

Optimal Input Selection for Neural Fuzzy Modelling With Application to Charpy Energy Prediction

Yong Y. Yang, Mahdi Mahfouf, Qian Zhang

Department of Automatic Control and Systems Engineering

IMMPETUS, The University of Sheffield

Mappin Street, Sheffield S1 3JD, United Kingdom

y.y.yang@sheffield.ac.uk

Abstract— Input variables selection plays a critical role in data-driven modelling, especially for complex systems with high dimensionality between the input/output space. In this paper, a new artificial neural network based forward input selection scheme is proposed. The objective of the proposed scheme is to select the smallest number of important variables as model inputs, which will then be used for neural-fuzzy data modelling. The proposed input selection scheme is applied to a case study of Charpy impact energy prediction, with data extracted from an industrial database. Model performance has been compared with previous results where a much larger input set was used. Simulation results show that the number of inputs for the Charpy data model can be significantly reduced with little performance degradation. Also, the performance of the proposed scheme outperforms both the standard correlation analysis and fuzzy clustering based input selection schemes

Keywords: *Model input selection; fuzzy clustering; neural network; neural fuzzy system; Charpy impact energy.*

I. INTRODUCTION

Data-driven modelling is becoming more popular due to the rapid growth in computing power, the wide availability of data/information, and the readily available software packages where different data modelling strategies can be readily implemented and tested. Modelling tools and algorithms have also evolved into more complex structures capable of dealing with highly non-linear and multi-dimensional input/output data. However, in many real-world modelling scenarios, the available data often contain hundreds or even thousands variables, and the influences of many variables on associated outputs are not well understood. For such complex modelling tasks, a heavy-handed input selection method is essential, so that the dimension of the final model input space can be reduced to a manageable size. Due to the increasing demand for modelling large scale data, as is almost always the case in data mining [1], research in model input selection methodologies is becoming a hot topic area. Artificial neural network (NN) is a very popular model framework, due to the existence of well-established error back-propagation training algorithms and its capability of being a universal approximator for complex nonlinear mapping [2]. As the computational intensity and the number of training samples required increase

exponentially with the input space dimension, applications of NN model in very high dimensional problems quickly become prohibitive. Several methods have already been investigated for input selection in the past, aimed at assisting NN modelling for a large number of potential inputs. A concise review can be found in [3], with the focus being on input selection methods using mutual information criterion and genetic algorithms. Fuzzy logic systems are another popular modelling paradigm, and a variety of related input selection schemes have been developed [4-5]. It is even more important to use a small input set in fuzzy modelling than NN, as one of the key advantage of transparency related to a fuzzy logic system will be lost even when the number of inputs is moderate. Feature transformation [1, 6], including principal component analysis and independent component analysis, is often very effective in order to reduce the nominal number of inputs, but one needs to be reminded of the fact that the individual features thus formatted involve a combination of the original variables, and the physical interpretation of such features may not be easy. Input selection via feature transformation is not considered further in this paper.

In this paper, an NN-based forward input selection scheme is proposed. The input selection scheme belongs to the category of model based direct search [5], and it starts by selecting only one input from all the potential inputs such that the corresponding model trained on the selected input produces the best model performance when compared to all other NN models with a different single input. The first input identified is thus ranked as the most important input variable, and will always be included in the subsequent enlarged input set. As an exhaustive search of input combinations leads to an exponential growth with the number of potential inputs in computation, hence it is infeasible for a search particularly involved with a large number of variables. The forward input selection scheme provides a pragmatic yet an effective solution to find the smallest number of important variables, and upon which a compact data model can be developed. The remainder of the paper is organized as follows. Section 2 introduces the aim, objective and motivation for input selection in neural-fuzzy data modelling, and describes two commonly used input selection methods based on correlation analysis and fuzzy importance ranking. In Section 3, an NN-based forward input

selection scheme is developed, together with the key techniques for the implementation of the forward input selection scheme. Section 4 is devoted to the application of the proposed input selection scheme to a case study problem of Charpy impact energy data modelling, with the data extracted from an industrial database. The preliminary results of the developed model are compared to models derived from other input configuration schemes, including correlation coefficient analysis, fuzzy importance ranking, and to some previously developed models using the full input data. Finally, Section 5 concludes the paper with suggestions for future research.

II. SOME COMMON INPUT SELECTION METHODS

Input variable selection is a vital step in complex system modelling where the input–output relationship is often not well understood. An unnecessarily large number of inputs can be detrimental for effective data modelling, not only that the high dimensional space will cause difficulties due to ‘the curse of dimensionality’, but also the irrelevant inputs tend to confuse the true input-output relationships. Moreover, it is well-known that the number of data examples needed to tune a model grows exponentially with the dimension of the input space, and the resource required for collecting an increasing large data set with good quality is very expensive. As pointed out in [7], the objective of input variable selection is three-fold: improving the model prediction performance, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data in the first place.

Most of human experience is limited to low-dimensional spaces, partly due to the fact that we live in a low 3D space. A large degree of unease will be felt if humans are confronted with a high dimensional space, even if it is slightly larger than 3D. Our visualization ability and degree of imagination will easily run out of limit when facing a higher dimensional space, and many properties of these high dimensional spaces often appear to be counter-intuitive. Smits et al [8], for example, summarised some of these counter-intuitive properties: (i) samples sizes with the same data density increase exponentially with dimensionality; (ii) an increasingly large radius is needed to enclose a given fraction of data points even when the data are uniformly distributed; (iii) most data points are closer to an edge than to another internal data point, which indicates extrapolation is the norm rather than the exception in higher dimensional space; and (iv) almost every point is an outlier in its own projection on the line defined by the prediction point and the origin. In the following, two common input selection methods based on correlation analysis and fuzzy clustering ranking are briefly introduced, and later they will be compared with the NN-based forward input selection scheme to be developed in Section III.

A. Input Selection Based on Correlation Analysis

The correlation coefficient analysis can prove sufficient enough to help identify the input/output dependency if the process under investigation is linear [3]. Even if the system relating the original data is nonlinear, correlation analysis is often worthwhile as it can help to reach a better understanding

of the data to be modelled. The Pearson’s correlation coefficient can be calculated via the following equation:

$$\rho_{ij} = \frac{\sum_{k=1}^N (x_i(k) - \bar{x}_i)(x_j(k) - \bar{x}_j)}{\sqrt{\sum_{k=1}^N (x_i(k) - \bar{x}_i)^2 \sum_{k=1}^N (x_j(k) - \bar{x}_j)^2}} \quad (1)$$

where $\rho_{ij} \in [-1, 1]$ is the correlation coefficient between variable x_i and x_j , $x_j(k)$ is the value of the k th sample, N is the total number of samples, and $\bar{x}_i$ is the corresponding sample mean. When one of the variables is an output, ρ_{ij} represents the linear dependency between the output and the given input, otherwise it indicates the co-linearity between the two involved inputs.

A compact form to summarize all the correlation coefficients is a correlation matrix, which is particularly useful when a large amount of variables are involved. A correlation matrix $\mathbf{R}$ is defined as follows:

$$\mathbf{R} = [\rho_{ij}]_{n \times n} = \begin{pmatrix} \rho_{11} & \cdots & \rho_{1n} \\ \vdots & \ddots & \vdots \\ \rho_{n1} & \cdots & \rho_{nn} \end{pmatrix} \quad (2)$$

Where ρ_{ij} is the correlation coefficient defined in (1), n is the total number of data variables. $\mathbf{R}$ is a symmetric matrix with $\rho_{ij} = \rho_{ji}$, and all the diagonal elements $\rho_{ii} = 1$.

The full correlation matrix provides one with the mutual linear dependency among each pair of variables. A large $|\rho_{ij}|$ value represents a strong linear correlation, hence a large dependency between the two variable i and j . As such, the application of correlation analysis for model input selection can be of double fold. First, the dominant correlation coefficients among the potential input variables are identified and they can then be used to detect the redundant input variables, which can and need to be eliminated to reduce the potential input space dimension. Second, the large correlation coefficients among the input variables and the outputs can be used to identify those variables to be selected as the important model inputs, while these variables corresponding to insignificant correlation coefficients are removed from the model input consideration. It should be noted here that the correlation coefficients defined in (1-2) only indicate the linear dependency between the involved variables, while no causal relationship can be inferred. Table I lists the input-output correlation coefficients obtained from the Charpy impact energy data, ordered in columns by the magnitude. The full correlation matrix also reveals that there is no severe redundancy among the 16 original input variables, with the most dominant correlation coefficients being $\rho_{12,13} = 0.59$, $\rho_{8,9} = 0.53$, $\rho_{7,8} = 0.48$, and $\rho_{14,16} = 0.46$. The association of the variable x_i , $i = 1, 2, \dots, 16$ and its physical name is given in Table IV.

TABLE I. THE INPUT-OUTPUT CORRELATION COEFFICIENTS

x_i	ρ	x_i	ρ	x_i	ρ	x_i	ρ
x_{15}	0.38	x_{10}	-0.19	x_{12}	0.12	x_{11}	0.02
x_4	-0.28	x_6	0.18	x_1	-0.12	x_5	0.02
x_{14}	0.24	x_2	-0.15	x_3	0.08	x_7	0.01
x_{13}	0.20	x_{16}	0.13	x_9	0.08	x_8	0.00

B. Input Selection via Fuzzy Clustering

There exist many different methods of using fuzzy logic for input selection [4], including fuzzy c-means clustering, optimal Mahalanobis distances, ‘fuzzy curves’ method, etc. In this paper, the input selection method presented in [9] has been adapted for identifying the important input variables. An initial fuzzy model was first generated through the competitive clustering method using all possible m variables as the initial inputs. The initial fuzzy model has p fuzzy rules, which best divide the full model data into p clusters centred at $c_i \in R^m$ ($i = 1, 2, \dots, p$). The fuzzy model structure used for input selection is shown in Fig. 1, where ϕ_i is the membership function set (consists of p members) related to the i th input variable, and Rz_i is the range of z_i which represents the output change attributed to the i th input x_i taking values from across the corresponding centre locations in $\{c_{ki}, k = 1, 2, \dots, p\}$.

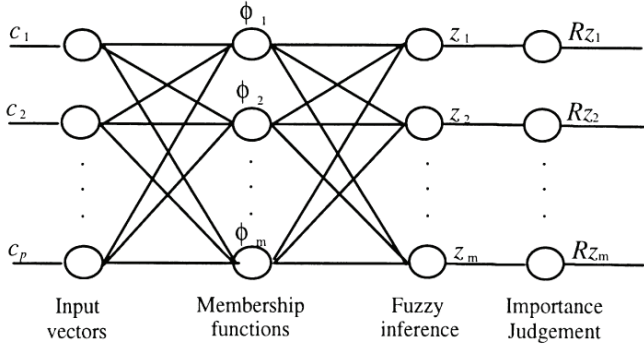


Figure 1. Fuzzy model structure for input variable selection [9]

The basic idea of ranking the importance of each input variable from the above initial model is based on calculating the contribution of individual input variations to the model output. The larger the output change caused by a specific input variable variations quantifying by its central locations of in the p clusters, the more important this input variable is. The fuzzy model shown in Fig. 1 provides an effective mechanism to test the importance of individual input variables. If all antecedent clauses except the one associated with a particular input x_i under scrutinizing are kept constant, the resulting fuzzy model output variations can be captured in the specific output route Rz_i while all the other output routes $Rz_j, j \neq i$, will be constant. Hence, with the inputs including the full variations specified by the cluster centers $\{c_k, k = 1, 2, \dots, p\}$, output variations for each input variable can be extracted from the corresponding output nodes $Rz_1, Rz_2, \dots, Rz_m$ in Fig. 1. An importance factor designed to provide criteria for input variable selection is based on the magnitude of output variations attributed to individual inputs, defined as follows:

$$F_i = Rz_i / \max(Rz_1, Rz_2, \dots, Rz_m) \quad (3)$$

Where $F_i \in (0, 1]$ is the importance factor for the i th variable x_i , and $Rz_j, j = 1, 2, \dots, m$ are the output variation range associated with the corresponding input x_j variations. The x_i associated with $F_i = 1$ is the most important input variable.

Two important design parameters, the cluster radius r_c and the membership width σ_c , are responsible for controlling the initial fuzzy partition of the full input data space, and hence

determine the initial fuzzy rule structure. These parameters need to be tuned carefully, as they directly influence the quality of input variable ranking. For the Charpy impact energy data after being normalized into a high dimensional hyper-cube, these two design parameters are selected as $r_c = 0.7$ and $\sigma_c = 0.2$. The initial fuzzy model obtained contains 36 fuzzy rules, and the importance factor for all the original input variables are listed in Table II, ordered in columns according the value of the associated importance factor F_i .

TABLE II. THE INPUT IMPORTANCE FACTOR

x_i	F_i	x_i	F_i	x_i	F_i	x_i	F_i
x_9	1.00	x_8	0.59	x_2	0.40	x_{10}	0.26
x_5	0.85	x_{15}	0.58	x_{16}	0.31	x_{14}	0.13
x_4	0.84	x_{12}	0.56	x_{11}	0.30	x_3	0.13
x_{13}	0.60	x_1	0.53	x_7	0.26	x_6	0.10

III. NN-BASED FORWARD INPUT SELECTION

A. NN-Based Forward Input Selection

An artificial neural network model based forward selection procedure is proposed here to assist identifying the most important input variables for data-driven modelling. A 3-layer feed-forward neural network is adopted as the base model, and its output can be expressed as follows [10]:

$$\hat{y}(k) = \sum_{j=1}^M w_j \phi_j \left(\sum_{i=1}^n w_{ij} x_i(k) + w_{j0} \right) + w_0 \quad (3)$$

where $\hat{y}(k)$ is the NN output prediction for the k th sample, w_j and w_{ji} are the relevant NN weighting coefficients at the output layer and hidden layer, respectively, w_0 and w_{j0} are the bias of the output layer and hidden layer, respectively, ϕ_j is the nonlinear activate function of the j th hidden neuron, n is the dimension of the NN input space, and M is the total number of hidden neurons.

A forward input selection procedure is proposed with the above NN model as a performance evaluator. The input selection is implemented through an iterative process, starting from selecting a single input for the NN modelling. All the potential inputs are sampled as the single input candidate, and the neural network model with the minimum root mean squared error (RMSE) is identified as NN*. The inputs corresponding to the ones used in NN* are ranked as the most important inputs identified so far. In the next iteration of input selection, the inputs of the NN is increased by one from the identified input set, with the newly increased input selected from all remaining potential input candidates. The above forward input selection procedure is repeated until either a sufficient number of inputs identified or no performance improvement achieved by adding more inputs. The NN training procedure used here is adapted from the one used in [11], with critical training parameters fixed via trial-and-error tuning. The number of hidden neurons is determined by the dimension of the input using heuristic rules, as optimisation here will be too computationally cost. The iterative forward input selection algorithm is summarized as follows:

Step i: Initialisation. Set $\mathbf{x}^* = \{\}$ and n_{max} - the maximum number of inputs allowed. Put all the variables into the potential input candidate set $\mathbf{X} = [x_1, x_2, \dots, x_N]$, where N is the maximum number of potential input variables.

Step ii: Start a new iteration of input selection. For each of the potential input candidate $x_j \in \mathbf{X}$ perform neural network modelling with extended input $\mathbf{x}^j = [\mathbf{x}^*, x_j]$ and $H = \min(5, 2 \cdot \dim(\mathbf{x}^j))$. Compute the corresponding NN model performance $J(\mathbf{x}^j)$, $j = \{1, 2, \dots, \dim(\mathbf{X})\}$. In this step, all the NN models has the same number of inputs $n = \dim(\mathbf{x}^*) + 1$, and there is a total number of $N - \dim(\mathbf{x}^*)$ neural network models trained.

Step iii: Identify the optimal inputs obtained so far. Determine the optimal inputs in the current iteration $\mathbf{x}^* = \mathbf{x}^{j^*}$, where j^* is determined by using the criterion $j^* = \text{argmin}_j J(\mathbf{x}^j)$. Update the potential input set $\mathbf{X}$ by eliminating the associated x_{j^*} ($\mathbf{X} = \mathbf{X} \setminus x_{j^*}$). Notice here that the dimension of the potential input candidate set is reduced by one in each iteration.

Step iv: Termination check. If the best NN performance in the current round $J(\mathbf{x}^*)$ does not improve over the previous one, or if $\dim(\mathbf{x}^*) \geq n_{max}$, or if $\mathbf{X}$ is empty, then go to Step v, otherwise return to Step ii for the next iteration of optimal input selection.

Step v: Final input ranking. The elements in $\mathbf{x}^*$ are the most important input variables identified by the NN forward selection procedure. The importance ranking is the order of inclusion of the elements in $\mathbf{x}^*$.

Table III summarises the input variable ranking results for the Charpy impact energy data using the NN-based forward input selection procedure. Due to its powerful input-output mapping capability, a NN-based framework can deal with nonlinear behaviour in the data set effectively. However, the computations involved in this NN-based forward input ranking are rather heavy, especially when a large high dimensional data set is involved. For the Charpy data input ranking task with a moderate 16 potential inputs, a total number of 120 neural networks need to be trained via some kind of error back-propagation algorithm. The procedure may meet some difficulties for ranking very high dimensional input space, as not only the number of neural networks need to be trained will be much larger, but also the training for each individual NN with a large number of inputs will be computing and resource intensive.

TABLE III. INPUT RANKING USING THE NN FORWARD SELECTION

Rank	x_i	Rank	x_i	Rank	x_i	Rank	x_i
1	x_{15}	5	x_6	9	x_8	13	x_5
2	x_2	6	x_7	10	x_{11}	14	x_{12}
3	x_4	7	x_1	11	x_{10}	15	x_{14}
4	x_{16}	8	x_3	12	x_{13}	16	x_9

B. Neural Fuzzy Modelling with Reduced Input Variables

Neural networks and adaptive fuzzy systems are two widely used data modelling paradigms, due to the existing of well-established training algorithms and their ability as universal approximator [11, 12]. Both models can approximate any continuous non-linear function with arbitrary accuracy, provided that enough hidden neurons or fuzzy rules have been used. After the optimal model inputs with significantly reduced dimension has been identified, a final data model can be developed using an appropriate neural fuzzy modelling framework. In this paper, a feed-forward neural network is chosen for the final stage of data modelling. The number of hidden neurons H , which is the most critical NN structure parameter, is optimised through an interactive search method. Fig. 2 shows the iterative NN training strategy incorporating structure optimisation, implemented using the built-in training utilities provided by the MATLAB Neural Network toolbox [13]. The search range for the hidden neurons $[H_{min} \sim H_{max}]$ for the NN structure optimisation should be tuned according to the complexity of the data, as well as the dimension of the input space. The Levenberg-Marquardt training algorithm [14-15] is adopted to carry out the error back-propagation, and an earlier training stopping option is also added within the NN training routine in order to prevent the NN model from being over-fitting.

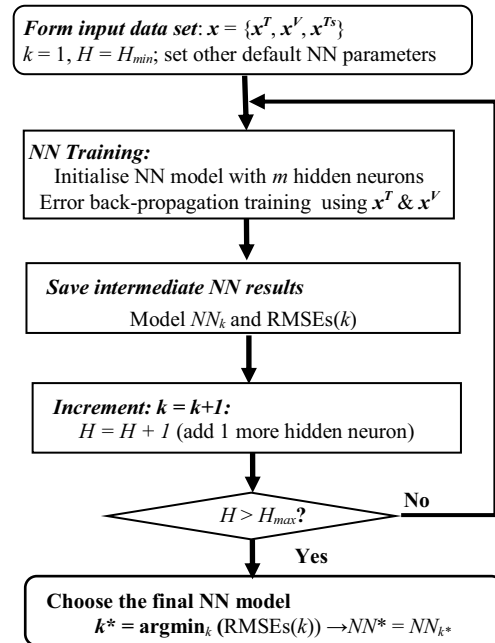


Figure 2. An iterative NN training scheme with structure optimisation

IV. CASE STUDY - CHARPY IMPACT DATA MODELLING

A. Overview of the Charpy Impact Energy Data

The Charpy impact energy data, together with its relevant input variables, were extracted from an industrial database pertaining to TATA Steel Europe. The data set consists of 1661 records, each with measurements of the steel chemical compositions, heat treatment specifications, and Charpy impact

test conditions. Some important statistics of the Charpy data are summarised in Table IV.

TABLE IV. CHARPY IMPACT DATA STATISTICS

x_i	variable name	min - max	mean	σ (std.)
x_1	Test Depth (mm)	5.5 - 146.0	20.8	14.5
x_2	Specimen Size (mm)	11 - 381	172.5	80.8
x_3	Test Site (coded)	2 - 6	3.80	1.12
x_4	C (% wt)	0.13 - 0.52	0.394	0.058
x_5	Si (% wt)	0.11 - 0.38	0.255	0.032
x_6	Mn (% wt)	0.41 - 1.75	0.841	0.217
x_7	S (% wt)	0.001 - 0.052	0.019	0.009
x_8	Cr (% wt)	0.11 - 3.25	1.075	0.245
x_9	Mo (% wt)	0.02 - 0.98	0.239	0.086
x_{10}	Ni (% wt)	0.03 - 4.21	0.368	0.519
x_{11}	Al (% wt)	0.003 - 0.047	0.027	0.005
x_{12}	V (% wt)	0.001 - 0.260	0.009	0.029
x_{13}	Hardening Temp.	810 - 980	865.1	17.1
x_{14}	Cool Medium	1 - 3	2.8	0.482
x_{15}	Temper Temp. (C)	450 - 730	651.6	49.2
x_{16}	Testing Temp. (C)	-59 - 23	-8.0	26.0
y (output)	Impact Energy (J)	6.1 - 241.3	89.70	6.13

It is known that the above Charpy impact energy data are difficult to model, due to the existence of a large scatter in the measurements, the poor repeatability of the Charpy impact test and the highly sparse sample distribution. Several previous studies have been conducted on the modelling of this data set, where the focus is on model performance improvement by using advanced modelling strategies [16-19]. The work presented in this paper represents a first attempt aimed at further model input optimisation.

B. NN Modelling Results for Reduced Inputs

The reduced input sets containing 9 variables are selected from the original 16 input variables according to the different input selection methods discussed earlier, and are used as the final inputs for the Charpy impact energy data modelling. The reduced input Charpy data are then divided into three data subsets, for training (x^t), validation (x^v) and testing (x^s). The data ratio of x^t : x^v : x^s is set to 60:25:15, identical to most of the previous modelling work so that the performances are directly comparable. The maximum number of iterations (epochs) for the inner NN training is set to 200, and the training algorithm is fixed to the Levenberg-Marquardt algorithm. The search range for the hidden neurons is set to $H_{min} = 5$ and $H_{max} = 20$. Some typical results obtained via the iterative NN modelling scheme as shown in Fig. 2 are presented below.

According to the NN-based forward input selection scheme (see Table III), the reduced model input set x_N^* is as follows:

$$x_N^* = [x_{15}, x_2, x_4, x_{16}, x_6, x_7, x_1, x_3, x_8]^T \quad (4)$$

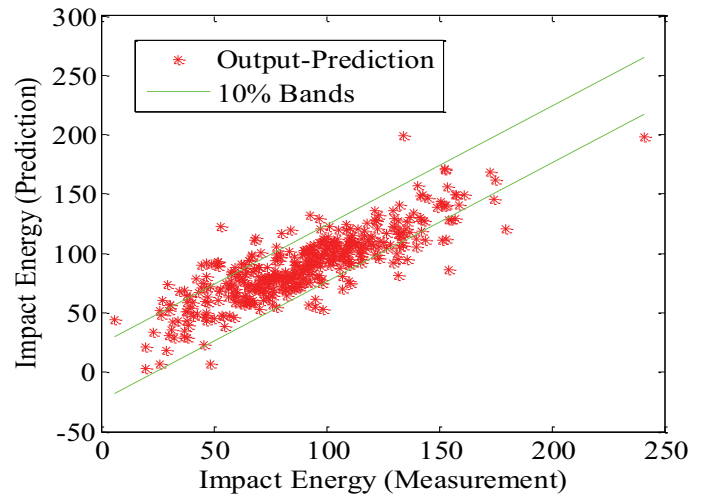
The optimal NN model trained from x_N^* has $m^* = 12$ hidden neurons, and the model performance achieved is $RMSE(T, V, Ts) = (15.52, 18.36, 19.83)$. Fig. 3 shows the NN model predictions of the Charpy impact energy against the measurements for the validation and testing data set. Prediction

accuracy is better for the training data set, which is not shown due to the constraints in space.

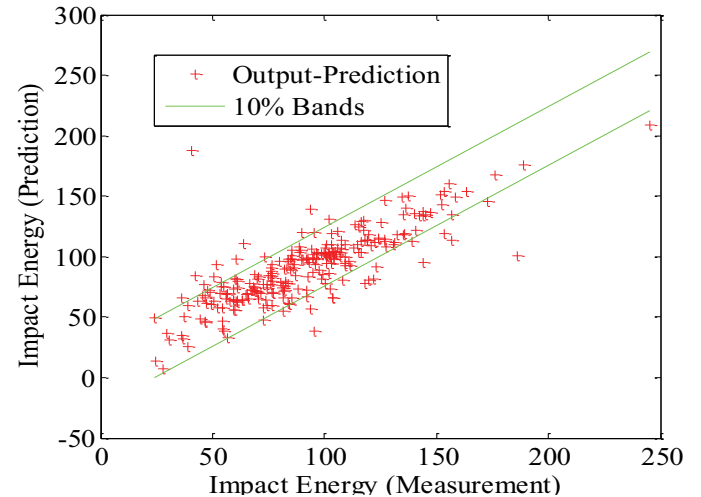
Similar modelling practices have also been carried out for the reduced model inputs selected using the correlation coefficient criterion and the fuzzy input importance ranking criterion. The corresponding constituents for the reduced model input sets, according to Tables I-II, are given as follows:

$$\begin{aligned} x_F^* &= [x_9, x_5, x_4, x_{13}, x_8, x_{15}, x_{12}, x_1, x_2]^T \\ x_C^* &= [x_{15}, x_4, x_{14}, x_{13}, x_{10}, x_6, x_2, x_{16}, x_{12}]^T \end{aligned} \quad (5)$$

Where x_F^* is the reduced input set selected using the fuzzy input importance ranking, and x_C^* is the reduced inputs obtained by the input-output correlation. The optimised NN model trained using x_F^* produces a $RMSE(T, V, Ts) = (18.84, 23.47, 27.49)$, with $H^* = 15$ hidden neurons. Similarly, the NN model generated based on x_C^* has a $RMSE(T, V, Ts) = (15.57, 20.99, 20.53)$, with $H^* = 16$. The NN predictions for the testing data set from these two models are shown in Fig. 4, and Table V summarises the RMSE performance comparisons, including performances from some previously developed models [17-19].

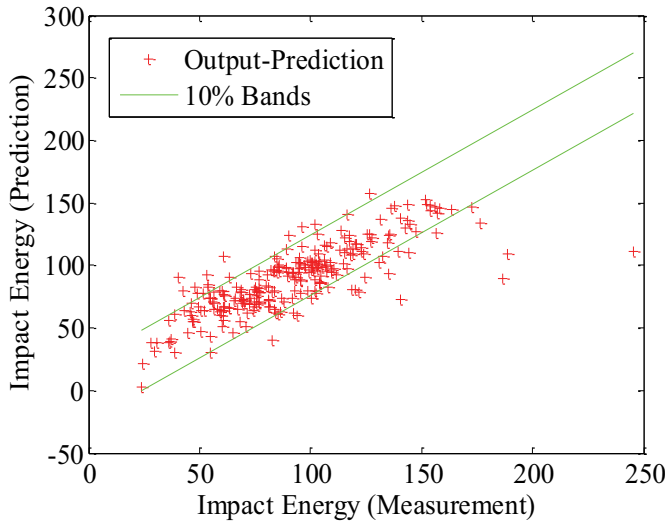


(a) For the validation data set x^v

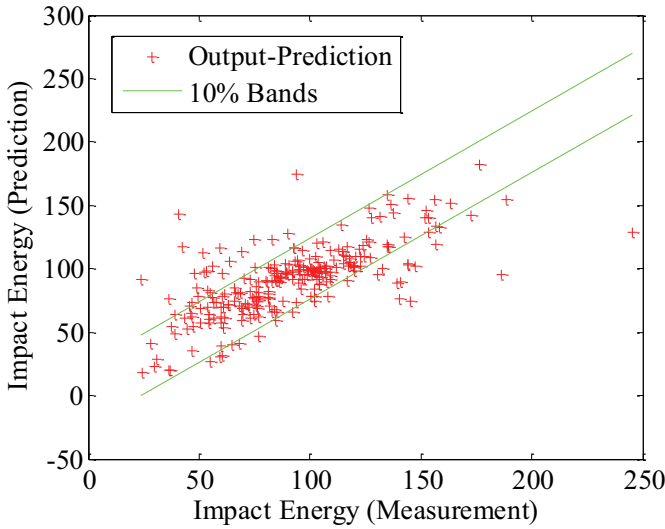


(b) For the testing data set x^s

Figure 3. NN (using x_N^*) predictions for Charpy impact energy



(a) NN model using correlation based inputs x_c^*



(b) NN model based on fuzzy ranking inputs x_f^*

Figure 4. NN predictions of the Charpy impact energy (testing data)

TABLE V. RMSE PERFORMANCE COMPARISONS

Modelling Methods	Number of Inputs	Train Error	Validation Error	Test Error
Neural-Fuzzy [17]	16	16.7	-	20.40
GrC – NF [18]	16	14.66	21.24	20.42
GA – NN [19]	16	14.32	17.94	18.96
NN – Corr. Inputs	9	15.57	20.99	20.53
NN – Fuzzy Inputs	9	18.84	23.47	27.49
NN – NN Inputs	9	15.52	18.36	19.83

C. Brief Discussions

The NN based forward input selection scheme performs best among the three input selection methods considered in this paper. Considering the large uncertainty involved in the Charpy impact energy measurements, the RMSE error performance of below 20 achieved by the proposed modelling scheme is good.

The reduced input set identified by the correlation analysis leads to a better NN model than that of using fuzzy ranking input selection. This may indicate that a more careful tuning is needed when complicated methodology such as the fuzzy input ranking is to be used.

The model performance using a reduced 9-input obtained via NN-based forward input selection scheme is in par with the previous developed models where the full 16 input variables were used. The NN-based forward input selection scheme outperforms the Neural-Fuzzy model being reported in [7]. It also produces better RMSE performances for the validation and testing data sets, as compared to the Granular Computing Neural Fuzzy (GrC-NF) model [18], but a slightly worse RMSE value for the training data set. The performances for the validation and testing data are deemed to be more important than that of the training data, due to the possible model over-fitting phenomenon. The GA-NN model, with the structure and the parameters optimised using a GA-based algorithm, did produce an overall better RMSE performance than the current reduced input model, but the difference is not significant. It may be argued that the significant reduction of input dimension (from 16 to 9), may be worth the slight increase in RMSE.

V. CONCLUSIONS

In this paper, an NN-based forward input selection scheme is proposed to deal with the optimal input selection problem. The principle for selecting the most importance input variables relies on a direct search among the potential candidates for model performance optimisation. Input selection plays an important role in data-driven modelling in general, especially for complex systems with a large number of potential inputs. After the reduced input set is identified, it can be used in an appropriate modelling scheme for the final data model development. In this paper, a feed-forward NN model framework is employed to deliver the final data model. The integrated model-input selection and model optimisation are implemented using the MATLAB Neural Network toolbox, with the training guided by an error back-propagation algorithm. The number of hidden neurons, a critical parameter determining the NN structure, is also optimised in the final NN modelling stage. The model prediction performance, in terms of RMSE, is then compared to two other input selection methods, one based on correlation analysis, and the other based on fuzzy input ranking. Preliminary results obtained in the case study for Charpy impact energy prediction show that the NN-based forward input selection method produces the best overall model performance among the three input selection methods compared.

The model performance obtained from the NN-based forward input selection is also compared with the performances of some previous developed Charpy impact energy prediction models, including the neural-fuzzy model, the granular computing neural fuzzy model, and the NN model with structure optimised via a genetic algorithm. Even though all the previous models used the full 16 model inputs, the NN model developed in this paper with a reduced 9 inputs managed to outperform two out of three previous models, while a slight RMSE performance deterioration is recorded in the third case of full input NN model with genetic algorithm based structure

optimisation. However, the benefit of a much compact model, resulting from a smaller input space, should generally outweigh the small RMSE-based performance loss. In training data requirement, for example, 512 training samples will be needed to train for the reduced input model proposed in this paper assuming a training data density of 2 samples per input, while a huge number of 65536 training data samples will be required to provide the same training data density across the high input space of 16 dimensions. This is an overwhelming 128 fold increase in training data requirement from a 9-input model to a 16 input model. This shows one of the ‘counter intuitive’ properties of a high dimensional space: the effect of reducing the number of inputs is also exponential, albeit in the reverse direction.

One disadvantage of using the proposed NN-based forward input selection is that the computational resources required are still high, and may be prohibitive for very large scale data with a high dimension. Hence, designing novel fast input raking algorithms based on the forward input selection principle will be beneficial. Also, due to the intensive calculations involved in error back-propagation algorithm, developing a fast NN training algorithm will be worth the effort. We will investigate new methodologies relating to these fields in future research.

ACKNOWLEDGMENT

Financial support from the UK-EPSRC is acknowledged under Grant EP/F023464/1. The authors wish to thank Tata Steel Europe for providing the Charpy impact energy data used in this research as well as expert knowledge for the data pre-processing and modelling.

REFERENCES

- [1] I. H. Witten, and E. Frank, *Data Mining - Practical Machine Learning Tools and Techniques*, 2nd Ed., Morgan Kaufmann Publishers, San Francisco, 2005.
- [2] K. Hornik, “Approximation capabilities of multilayer feedforward networks”, *Neural Networks*, vol. 4, no. 2, pp. 251-257, 1991.
- [3] M. P. C. Couto, “Review of input determination techniques for neural network models based on mutual information and genetic algorithms”, *Neural Comput & Application*, vol. 18, pp. 891–901, 2009.
- [4] Y. Hou, J. M. Zurada, W. Karwowski, W. S. Marras, and K. Davis, “Identification of key variables using fuzzy average with fuzzy cluster distribution”, *IEEE Transactions On Fuzzy Systems*, vol. 15, no. 4, pp. 673-685, 2007.
- [5] R. Sindelár, and R. Babuska, “Input selection for nonlinear regression models”, *IEEE Transactions On Fuzzy Systems*, vol. 12, no. 5, pp. 688-696, 2004.
- [6] S. Theodoridis, and K. Koutroumbas, *Pattern Recognition*, 4th Ed., Elsevier Inc., 2009.
- [7] I. Guyon, and A. Elisseeff, “An introduction to variable and feature selection”, *Journal of Machine Learning Research*, vol. 3, pp. 1157-1182, 2003.
- [8] G. Smits, A. Kordon, K. Vladislavleva, E. Jordaán, and M. Kotanche, “Variable selection in industrial datasets using pareto genetic programming”, *International Congress on Modelling and Simulation MODSIM 05*, December 2005, Melbourne University, Vic.
- [9] D. A. Linkens, and M. Y. Chen, “Input selection and partition validation for fuzzy modelling using neural network”, *Fuzzy Sets and Systems*, vol. 107, pp. 299-308, 1999.
- [10] M. Mahfouf, and Y. Y. Yang, “A GA-optimised ensemble neural network model for charpy impact energy predictions”, *IFAC MMM2010*, The 13th Symposium on Automation in Mining, Mineral and Metal Industry, 2-4 August 2010, Cape Town, South Africa.
- [11] K. Hornik, “Approximation capabilities of multilayer feedforward networks”, *Neural Networks*, vol. 4, no. 2, pp. 251-257, 1991.
- [12] B. Kosko, “Fuzzy systems as universal approximators”, *IEEE Transactions on Computers*, vol. 43, no. 11, pp. 1329-1333, 1994.
- [13] The MathWorks Inc., *Neural Network Toolbox 7 User's Guide*, http://www.mathworks.com/help/pdf_doc/nnet/nnet.pdf, 2010.
- [14] C. M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford, pp. 290-292, 1995.
- [15] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, London, 2006.
- [16] J. Tenner, *Optimisation of the Heat Treatment of Steel using Neural Networks*, PhD Thesis, The University of Sheffield, 1999.
- [17] M. Y. Chen, and D. A. Linkens, “A systematic neural-fuzzy modelling framework with application to material properties of alloy steels”, *IEEE Transactions on System, Man and Cybernetics – Part B: Cybernetics*, vol. 31, no. 5, pp. 781-790, 2001.
- [18] G. Panoutsos, and M. Mahfouf, “Granular computing and evolutionary fuzzy modelling for mechanical properties of alloy steels”, *Proceeding of 16th IFAC World Congress*, Prague, Czech Republic, 4-8 July, 2005.
- [19] M. Mahfouf, Y. Y. Yang, and Q. Zhang, “Modelling Charpy impact energy of heat-treated-steel using efficient neural-networks and genetic-algorithms”, *IFAC MMM2009 Workshop on Automation in Mining, Mineral and Metal Industry*, 14-16 October 2009; Viña del Mar, Chile.