GMDH-Based Knowledge Extraction from Data

J.-A. Müller

University of Applied Sciences, Dresden (HTW), Germany

By means of application of GMDH and self-organisation, the possibility to automate the whole data analysis process named knowledge extraction from data is described. Different GMDH based modelling algorithms are implemented: dimensionality reduction, missing value elimination, active neurons, enhanced network synthesis and creation of systems of equations, combining of alternative models, – but also validation to make knowledge extraction systematically, fast and easy-to-use even for large and complex systems.

Key words: GMDH, self-organisation, data mining, knowledge extraction from data

1. Introduction

Knowledge extraction from data, based on self-organising modelling technology is sufficiently different from ensemble of algorithms and tools, today united under the slogan 'data mining'.

The first distinction of this new kind of algorithms to data mining is the objective process of modelling without almost subjective intervention of the user. In our time data mining algorithms and tools, which are based on traditional common used algorithms of mathematical statistics and on algorithms of artificial intelligence demand an comfortable knowledge about the underlying basic methodologies (statistical theory, theory of Neural networks, Fuzzy theory, genetic algorithms a.o.), a priori information about the structure of the mathematical model and experience in their successful application but also deep understanding in the application field. One most important condition for a successful data mining is therefore the combination of experience in application of these tools and know-how in the application field. For a more sophisticated approach it is most important to limit the user involvement in the entire knowledge extraction process to the inclusion of well-known a priori knowledge. This makes the process more automated and more objective. Most users' primary interest is in generating useful and valid model results without having to have extensive knowledge of mathematical, cybernetic, statistical techniques and principles of artificial intelligence and without deep understanding and theoretical systems analysis of processes in application field. More than this, in most practical cases there is no sufficient time for complex dialog driven modelling tools but a lot of analysis and predictions every time has to be realised.

Secondly there is a big difference between interpolation tasks solved by traditional statistical and known artificial intelligence tools and knowledge extraction from data. Data mining works using interpolation algorithms of artificial intelligence without application of self-organisation of models. Most score functions, which rank models as a function of how useful the models are to the data miner, are based on error. Results of data mining are valid only in the sample space of all given data. Data mining is able to solve any interpolation tasks, it means in the result of data mining is

generated a description of the given data in form of models and patterns derived from it. Models and patterns contain information of a given data set.

New information or knowledge for new data not contained in the given data set can not be derived without continuing of the learning process with new data or on the base of inductive methods. An inductive knowledge extraction employs the same algorithms, however with application of model self-organisation. Therefore knowledge extraction results in models of large generalisation power and accuracy. Knowledge extracted by inductive methods is valid not only on given data base but also on new data. This is possible only using score functions based not only on error but also on bias, using a division of data in two or more parts or cross-validation. Selection feature depends from dispersion of noises. Knowledge extraction from data, which is able to derive knowledge contained in new data on the base of inductive principles, includes as an essential part the GMDH approach.

2. Group Method of Data Handling (GMDH)

The basic idea of GMDH approach, developed by A.G.Ivakhnenko, originated from several interpolation tasks which had to solve in technical automatic control. From systems theory was known, that every system can be described by Volterra functional series. Discrete analogue of the Volterra functional series are higher-order polynomials of the Kolmogorov-Gabor form, which consists a lot of unknown parameters. Up to beginning of GMDH there were only two ways to estimate the unknown parameters, by means of

- least squares methods, which demands the solution of the Gaussian normal equations,
- adaptive methods, such as stochastic approximation.

Application of classical approach of systems of normal equations is possible only, if the number of observations is much more (normally five to ten times) than number of unknown parameters. More than this the solution of system of equations was efficient only up to a medium number of unknown parameters.

Ivakhnenko's approach is the following:

1. to use in pairs the variables;
2. in a recursive way to solve the big system of equations by many little systems of equations (3 to 7 equations). For every pair of variables gives the possibility to solve the interpolation task independent from the number of unknown parameters in an effective way. Unfortunately the total number of combination gets very large very quickly. This is known as the combinatorial explosion and can very quickly defeat any computer.
3. Because there is no practical way to check for every combination, some other method of estimating needs to be used. The third part of GMDH is the multi-layered approach. According to this principal instead of generation of all possible combinations in one layer using the

principal of selection of best models among all generated models with two variables are selected the best who in the next layer are again combined.

The basic idea is that once the elements on a lower level are estimated and the corresponding intermediate outputs are computed, the parameters of the elements of the next level can be estimated then. In the first layer, all possible pairs of the inputs are considered and all or only some best models of the layer (intermediate models) - in the sense of the selection criterion - are used as inputs for the next layer(s). In the succeeding layers all possible pairs of the intermediate models from the preceding layer(s) are connected as inputs to the units of the next layer(s). This means that the output of a unit of a processed level is or may become dependent from a local threshold value an input to several other units in the next level. Finally, when additional layers provide no further improvement, the network synthesis stops.

The principal of generation of models with growing complexity gives an architecture near the neural networks of perceptron type, but

- instead of stochastic mutations is used a systematic procedure of increasing the complexity of generated model variants;
- instead of adaptation of unknown parameters is used the estimation of parameters by means of little systems of normal equations for every neuron (transfer function);
- in neural network applications the structure of the network is preselected (number of layers, number of neurons in a layer a.o.). The objective of GMDH approach is to estimate networks of the right size with a structure evolved during the estimation process.

This inductive approach is composed of:

- the cybernetic principle of self-organisation as an adaptive creation of a network without subjective points given;
- the principle of external complement enabling an objective selection of a model of optimal complexity and
- the principle of regularization of ill-posed tasks.

and used in self-organising modelling (SOM), further developed by Lemke and Müller [1], focusing on application of cross-validation principles, optimisation of the structure of the transfer functions (active neurons) and generation of systems of equations a/o. and realised in software “KnowledgeMiner” (www.knowledgeminer.net). The task is the following:

given: large number of candidates (alternative models, generated with growing complexity), where the underlying model is not known;

in demand: to choose the candidate model of the right complexity to describe the training data.

3. Knowledge extraction from data

In contrast to Neural Networks using Genetic Algorithms as an external procedure to optimise the network architecture and several pruning techniques to counteract over-training, the SOM approach introduces principles of evolution - inheritance, mutation and selection - for generating a network structure systematically enabling automatic model structure synthesis and model validation. Models are generated adaptively from data in form of networks of active neurons in an evolutionary fashion of repetitive generation of populations of competing models of growing complexity, their validation and selection until an optimal complex model - not too simple and not too complex - have been created. If this, but also data reduction, pre-processing and validation of model results is adjusted during the process of self-organization, this whole process is called knowledge extraction from data based on self-organisation. Among other steps such an approach realises in an objective way

- data transformation;
- pre-processing, such as elimination of missing values;
- data (dimensionality) reduction in state and/or sample space;
- choice of appropriate model and accordingly data mining algorithm;
- self-organisation of transfer functions (neurons);
- generation of alternative models with different variables and growing complexity in each layer;
- self-organisation of networks;
- for multi-output systems: self-organisation of systems of networks (autonomous systems of equations);
- validation of generated models;
- self-organisation of hybrid models;
- using some control module on the base of generated models and/or predictions automatically to derive decisions.

This approach is implemented in software “KnowledgeMiner” and supported by AppleScript, In such a way is given not only a theoretical foundation but also an effective software base to realise knowledge extraction from data.

4. Self organisation of data preparation

Data preparation builds a ready to model dataset. But preparing data for modelling has been an extremely time-consuming process (nearly 60 % [2]). A first objective in preparing the data set is to prepare the data in such a way that the information content is best revealed for the tool to see. A second objective is to obviate the problems where possible. This techniques can reduce the error rate in a module, reduce model building time and give enormous insight into the data and therefore is a source of most important benefits. If there is some automatic action that can correct the problem, so much the better [2].

4.1 Missing values

In most practical applications we have to do with some of the values in the data set not observed. A general problem in replacement of missing values is that there may be some information content, carried by the actual pattern of measurements missing. Creating and inserting some replacement value for the missing value the objective it is to guarantee that this values at least are neutral that is neither adds nor subtracts information from the data set. Poorly chosen values adds information to the data set, that is not really present the missing value and in such a way distorting the data. It is necessary to preserve the between-variables relationship, which will be explored in the next modelling step.

With the variables values that are jointly present in the initial sample data set good estimates of missing values of any variable can be made using SOM algorithms. This approach has two steps:

1. deletion of missing values (list deletion): All observations with one or more missing values are deleted. On the base of such reduced data set by means of SOM algorithm are generated for every variable with missing values a linear or nonlinear model, where as variables are used all variables of the whole data set without the considered variable including the output variables. Specially for small data sets is convinient the advantages of SOM algorithm generate models on small data sets.
2. Using the generated models for every variable by means of what-if-prediction the missing values can be estimated. After this the whole data set can be used to solve the data mining task.

If there are more than one missing values in one and the same record may be it is difficult to estimate the unknown values by regression models because of conflicts between variables with missing values (interdependence between variables to predict). In this case Analog Complexing (AC) classification is useful. For the given record with missing values by AC classification are selected most similar records which does not contain missing values. After this, the missing values are estimated as mean value of measured values of similar records.

4.2 Data (dimensionality) reduction

A crucial problem is determining how much data is needed for modelling. Reducing the dimensionality it is necessary to enhance the relationships really in the data. Therefore the mostly not proofable demand is the data sets needs to be representative. Secondly a concentration of instances has to enhance the whole information about between variable relationships but also the variability of individual variables. Practical data mining application has to handle with mountains of data, i.e. tables with high dimension. Besides the known theoretical dimensionality problem [3] there is a dimension limit of all known tools connected with computer time and working storage. Therefore a step of high priority is the objective choice of essential variables - state space reduction -but also the exclusion of redundant observations - sample space reduction.

4.2.1 State space reduction

a. Modular approach

The modular approach relies on decomposing a given task among a number of networks, each solving a seperate sub-task. Given are N variables x_i, from which M variables y_j has to be predicted. The following approach can be used for dynamic systems:

1. Randomly the variables are grouped in P groups with nearly the same number of variables. Every group gives a sub-data set.
2. For every sub-data set is generated a system of equations by means of SOM and estimated the system prediction for all variables. If the variables y_j is included in the group, a prediction is obtained, in the other case a model of variables y_j depending of all variables included in the group is generated and by means of what- if - prediction the unknown prediction of y_j evaluated.
3. The obtained P predictions for every variable y_j are combined by means of SOM algorithms (look at section 7).

b. Self-organising variable selection

The basic idea is to use the GMDH principle, it is the aim to reduce high dimensional problems by solving many small problems. The variable set S_0 with a high number of variables is divided into m subsets with equal number of variables. In the first generation for every combination in pairs of subsets linear or non-linear models are generated. Every generated model contains variables, the set of all variables, contained in $\binom{m}{2}$ models of the first generation give the variable set S_1 of first generation. In the second generation the new variables set S_1 will be divided in equal subsets and for every combination in pairs of subsets again linear or non-linear models are generated. The whole set of variables, contained in models of second generation gives the variables set S_2 and so on. This has to continue up to a given number of variables is reached.

Obviously for very high number of variables such an approach does need many computational effort, the number of models which has to be generated is too much. In this case only a partition in equal groups of variables is useful, but there is the danger, that not all influences in pairs on the output are considered.

4.2.2 Sample space reduction

We have to differ between stationary processes or homogeneous samples of observations and non-stationary processes or inhomogeneous samples of observations. In both cases we can use self-organising clustering to generate homogeneous or stationary parts of the whole process /sample. But combining of results is possible only in first case. The second case gives models for different situation. Using this model ensemble it needs therefore a classification of the given situation and after this the application of the corresponding model.

a. Stationary processes

If there is a big number of observations, i.e. a high sample size one useful approach the sample has to be divided by clustering in several clusters of sample observations. After clustering every cluster can represented by one observation. This representative can be selected as one specific observation, mean value of all obervations in the cluster, observation, which has the lowest distance to all other a.s.o. Principally it has to be considered the fact, that redundancy of samples contains some information. To delete the redundant instances means to lost this information. Therefore it is better to use weights, which give instances with repeated realisations a higher weight than such, which only one times will be.

b. Nonstationary processes:

By means of AC clustering the observations are divided into a small number of clusters with similar records. Every cluster contains observations of a nearly stationary process. After this for every cluster can generated by means of SOM algorithms a special model.

5. Self-organising modelling (SOM)

Self-organising modelling consists of several steps of self-organisation, such as

- self-organisation of transfer functions (neurons);
- self-organisation of networks;
- for multi-output systems: self-organisation of systems of networks (autonomous systems of equations);

In such a way it is possible to generate from short and noisy data samples

- linear/nonlinear time series models,
- static/dynamic linear/nonlinear multi-input/single-output models,
- systems of linear/nonlinear difference equations (multi-input/multi-output models),
- systems of static/dynamic mutli-input/multi-output fuzzy rules

described analytically in all four cases. More than this for high noise level nonparametric models (pattern/cluster) are obtained by Analog Complexing. Using Analog Complexing not only prediction but also cluster analysis (AC clustering) and classification (AC classification) [1] is possible.

5.1 Self-organisation of transfer functions (neurons)

A GMDH algorithm realises for each created neuron an optimisation of the structure of its transfer function (Active Neuron). Each transfer function f_k is adaptively created by another self-organising process and they may differ one from another by their number of variables used and by their functional structure and complexity. SOM has implemented a complete second order polynomial as default analytical elementary model structure:

$$f(v_i, v_j) = a_0 + a_1 v_i + a_2 v_j + a_3 v_i v_j + a_4 v_i^2 + a_5 v_j^2.$$

This abstract elementary model defines the class of possible models for this level of self-organisation. The arguments v_i, v_j represent all kinds of input data like non lagged input variables $x_{i,t}$, lagged input variables $x_{i,t-n}$, derivative input variables or even functions or models, e.g., $\sqrt{x_i}$, $1/x_i$, $\sin(x_i)$ or $\log(x_i)$. The true model of every created neuron is instantiated adaptively by self-organisation. As a result, the synthesised network is a composition of different, a priori unknown neurons, and their corresponding transfer function have been selected from all possible linear or non-linear polynomials: $f(v_i, v_j)$

5.2 Self-organisation of networks

The second level of self-organisation employs a multi-layered iterative GMDH algorithm. There are two enhancements to the basic algorithm, however.

The first difference is that the neurons must not have two input variables due to their self-selecting capability. The second difference of SOM algorithm is applying a so-called layer-break-through structure: all original input variables v_i and all selected F_p best models of all p preceding layers are used as inputs to create the model candidates of layer p+1. The enhanced version breaks open this fixed layer dependence structure, and it allows considering any selected model candidate (or original input variable) as input information at any layer of model creation.

This greater flexibility of model synthesis, however, also amplifies the danger that models are becoming increasingly collinear with growing number of layers. To avoid evolution of collinear input variables generated during modelling, a statistical test is processed before any new neuron will be created excluding collinear input pairs from modelling in this way. Such an algorithm for self-organisation of multi-layered networks of active neurons performs the creation of an optimal complex network structure (optimal number of neurons and number of layers) and selection of a number of best model candidates out of populations of competitive models.

5.3 Self-organisation of systems of networks (autonomous systems of equations)

Complex systems usually have several output variables. The goal of modelling systems of equations using GMDH is to self-organise a model for each output variable and to identify the interdependence structure between the system variables including separating variables into endogenous and exogenous variables according to their corresponding model quality. After modelling a system of m equations, SOM selects a best autonomous system consisting of m* equations ($m^* < m$) necessary to describe the system completely. Here, the number m* of equations the best system consists of and its composition of variables is completely detected by the algorithm using a system criterion. All variables of the identified best system can be considered as endogenous variables of the system. All remaining variables which may be part of the autonomous system are either exogenous or are identified as exogenous due to an insufficient data basis.

6. Validation

A very important and still unsolved problem in knowledge extraction from data is analysis and validation of the obtained models. This evaluation process is an important condition for application of models obtained by data mining. Only from data analysis it is impossible to decide whether the estimated model can reflect the causal relationship between input and output adequately or whether it is a stochastic model with non-causal - correlations.

6.1 Data driven approach

The data-driven approach generates a description of the system behaviour from observations of real systems evaluating how it behaves (output) under different conditions (input). This is similar to statistical modelling and its goal is to infer general laws from specific cases. The mathematical relationship that assigns an input to an output and that imitates the behaviour of a real-world system using these relationships usually has nothing to do with the real processes running in the system, however. Cherkassky [3] underlined that the task of learning/estimation of statistical dependency between (observed) inputs and outputs can occur in the following situations or any combination of them:

- output causally depend on the (observed) inputs;
- inputs causally depend on the output(s);
- input-output dependency is caused by other (unobserved) factors;
- input-output correlation is non-causal.

It follows that causality cannot be inferred from data analysis alone; instead each of the 4 possibilities or there combination is specified and therefore causality must be assumed or demonstrated by arguments outside the data [3] and can not proofed by a technical validation process.

With insufficient a priori information about the system to be modelled, there are several methodological problems we have to focus on before applying data-driven methodologies. The incomplete - since finite - data basis we always use leads to an indeterminacy of the model and the computational data derived from it. This incompleteness of the theoretical knowledge and the insufficient data base causes the problems mentioned in [1].

SOM focuses tightly on the practical application of results, it is very successful to generate a black box that is known to work under specific conditions and applications. In this paper therefore validation means to proof the derived pattern from data exists actually and is important for practical applications or it is only stochastic.

6.2. Validation of input/output models

By means of Monte Carlo simulation [4] a noise sensitivity characteristic, was generated that provides the required external information which helps to decide the generated input/output model is valid or not. The idea here was building models on a subsequently increasing number of potential inputs M and random samples N several times to get a characteristics for a certain algorithm on how strong the algorithm can filter out noise based on a given data set dimension (N, M). In result, a boundary area Q_u =f(N, M) was obtained that any model must exceed to be considered valid to a certain degree of significance in that it reflects relevant relations in the data.

Based on the simulation data of 750 samples and 2 respectively 4 inputs when also using the inverse of N and M) the following model - named in the following test function - was created by "KnowledgeMiner":

$$\hat{Q}_u = \hat{a}\,\frac{1}{N} + \hat{b}\,\frac{M}{N}, 0 \le \hat{Q}_u \le 1,$$

and concluding from that model:

$$N \ge f\left(M \mid Q_u \le \varepsilon\right) = cM + d, c = \frac{\hat{b}}{\varepsilon}, d = \frac{\hat{a}}{\varepsilon}\ \text{, and} \qquad M \le f\left(N \mid Q_u \le \varepsilon\right) = eN + g, e = \frac{\varepsilon}{\hat{b}}, g = -\frac{\hat{a}}{\hat{b}}\ .$$

To evaluate if the extracted relation Q_u=f(N, M) do extrapolate well, we run the simulation on N, M values that were not included in the data set used for Q_u model estimation. This simulation shows for M=100 and N=10 (50) 810 that theoretical model $Q_u = \overline{Q} + 2s_Q$ and estimated model $\hat{Q}_u = f(N, M)$ are fitting very close, which confirms applicability of the estimated model on the extrapolated parameters.

Concluding from these simulations it seems reasonable that the obtained test function $Q_u(N,M)$ provides a tool that helps to estimate on the fly the validity of a model generated using GMDH. Given a data set of dimension (N, M), a model's quality Q^M can be calculated and compared to a corresponding threshold Q_u. This threshold expresses a "model quality" level that can be obtained when simply using random numbers as a data basis. For a model of a quality $Q^M \le Q_u$, it cannot be verified - due to missing error cases - whether the model reflects some relevant relations or if it just models noise. Therefore, such a model has to be considered invalid. For the other test case, $Q^M > Q_u$, it can be concluded that the probability of the test indicating a model valid for actually non-relevant relations in the data decreases fast asymptotically as the difference $Q^M - Q_u$ rises. This is a most important fact, because, having the error rate available this time, this implies that as ΔQ rises, the probability of testing an actually valid model valid quickly increases to almost 1.

Looking at a model quality or model error criterion does not suffice to state a model valid or not, and thus considering it a good model that generalise well. The "closeness of fit" hype is misleading: Even an ideally fitted model (R^2=1) can reflect non-causal, i.e., random relations, exclusively, as well as the "worst" fitted model (R^2=0) can be the "best" or "true" model as this test clearly shows. A model's closeness-of-fit-criterion needs justification with the "working characteristics" of the

algorithm it was created with. In this context, this noise sensitivity characteristics provides the required external information to be able stating a model not being valid or being valid the more as the model's quality Q^M distinguishes from an externally given quality level Q_u ($Q^M - Q_u >> 0$).

7. Improvement of model results.

If the model is not valid

- in the data base are not included most important input variables. Therefore the investigated variable cannot be explained by a input-output model sufficiently. The variable should by considered as exogenous and should be described by a time series model or by Analog Complexing.
- the data base is not well-behaved, i.e. there are more variables than observations. Besides methods of dimensionality reduction quality of model results can be improved by combining.

In many fields, such as economy, there are only a little number of observations which is the reason for uncertain results. The results obtained by models with small sample are in most cases un-sufficient. All methods of automated model selection lead to a single best model. On this base are made conclusion and decision as if the model were the true model. However this ignores the major component of uncertainty, namely uncertainty about the model itself. To improve model results artificial generation of more training cases by means of jittering, randomising a.o. is a powerful way.

Many researches have shown, that simply combining the output of many predictors can generate most accurate prediction that of any of the individual predictor. Theoretical and empirical work [5] has shown, that a good ensemble is one where the individual networks are both accurate and make their errors on different parts of the input space. Combining the output of networks is useful only if there is disagreement on some inputs, topology, parameter a.o. Combining several identical networks produces no gain.

The task of combining is: given an ensemble of predictors, sought is a prediction by means of voting or averaging (simple, weighted, Bayesian).

Combining the corresponding outputs of a number of trained networks is similar to creating a large network in which the trained networks are sub-networks operating in parallel and the combination weights are the connection -weights of the output layer. It is possible to generate a combination of models (synthesis) by SOM algorithms itself. The big advantage of this approach is that automatically by self-organisation is selected the best (voting) or combined some of the best models linearly or nonlinearly.

8. Conclusions

GMDH based algorithms and self-organisation can be used to automate the whole data mining process, i.e. models have been created adaptively and data preparation will be self-organised in special missing values are estimated and dimensionality is reduced. Properly applied the data

preparation process prepares both the data and the user. When data is correctly prepared and surveyed the quality of the models produced will depend mostly on the content of the data, not so much on the ability of the user. The approach is implemented in software "KnowledgeMiner" and supported by AppleScript. A successful real-life application is Carcinogenicity Prediction of Aromatic Compounds, included in this issue [6].

Literature

1. Müller J.-A., Lemke F. Self-Organising Data Mining. Libri, Hamburg 2000.
2. Pyle D. Data Preparation for Data Mining. Morgan Kaufman Publ. San Francisco 1999.
3. Cherkassky V., Mulier F. Learning from Data. J. Wiley&Sons. New York 1998.
4. Lemke F., Müller J.-A. Validation in Self-Organising Data Mining. ICIM 2002. Lviv 2002.
5. Sharkey A.J.C. Combining Artificial Neural Nets: Ensemble and Modular Multi -Net Systems. Springer: London, 1999.
6. Lemke F., Benfenati E. Carcinogenicity Prediction of Aromatic Compounds Using Self-organising Data Mining. USIM, 2, 2003.