Data Mining using Inductive Modeling Approach

Godfrey C. Onwubolu

School of Engineering & Physics, University of the South Pacific, Private Bag, Fiji

Onwubolu_g@usp.ac.fj

Abstract. The rate at which organizations are acquiring data is getting out of proportion and managing such data so as to infer useful knowledge that can be put to use is increasingly becoming important and challenging. Data Mining (DM) is one such relatively recently technology that has emerged that is employed in inferring useful knowledge that can be put to used from a vast amount of data. This paper proposes a new design methodology which is a hybrid of differential evolution (DE) and Group Method of Data Handling (GMDH) for self-organizing data mining. The new hybrid implementation is applied to the data mining activity of prediction of soil moisture, which is an aspect of hydrology. Experimental results indicate that the proposed approach is useful for data mining technique for forecasting hydrological data.

Keywords:

Inductive modeling; Self-Organizing Data Mining, DE, GMDH

1 Introduction

Data Mining (DM) is an important component of the emerging field of knowledge discovery in databases (KDD). Large databases of digital information are ubiquitous. Current hardware and database technology allow efficient and inexpensive reliable data storage and access. With the exponential rate at which data is becoming available to user, one question that needs to be answered is, what else do we do with all the available data? This is where opportunities for KDD and consequently DM naturally arise. However, whether the context is business, medicine, science, engineering, or government, the datasets themselves in raw form are of little direct value. What is of value is the knowledge that can be inferred from the data and put to use.

The KDD process is well documented in the literature [2—7] and the outline of some of its basic steps is as follows:

1. Developing an understanding of the application domain and the relevant prior knowledge and identifying the goal of the KDD process from the customer's viewpoint.
2. Creating a target data set: selecting a data set, or focusing on a subset of variables or data samples, on which discovery is to be performed.
3. Data cleaning and preprocessing. Basic operations include removing noise if appropriate, collecting the necessary information to model or account for noise, deciding on strategies for handling missing data fields, and accounting for time-sequence information and known changes.
4. Data reduction and projection: finding useful features to represent the data depending on the goal of the task. With dimensionality reduction or transformation methods, the effective number of variables under consideration can be reduced, or invariant representations for the data can be found.
5. Matching the goals of the KDD process (step 1) to a particular data-mining method. For example, Fayyad [2—5] described the different methods of summarization, classification, regression, clustering, and so on.

6. Exploratory analysis and model and hypothesis selection: choosing the data mining algorithm(s) and selecting method(s) to be used for searching for data patterns. This process includes deciding which models and parameters might be appropriate (for example, models of categorical data are different than models of vectors over the real dataset) and matching a particular data-mining method with the overall criteria of the KDD process (for example, the end user might be more interested in understanding the model than its predictive capabilities).
7. Data mining: searching for patterns of interest in a particular representational form or a set of such representations, including classification rules or trees, regression, and clustering. The user can significantly aid the data-mining method by correctly performing the preceding steps.
8. Interpreting mined patterns, possibly returning to any of steps 1 through 7 for further iteration. This step can also involve visualization of the extracted patterns and models or visualization of the data given the extracted models.
9. Acting on the discovered knowledge: using the knowledge directly, incorporating the knowledge into another system for further action, or simply documenting it and reporting it to interested parties. This process also includes checking for and resolving potential conflicts with previously believed (or extracted) knowledge.

Similarly, the summary of the data mining process as is as follows [2—5]:
1. Data gathering, e.g., data warehousing.
2. Data cleansing: eliminate errors and/or bogus data.
3. Feature extraction: obtaining only the interesting attributes of the data.
4. Pattern extraction and discovery.
5. Visualization of the data.
6. Evaluation of results; not every discovered fact is useful, or even true; judgment is necessary before following the conclusions of the user's software.

A wide variety of data-mining model representation methods exist, but here, we only focus on a subset of popular techniques, which include *decision trees* and *rules* [8], [9], *linear models*, *nonlinear models* e.g., neural networks (see [7], [10], [11] for more detailed discussions), *example-based methods* (e.g., nearest-neighbor and case-based reasoning methods) [12], *probabilistic graphical dependency models* e.g., Bayesian networks [13]—[15], and *relational attribute models* [16]. Model representation determines both the flexibility of the model in representing the data and the interpretability of the model in human terms. Typically, the more complex models may fit the data better but may also be more difficult to understand and to fit reliably. While researchers tend to advocate complex models, practitioners involved in successful applications often use simpler models due to their robustness and interpretability [2—5].

In this paper, we present the DM process outlined in this Section, applied to hydrological data acquired at the School of Engineering & Physics, University of the South Pacific, Fiji to demonstrate the usefulness of this emerging technology in practical real-life applications. The hydrological data is soil moisture observed using automated instruments.

2 Self-Organizing Data Mining

Today, knowledge extraction from data is the key to success in many fields. Knowledge extraction techniques and tools can assist humans in analyzing the mountains of data and to turn information contained in the data into successful decision making. Experience gained from expert systems, statistics, Neural Networks or other modeling methods has shown that there is a need to try to limit the involvement of modelers (users) in the overall knowledge extraction process to the inclusion of existing a priori knowledge, exclusively, while making the process more automated and more objective. Additionally, most users' interest is in results in their field and they may not have time for learning advanced mathematical, cybernetic and statistical techniques and/or for using dialog driven modeling tools. Self-organizing modeling (SOM) is based on these demands and is a powerful way to generate models from ill-defined problems. A powerful method for model self-organization is the Group Method of Data Handling (GMDH) invented by Ivakhnenko [17],[18]. GMDH combines the best of both statistics and Neural Networks

features while considering a very important additional principle of *induction*. This cybernetic principle enables GMDH to perform not only in advanced model parameter estimation but, more important, to perform an automatic model structure synthesis and model validation, too. GMDH creates adaptively models from data in form of networks of optimized transfer functions (active neurons) in a repetitive generation of populations (layers or generations) of alternative models of growing complexity and corresponding model validation and fitness selection until an optimal complex model which is not too simple and not too complex (over-fitted) has been created. Neither, the number of neurons and the number of layers in the network, nor the actual behavior of each created neuron (transfer function of active neuron) are predefined. All these are adjusted during the process of self-organization by the process itself. As a result, an explicit analytical model representing relevant relationships between input and output variables is available immediately after modeling. This model contains the extracted knowledge applicable for interpretation, prediction, classification or diagnosis problems. For detailed discussion of GMDH for self-organizing data mining applications, see [19]. Other self-organizing network variants derived from GMDH include polynomial neural networks [20]. In a wider sense, the spectrum of self-organizing modeling contains regression-based methods, rule-based methods, symbolic modeling and nonparametric model selection methods. Table 1 shows some data mining functions and more appropriate SOM algorithms for addressing these functions (FRI: Fuzzy rule induction using GMDH, AC: Analog Complexing).

Table 1. Algorithms for self-organizing modeling (see [19] for such classification)

Data Mining functions	**Algorithm**
classification	GMDH, FRI, AC
clustering	AC
modeling	GMDH, FRI
time series forecasting	AC, GMDH, FRI
sequential patterns	AC

3 Inductive Modeling Applications to Data Mining

Inductive modeling aims at constructing an efficient and effective model of high dimensional data. In a given set of inputs, system state, and outputs, the third component is always deducible with the other two at hand (see Figure 1). A training dataset of inputs, **X**, and system states, **S**, can be used to estimate the ensuing outputs, **Y**, in a *prediction* or forecast model. It is a *modeling* or design problem to obtain a system, **S**, for given inputs and outputs **X** and **Y**. A *control* problem is to seek the optimal inputs, **X** for a given system states, **S**, can be used to estimate the ensuing outputs, **Y**. These concepts are well described [21]. Consequently three scenarios exist:

(i) Prediction—the output associated with the unseen input data is estimative, given the input training data and the system model;

(ii) Modeling—a modeling problem is intended to model a system while knowing the set of inputs and outputs;

(iii) Control—in a control problem, the goal is to find the best inputs for a given system with known outputs.

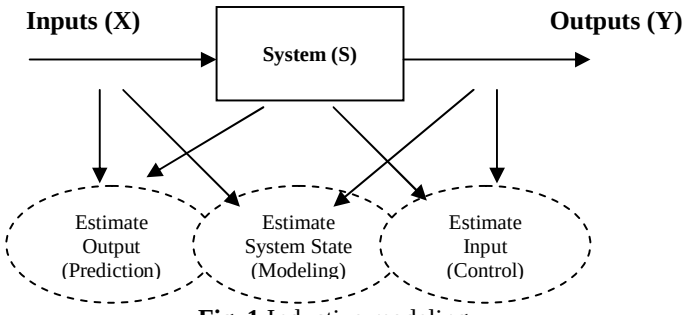

Fig. 1 Inductive modeling

4 The Group Method of Data Handling (GMDH)

The basics steps involved in the original Group Method of Data Handling (GMDH) modeling approach [17] are as follows:

The basics steps involved in the original Group Method of Data Handling (GMDH) modeling are:

Preamble: collect regression-type data of *n*-observations and divide the data into training and testing sets: $x_{ij};\ y_i \qquad i=1,2,...,n;\ j=1,2,...,m$

Step 1: Construct ${}^{m}C_2$ new variables $Z_1, Z_2, Z_3, ..., Z_{\binom{m}{2}}$, in the *training dataset* for all independent variables (columns of X), two at a time $\left(x_{i,k-1}, x_{i,k};\ i\in[1,m], k\in\left[2,\binom{m}{2}\right]\right)$ and construct the regression polynomial:

$$Z_1 = A + Bx_1 + Cx_2 + Dx_1^2 + Ex_2^2 + Fx_1x_2 \quad \text{at points } (x_{11}, x_{12}) \tag{1}$$

$$Z_k = A + Bx_{k-1} + Cx_k + Dx_{k-1}^2 + Ex_k^2 + Fx_{k-1}x_k \quad \text{at points } (x_{i,k-1}, x_{i,k}) \tag{2}$$

Step 2: For each of these regression surfaces, evaluate the polynomial at all *n* data points (i.e. using *A*, *B*, *C*, *D*, *E*, and *F* obtained from $x_{i,k-1}, x_{i,k};\ y_i$ for training). The coefficients of the polynomial are found by least square fitting as given in Press *et al* [22], or singular value decomposition for singular-value problems as given in [23] using the data in the training set.

Step 3: Eliminate the least effective variables: replace the columns of X (old variables) by those columns of Z (new variables) that best estimate the dependent variable *y* in the testing dataset such that

$$d_k^2 = \sum_{i=n_1+1}^{n}(y_i - z_{i,k})^2, \qquad k\in\left[1,2,...,\binom{m}{2}\right] \tag{3}$$

Order Z according to the least square error $d_k \mid \|d_j\| < R$ where *R* is some prescribed number chosen a priori. Replace columns of X with the best Z's $(Z_{<R})$; in other words $X_{<R} \leftarrow Z_{<R}$

Step 4: Test for convergence. Let $DMIN = d_l$ where *l* = number of iterations. If $DMIN_l = DMIN_{l-1}$ go to Step 1, else stop the process.

Since the introduction of GMDH, there have been variants devised from different perspectives to realize more competitive networks.

5 The Differential Evolution Algorithm

The outline of enhanced DE [24-26] used for the hybrid DE-GMDH is given as:

Initial Phase

1. *Population Generation:* An initial number of discrete trial solutions are generated for the initial population.

Conversion

2. *Discrete to Floating Conversion:* This conversion scheme transforms the parent solution into the required continuous solution.

3. *DE Strategy:* The DE strategy transforms the parent solution into the child solution using its inbuilt crossover and mutation schemas.

4. *Conversion from Floating to Discrete form:* This conversion schema transforms the continuous child solution into a discrete solution.

Mutation

5. *Relative Mutation Scheme:* Formulates the child solution into the discrete solution of unique values.

Improvement Strategy

6. *Mutation:* Standard mutation is applied to obtain a better solution.

7. *Insertion:* Uses a two-point cascade to obtain a better solution.

8. *Repeat:* Execute steps 2-7 until reaching a specified cutoff limit on the total number of iterations.

Local Search

9. *Local Search:* Is initiated if stagnation occurs

6 The Hybrid DE-GMDH Technique

The outline of the *DE-GMDH* technique which is fully described in [27] consists of the following steps.
Step 1: *Determine system's input variables*: Define the input variables of the system as $\boldsymbol{x}_i \ (i = 1, 2,..., n)$ related to output variable *y*.
Step 2: *Form training and testing data*: The input-output data set $(\boldsymbol{x}_i, y_i) = (x_{1i}, x_{2i},..., x_{ni}, y_i), i = 1, 2,..., n$ (n: the total number of data) is divided into a training and a testing dataset. Their sizes are denoted by n_{tr} and n_{te} respectively, and $n = n_{tr} + n_{te}$. The training data set is used to construct the *DE-GMDH* model. Next, the testing data set is used to evaluate the quality of the model.
Step 3: *Determine initial information for constructing the DE-GMDH structure*: We determine initial information for the *DE-GMDH* structure: i) The termination method. ii) The maximum number of input variables used at each node in the corresponding layer. iii) The value of the weighting factor of the aggregate objective function.
Step 4: *Determine polynomial neuron (PN) structure using DE design*: Determining the polynomial neuron (PN), is concerned with the selection of the number of input variables, the polynomial order, and the input variables to be assigned in each node of the corresponding layer. The PN structure is determined using DE design. The DE design available in a PN structure uses a solution vector of DE is the one illustrated in Fig. 2 in which the design of optimal parameters available within the PN (viz. the number of input variables, the order of the polynomials, and input variables) at last leads to a structurally and parametrically optimized network, which is more flexible as well as simpler in architecture than the conventional *GMDH*. Each sub-step of the DE design procedure of three kinds of parameters available within the PN has already been discussed. The polynomials differ according to the number of input variables and the polynomial order. Several types of polynomials can be used such as Bilinear, Biquadratic, Modified biquadratic, Trilinear, Triquadratic, and Modified triquadratic.
Step 5: *Coefficient estimation of the polynomial corresponding to the selected node (PN)*: The vector of the coefficients of the PDs is determined using a standard mean squared error by minimizing the following index:

$$E_r = \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} (y_i - z_{ki})^2, \qquad k = 1, 2,..., r, \tag{4}$$

where z_{ki} denotes the output of the *k*-th node with respect to the *i*-th data, r is the value in the second system parameter $P_2 \in [1, r]$ and n_{tr} is the number of training data subsets. Evidently, the coefficients of the PN of nodes in each layer are determined by the standard least square method. This procedure is implemented repeatedly for all nodes of the layer and also for all *DE-GMDH* layers starting from the input layer and moving to the output layer.
Step 6: *Select nodes (PNs) with the best predictive capability, and construct their corresponding layer*: As shown in Fig. 2, all nodes of the corresponding layer of *DE-GMDH* architecture are constructed by DE optimization. The generation process of PNs in the corresponding layer is described in detail as the design procedure of 4 sub-steps. A sequence of the sub-steps is as follows:
***Sub-step 1*)** We determine initial DE information for generation of the *DE-GMDH* architecture. That is, the number of generations and populations, mutation rate, crossover rate, and the length of a solution vector.
Sub-step 2*)** The nodes (PNs) are generated by DE design as many as the number of populations in the 1st generation. Where, one population takes the same role as one node (PN) in the *DE-GMDH* architecture and each population is operated by DE as shown in Fig. 2. That is, the number of input variables, the order of the polynomials, and the input variables as one individual (population) are selected by DE. The polynomial parameters are produced by the standard least squares method. ***Sub-step 3*)** Evaluate the performance of PNs (nodes) in each population as already discussed. ***Sub-step 4*)** To produce the next generation, we carry out *mutation*, *crossover*, and *selection* operations using DE initial information and the fitness values obtained from ***sub-step 3. Generally, after these DE operations, the overall fitness of the population improves. We choose several PNs characterized by the best fitness values. Here, we select the node that has the highest fitness value for optimal operation of the next iteration in the *DE-GMDH* algorithm. The outputs of the retained nodes (PNs) serve as inputs in the subsequent layer of the network. The iterative process generates the optimal nodes of a layer in the *DE-GMDH* model.

Step 7: *Termination criterion:* The termination method exploited here the maximum number of generations predetermined by the designer to achieve a balance between model accuracy and its complexity.

6 Data Mining Experimentation

6.1 Data gathering

The hydrological data used for the data mining application described in this paper was acquired at the School of Engineering & Physics, University of the South Pacific, Fiji. The hydrological data is the soil moisture observed from 2004—2007 using automated instruments. Data logger is used to capture the data from the local station to a dedicated PC located in Physics laboratory. The transmitted weather data is then copied to Excel spreadsheets and archived on daily basis as well as monthly basis to ease data identification. The day-to-day management of instruments is undertaken by a Senior Technician.

6.2 Data cleansing

In order to utilize the acquired hydrological data, the author of this paper organized the data which where copied to CDs into a daily logical data and cross-checked all the data and eliminating all possible errors such as where the instrument failed (signified by some suspicious number) for a time interval and where blanks were found on the Excel data sheet. There is the need to eliminate such errors and/or bogus data for data integrity.

6.3 Feature extraction

The data logger used acquires levels of daily soil moisture. It was therefore necessary to extract only the interesting attributes of the data for our purpose.

6.4 Pattern extraction and discovery

The self-organizing DE-GMDH technique was used for the purpose of extraction and discovery of knowledge of the data acquired; this is the core of data mining. There were 881 rows of data for soil moisture. A time lag = 3 was used for experimentation. The forecasting evaluation criteria used for all the experiments is the mean squared error, MSE.

Case Study: University of South Pacific daily soil moisture

For the 881 rows of daily soil moisture (expressed as %) over the period of 2004—2007 the average, minimum, maximum, and standard deviation are 43.92, 20.14, 62.99 and 9.52, respectively. Using the time-lag approach, three columns of input data were generated with one column as output; the number of rows therefore reduced to 878. The external criterion that was used for the DE-GMDH approach for this particular experimentation is the mean square error, MSE. Figure 2 shows the DE-GMDH prediction and absolute difference error for the daily soil moisture data mining problem. Figure 3 shows the DE-GMDH prediction and percentage difference error for the daily soil moisture data mining problem. The absolute difference error, is found be within the range of ± 10. Here, there is a fairly good match between the measured and predicted values, showing that the proposed DE-GMDH model can be used as a feasible solution for soil moisture and other hydrological data forecasting.

In order to ascertain the efficacy of the proposed DE-GMDH approach for data mining, its results were compared to those of polynomial neural network, PNN [28] and its variant, genetically optimized PNN, gPNN [29] as shown in Table 2. The results show that the proposed DE-GMDH algorithm outperforms other approaches for both training (PI) and generalization (EPI) errors.

Table 2 Performance index of identified mode

Model	*PI*	*EPI*
PNN [28]	15.090992	12.062399
gPNN [29]	14.508822	12.062399
DE-GMDH [this paper]	0.0105188	0.009727

The DE-GMDH used for the work reported in this paper found an output sequence {1, 2, 3} and coefficients to be as follows {9.83E-06, 0.000158126, 0.000836589, 0.0271493, -0.00678421, 1.15E-07} leading to a predictive model for the soil moisture (*SM*) given as

$$SM = 0.000009 + 0.000158x(t-1) + 0.000836x(t-2) + 0.027149x(t-1)x(t-2) - 0.006784x(t-1)^2 + 0.0000001x(t-2)^2 \quad (5)$$

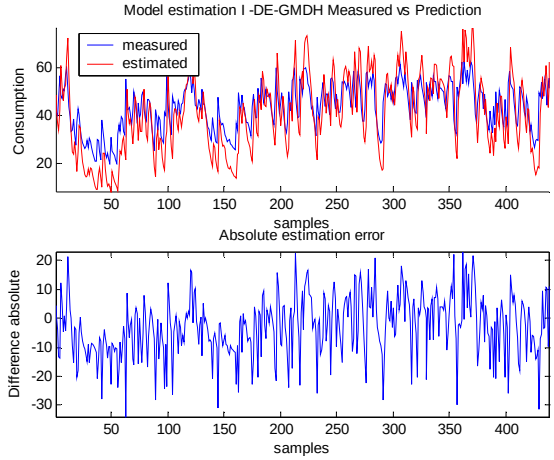

Fig. 2 The DE-GMDH prediction and absolute difference error for the soil moisture problem

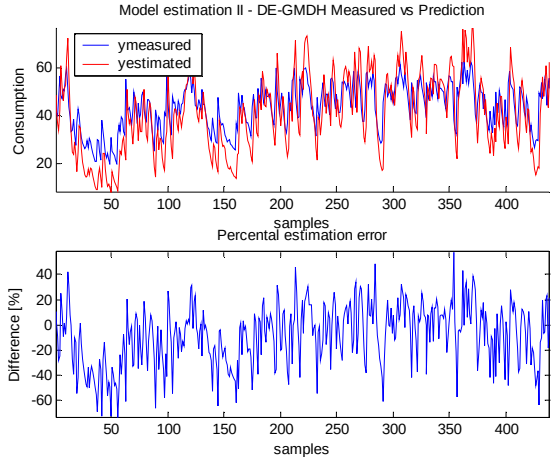

Fig. 3 The DE-GMDH prediction and percentage difference error for the soil moisture problem

6 Conclusions

This paper presents the data mining activity that was employed in mining soil moisture data for the period of 2004—2007. The self-organizing data mining approach employed is the Hybrid Differential Evolution and Group Method of Data Handling (DE-GMDH). Experimental results indicate that the proposed approach is useful for data mining technique for forecasting weather data. The results of DE-GMDH were compared to those of PNN and its variant, gPNN. The results show that the proposed DE-GMDH algorithm outperforms other approaches for both training (PI) and generalization (EPI) errors. This paper has shown that end-users of data mining should endeavor to follow the methodology for DM since suspicious data points or outliers in a vast amount of data could give unrealistic results which may affect knowledge inference. One major conclusion resulting from the studies carried out in implementing hybrid

DE-GMDH network is that population-based optimization techniques (genetic programming [GP], genetic algorithm [GA], differential evolution [DE], scatter search [SS], ant colony system [ACS], particle swarm optimization [PSO], etc.) are all candidates of hybridization with GMDH. In the past, only the use of GA and GP has been studied for hybridization with GMDH. Further research activities include incorporating more design features to improve the modeling solutions and to realize more flexibility.

References

[1] Codd, E. F. 1993. Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. E. F. Codd and Associates.

[2] Fayyad, U. M.; Djorgovski, S. G.; and Weir, N. 1996. From Digitized Images to On-Line Catalogs: Data Mining a Sky Survey. *AI Magazine* 17(2): 51–66.

[3] Fayyad, U. M.; Haussler, D.; and Stolorz, Z. 1996. KDD for Science Data Analysis: Issues and Examples. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), 50–56. Menlo Park, Calif.: American Association for Artificial Intelligence.

[4] Fayyad, U. M.; Piatetsky-Shapiro, G.; and Smyth, P. 1996. From Data Mining to Knowledge Discovery: An Overview. In *Advances in Knowledge Discovery and Data Mining*, eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 1–30. Menlo Park, Calif.: AAAI Press.

[5] Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P.; and Uthurusamy, R. 1996. *Advances in Knowledge Discovery and Data Mining.* Menlo Park, Calif.: AAAI Press.

[6] Brachman, R., and Anand, T. 1996. The Process of Knowledge Discovery in Databases: Human-Centered Approach. In *Advances in Knowledge Discovery and Data Mining*, 37–58, eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Menlo Park, Calif.: AAAI Press.

[7] Elder, J., and Pregibon, D. 1996. A Statistical Perspective on KDD. In *Advances in Knowledge Discovery and Data Mining*, eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 83–116. Menlo Park, Calif.: AAAI Press.

[8] Quinlan, J. 1992. *C4.5: Programs for Machine Learning.* San Francisco, Calif.: Morgan Kaufmann.

[9] Breiman, L.; Friedman, J. H.; Olshen, R. A.; and Stone, C. J. 1984. *Classification and Regression Trees.*Belmont, Calif.: Wadsworth.

[11] Friedman, J. H. 1989. Multivariate Adaptive Regression Splines. *Annals of Statistics*

[12] Dasarathy, B. V. 1991. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. Washington, D.C.: IEEE Computer Society.

[13] Heckerman, D. 1996. Bayesian Networks for Knowledge Discovery. In *Advances in Knowledge Discovery and Data Mining*, eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 273–306. Menlo Park, Calif.: AAAI Press.

[14] Whittaker, J. 1990. *Graphical Models in Applied Multivariate Statistics.* New York: Wiley.

[15] Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems.* San Francisco, Calif.: Morgan Kaufmann.

[16] Dzeroski, S. 1996. Inductive Logic Programming for Knowledge Discovery in Databases. In *Advances in Knowledge Discovery and Data Mining*, eds. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 59–82. Menlo Park, Calif.: AAAI Press. 19:1–141.

[17] Ivakhnenko, A. G.; 1971. Polynomial Theory of Complex Systems, IEEE Transactions on Systems, Man, and Cybernetics, pp. 364-378.

[18] Madala, H. R. Ivakhnenko, A. G.; 1994. Inductive Learning Algorithms for Complex Systems Modeling, CRC Press, Boca Raton.

[19] Mueller, J-A, and Lemke, F.; 1999. *Self-organizing Data Mining: An Integrated Approach to Extract Knowledge From Data*, Dresden, Berlin.

[20] Oh, S. K., Pedrycz, W.; 2002. The design of self-organizing polynomial neural networks, Inf. Sci. 141 237-258.

[21] Elder IV, J. F., and Brown, D. E.: Inductive and Polynomial Networks, Technical Report: IPC-TR-92-009, Institute for Parallel Computation and Department of Systems Engineering, University of Virginia, Charlottesville, VA, USA, 1992

[22] Press, W H, Teukolsky, S A, Vetterling, W. T., Flannery, B. P.; 1992. "*Numerical recipes in C: The art of scientific computing*". Cambridge University Press.

[23] Nariman-Zadeh, N., Darvizeh, A., and Ahmad-Zadeh, G. R., Hybrid genetic design of GMDH-type neural networks using singular value decomposition for modeling and predicting of the explosive cutting process, Nariman-Zadeh, Proc. Instn Mech. Engrs Vol 217 Part B: Nariman-Zadeh, 779-790.
[24] Davendra, D., *Hybrid Differential Evolution and Scatter Search*, MSc Thesis, The University of the South Pacific, 2003.

[25] Onwubolu, G. C., and Davendra, D., Scheduling flow shops using differential evolution algorithm, *European Journal of Operational Research*, 171, 674-692, 2006.
[26] Davendra, D., and Onwubolu, G. C., Scheduling flow shops using enhanced differential evolution algorithm, *European Conference on Modeling and Simulation (ECMS)*, Prague, Czech, 2007.
[27] Onwubolu, G. C., Design of Hybrid Differential Evolution and Group Method of Data Handling for Inductive Modeling, *International Workshop on Inductive Modeling*, Prague, Czech, September 23-26, 2007
[28] S.-K. Oh, W. Pedrycz, and B.-J. Park, "Polynomial neural networks architecture: analysis and design," *Computers and Electrical Engineering*, vol. 29, no. 6, pp. 703-725, 2003.
[29] S.-K. Oh, Park, B.-J., and Kim, H.-K, Genetically optimized hybrid fuzzy neural networks based on linear fuzzy inference rules, International Journal of Control, Automation, and Systems, 3(2), pp. 183-194, 2005.