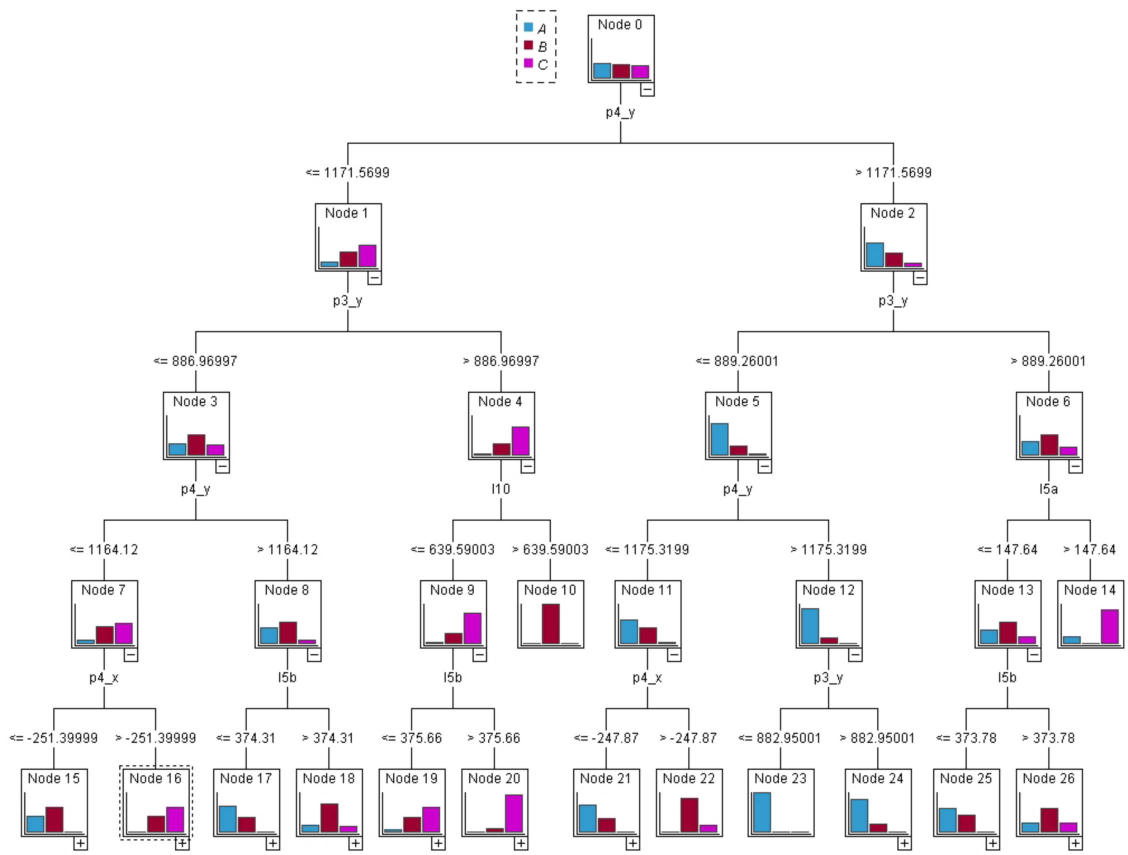

Figure 3 Decision Tree for Model DTI-3

Figure 3 shows a truncated tree for model DTI-3, as defined in Table 3. This tree is truncated for reasons of space, as the full tree is approximately four times larger. By starting at the root node, numbered node 0 in Figure 3, and proceeding along the branches according to the logical conditions at each node, a decision is reached at a leaf node. By scoring each parameter based upon its occurrence within the tree, and attaching greater weight to occurrences closer to the root node, an indication of the influence of that parameter upon system performance can be obtained. These scores are defined here as “significance metrics”.

Table 5 Calculation of DTI Significance Metric for Model DTI-3

Parameter	Additions to Metric	Summed metric
P4_y	1 + ¼ + ¼	1.5
P3_y	½ + ½ + 1/8	1.125
L5b	1/8 + 1/8 + 1/8	0.375
L10	¼	0.25
L5a	¼	0.25
P4_x	1/8 + 1/8	0.25

Table 5 shows how the significance metric for DTI-3 is computed. The parameters are weighted according to the proximity to the root node, which is the node at which the entire tree branches from (node 0 in Figure 3). The parameter used in the logical condition at the root node is afforded a weighting of 1, at the next level a weighting of ½, at the next level a weighting of ¼, and so on until the leaf nodes are reached. These weightings are summed across all occurrences of each parameter within the tree. This score is qualitative, rather than quantitative, hence the ranking of the parameters is of greater interest than the specific scores.

Information from ANN Models

The method of information extraction for the ANN models follows similar lines to the analytical model, using a sensitivity analysis to extract information. The method of sensitivity analysis used within this research is that which is contained within the Clementine DM software package, for which greater detail is given by Watkins [13]. The algorithm considers each input parameter in turn, and identifies the maximum and minimum values that are seen for that parameter within the dataset. Three further values are computed at 25%, 50% and 75% of the range between the minimum and maximum. Each instance within the dataset is then passed through the network in turn, and the parameter in question is held at each of the pre-computed 5 values and the output of the network recorded at each of these 5 values. The variation in output seen across these 5 values, in effect the greatest difference between the 5 output values, is recorded. This process is repeated using each instance in the dataset. The variations in output across all of the instances is then summed and normalised (in effect computing the average variance across all instances). This process is then repeated for each parameter in turn.

5. Comparison of Information

The information from the analytical model sensitivity analysis had been successfully used within industry, and thus was used as a benchmark. In order to present these results, a greyscale is used to indicate the ranking of each parameter, with parameters most influential upon system performance being located at the top of the list. The shade of gray attached to parameters in the ANN and DTI information columns correlate to the greyscale used in the analytical model information column.

Table 6 Comparison of Information Extracted from Analytical and DM Models

Deterministic Model Sensitivity Analysis		ANN Sensitivity Analysis		DTI Significance Metric	
L5b		P4_y		P4_y	
L3		P3_y		P3_y	
L4		L5b		L5b	
L2		P4_x		P4_x	
L3a		P3_x		L10	
L7a		L3		L5	

Key					
Higher rank in analytical model			Lower Rank in analytical model		

Table 6 shows the comparison of extracted information, where the greyscale discussed previously is used to highlight correlations in the rankings. It can be seen that the pivot points are most influential in the DM model rankings, but do not appear in the analytical model rankings. It is not clear why this should be the case, it is suggested that the manner in which the constraint modeller SWORDS fulfils constraints might have an effect but investigation of this has not yielded any results. It is noted that both the ANN and DTI models are prone to this anomaly, and this consistency suggests that both are describing a genuine relationship within the data, further suggesting that the root of the problem might lie in the manner in which the constraint modeller generated the data for DM modelling.

Table 7 Comparison of Information Extracted from Analytical and DM Models Excluding Pivot Points

Deterministic Model Sensitivity Analysis		ANN Sensitivity Analysis		DTI Significance Metric	
L5b		L5b		L5b	
L3		L3		L10	
L4		L2		L5	
L2		L1		L7	
L3a		L6		L11	
L7a		L9		L9	

Key					
Higher rank in analytical model			Lower Rank in analytical model		

Table 7 shows the comparison of extracted information when the pivot point information is removed from the ranked lists for both the DTI and ANN models. It can be seen that the information offers much greater correlation, where the same parameter is listed as most influential upon system performance in all 3 rankings. The ANN model ranking correlates with the analytical model ranking for the second and third most significant parameters, however the DTI ranking does not feature any of the remaining top 6 seen on the analytical model ranking.

Consideration of Table 2 suggests that the sensitivity analysis metrics for parameters placed 4^{th} or lower in the analytical model rankings are very similar, ranging fro 5.6 to 9.2 over 7 places as opposed to 14.8 to 52 over the top 3 rankings. This suggests that there is a minimal difference in influence between the 3^{rd} through 10^{th} ranked parameters, where such lack of clear distinction perhaps explains the lack of correlation between the ANN model ranking and the analytical model ranking.

It is suggested that the method of information extraction from the DTI model requires further consideration, where only the most significant parameter was seen to correlate with the analytical model ranking. It is suggested that reconsideration of the weightings attached to parameters at each level of the tree might improve the veracity of the extracted information. It is further suggested that the situation might be exacerbated by the relative instability of DTI algorithms, where small changes to the dataset or algorithm settings can lead to vastly different model structures due to the recursive construction of the model. In essence, any changes at the higher level nodes can result in notably different structure to the lower levels of the tree, and in cases where the information gain is similar for numerous possible logical conditions, it is feasible that changes to the data sample ort to algorithm parameter might act to significantly change the structure of the model and hence the extracted information.

Efforts to address this instability have been trialled, where the rationale of methods of aggregating predictions from DM models have been applied to aggregating extracted information. It is anticipated that these results will be published in due course.

6. Conclusion

A sensitivity analysis of an analytical model provided industrially-validated information regarding the ranking of parameters most influential upon system performance. This analytical mode was also used to generate data representing manufacturing data, where the effect random variation within tolerance of the parameters of a product could be related the performance of that product or system.

This data was used to generate a range of DTI and ANN models, from which one model of both types was selected for use in providing information. Such information was extracted from the ANN model via sensitivity analysis and from the DTI model via consideration of the structure of the constructed tree. This information was presented in the form of a ranked list, indicating the parameters most influential upon system performance.

This information was compared against information from the benchmark analytical model. Three of the top four ranked parameters were consistent across the information from the ANN model and the analytical model, and the top ranked parameter was consistent for both the DTI and analytical model. It was suggested that the instability of the DTI model and the relatively unrefined method of information extraction from this model prevented the correlation from being as good as that for the ANN model.

Although applied to sensitivity analysis, the methodological analysis described in this paper shows both how a DM approach can be used to represent and create models of engineering systems, with emphasis upon the analysis of manufacturing data, and that such models do not require recourse to the use of analytical computational modelling approaches. Methods of creating and interrogating DM models has been proposed, and the next stage of work will seek to apply these methods to the analysis of genuine manufacturing data.

References

[1] Taguchi, G., "Introduction to Quality Engineering: Designing Quality into Products and Processes". Productivity Pr., Tokyo, 1986

[2] Hicks, C. R. and Turner jr, K. V., "Fundamental Concepts in the Design of Experiments". Oxford University Press, New York, 1999

[3] Reich, Y. and Barai, S. V., "A methodology for building neural networks models from empirical engineering data", Engineering Applications of Artificial Intelligence, Vol. 13(2000, p. 685-694

[4] Kenney, L. P. J., Rentoul, A. H., Twyman, B. R., Kerr, D. R. and Mullineux, G., "A Software Environment for Conceptual Mechanism Design", Proceedings of the IMechE Part C - Journal of Mech. Eng. Science, Vol. 211(1997, p. 617-625

[5] Microsoft, "Microsoft Excel user's guide version 5.0". Microsoft Corporation, Redmond, USA, 1994

[6] Witten, I. H. and Frank, E., "Data Mining: Practical machine learning tools with Java implementations". Morgan Kaufmann, 2000

[7] Fu, L., "Neural Networks in Computer Intelligence". MacGraw-Hill, Singapore, 1994

[8] Quinlan, J. R., "Induction of Decision Trees", Machine Learning, Vol. 1(1986, p. 81-106

[9] Rumelhart, D. E., Hinton, G. E. and Williams, R. J.,Rumelhart, D. E. and McClelland, J. L., "Learning Internal Representations by Error Propogation", MIT Press, Cambridge, MA, 1986

[10] SPSS, "Clementine 7.0 User's Guide". Integral Solutions Limited, Chicago, 2002

[11] Maniezzo, V., "Genetic evolution of the topology and weight distribution of neural networks", IEEE Transactions on Neural Networks, Vol. 5(1), 1994, p. 39-53

[12] Reich, Y. and Barai, S. V., "Evaluating Machine Learning Models for Engineering Problems", Artificial Intelligence in Engineering, Vol. 13(1999, p. 257-272

[13] Watkins, D.,"Clementine's Neural Networks Technical Overview", SPSS, Woking, UK, 1997

M.D.Giess
University of Bath
Innovative Manufacturing Research Centre
Faculty of Engineering and Design
Claverton Down
Bath
BaNES

BA2 7AY
UK
Tel: +44 (0) 1225 386131
Fax: +44 (0) 1225 386928
email : m.d.giess@bath.ac.uk