MICRO ARRAY DATA ANALYSIS BASED ON BUSINESS OBJECTS AS PART OF A WORKFLOW RELATED GENE EXPRESSION

Dietmar P. F. Moeller
University of Hamburg, Department Computer Science
Chair Computer Engineering, AB TIS
Vogt Kölln Str. 30, D-22527 Hamburg, Germany
dietmar.moeller@informatik.uni-hamburg.de

ABSTRACT

The paper presents the application of Computational Modeling and Simulation in the Science and Engineering (CMSSE) domain of gene expression. Applied to micro array data analysis one may introduce soft computing algorithm for CMSSE as part of sensitization or preprocessing. While sensitization enable the conditioning of case study specific classifiers, scientists are able using specific linguistic IF-THEN rules to create appropriate fuzzy sets, that can be helpful for use in micro array data analysis. Henceforth, scientists will be empowered handling this classifiers in situ, phased to their measuring equipment and/or case study specific parameters, under case investigation.

INTRODUCTION

We live in a world of an imposing complexity and variety, a world where events more or less never repeat exactly. Human-world interaction based on a scientific approach seems to have a normalization in models, an abstract representation, as a powerful tool to understand real world phenomena. Therefore a big part of scientific work consists in formalization, which yields models of real world systems studied. This task clearly is scientifically oriented, in the sense to gain sufficient understanding of real world phenomena, generating respective representations, based on experiments and observations. Because the scientist attempts to create representations and laws that formalize verified hypothesis concerning real world phenomena.

The formalizations are only useful if they succeed in seizing the essential features of the real world. They permit extrapolation, that allows to generalize, often correctly, from past experience to future events from which we can learn how the real world system can be manipulated for own purposes, which is a kind of uncertainty. In our world which is more or less precise understandable or predictable, we are more conscious of uncertainty, that appears in form of imprecision, vagueness and ill defined, ill separable, and doubtful data. For this kind of uncertainty, or better soft information, we have to learn to understand the intrinsic systems dynamic.

Keeping formalizations running or doing extrapolation, deals with effectively information processing, which is a task, done by computing machines, has been directly introduced as a suitable tool of the scientific approach. But using non-precise information, which is called soft information, e.g. the blood pressure is lower than normal, represented by linguistic terms such as low, high, small, medium, large, big and so on, needs a specific form of computation, which is called soft computing. Soft computing, deals with fuzzy sets, neuronal nets, genetic algorithms, evolution strategy, probabilistic methods etc. Usually, these approaches in uncertainty, combining soft information with conventional scientific methods in a so called ad hoc manner, can be investigated using computation to show the validity of the approaches in relation to the specific case study. Therefore, during the past years, processing of uncertainty or soft information processing had been applied by different disciplines for a large variety in formal representations in the several scientific application domains. Applying soft computing techniques for those formalizations, one can impart an understanding that the formalization itself can not provide. Because soft computing is a collection of methods which can be expressed in terms of algorithms, belonging to the respective disciplines, that has been proved to be of vital importance to progress in all fields of endeavor.

In general, the common problems arising from formalization in science, and especially the possibility of applying in a wide range of scientific research the same methods while solving problems, has improved the cooperation between different disciplines and removed the rigid barriers of the past between them. However, although it seems that engineers and scientists, like physicians, will have the same goals in studying their systems. But there is still an essential difference between them, formalizing a real world research problem. For instance, the engineer is interested in system formalization reflecting normal operating conditions. His aim is to use the model in case of normal operating conditions, e.g. for optimized system control, or at least to keep it in a relative close vicinity of safe operating conditions and avoid the danger resulting from the formalized system running out of control. Anyhow there is no outstanding interest for engineers formalizing plants behavior outside its allowable operating conditions range.

In contrast, the scientist, like the physician, is not solely interested in formalization a real world problem

Proceedings 19th European Conference on Modelling and Simulation
Yuri Merkuryev, Richard Zobel, Eugène Kerckhoffs © ECMS, 2005
ISBN 1-84233-112-4 (Set) / ISBN 1-84233-113-2 (CD)

under normal conditions. He would prefer that the formal description adequately describes the systems un-nominal behavior, i.e. the systems behavior outside of normal limitations, like pathological states in case of hypertension versus normotension, or gene expression analysis in relation to the type or growth of a tumor, and there are serious limitations. But engineering techniques dealing with uncertainty are sometimes as much statements comparable to cognitive and linguistic sciences as they are about engineering, and hence they are comparable to science, like medicine.

In practice, the formalization of models itself is an iterative process, consisting of measurements at the real world system –if possible–, and computing strategies by changing the structure of the formal description in an effort to closely match the complex dynamic systems behavior. The computing strategy may be based on the category in the nearest neighbor sense, if the adapted representation is close enough to the previous one. In fact the formalization has served its purpose when an optimal match is obtained between computed results and data obtained from the real world system under test.

Soft information processing generate the basic insight that categories are not absolutely clear cut, they belong to lesser or greater degree to that category. Hence soft computing systems break with the tradition, that real world phenomena can be precisely and unambiguously characterized, which means divided into categories, and then manipulated according to precise and formal rules. From the mathematical point of view soft computing means multi-valuednes or multivalence. Logical paradoxes and the Heisenberg uncertainty principle led to the development of multivalence, and in the 1930s quantum theorists allowed for indeterminacy by including a third truth value in the bivalent logical framework. Systems scientist Zadeh in 1965 introduced the term fuzzy into the technical literature, and inaugurated a second wave of interest in multivalued structures –from systems to topologies– extending a bivalent indicator function iA of non-fuzzy subset A of X to a multi-valued indicator or membership function mA: X→[0,1]. This allows to combine multi-valued or fuzzy sets with point-wise operators of indicator functions for the large variety of fuzzy systems.

SOFT COMPUTING SYSTEMS APPLIED FOR PREPROCESSING

Soft computing systems is a name for systems with directly relationship to soft computing concepts like fuzzy sets, neuronal nets, genetic algorithms etc. The soft computing concepts based on fuzzy sets can be classified into pure fuzzy systems, Takagi and Sugeno fuzzy systems and fuzzy systems with fuzzification and defuzzification.

Pure Fuzzy Systems

The basic configuration of a pure fuzzy systems is based on a fuzzy rule base that consists of a collection of fuzzy IF-THEN rules, and the fuzzy inference engine that uses these fuzzy IF-THEN rules in order to determine a mapping output universe of discourse $U \subset R^n$ to fuzzy sets in the output universe of discourse $V \subset R$ based on fuzzy principles. Fuzzy IF-THEN rules are of the following form:

$$R(k):\ IF\ x_1\ is\ F_1^{(k)}\ AND\ ...\ AND\ x_n\ is\ F_n^k\ THEN\ y\ is\ G^k \quad (1)$$

where $F_i^{(k)}$ and $G^{(k)}$ are the respective fuzzy sets, $x = (x_1,...,x_n)^T \in U$ and $y \in V$ are input and output linguistic variables, respectively, and k = 1,2,...,m.

Each fuzzy IF-THEN rule defines fuzzy set $F_1^{(k)}$ x...x $F_n^{(k)} \rightarrow G^{(k)}$ in the product space U x V. Let A be an arbitrary fuzzy set in U, then the output determined by each fuzzy IF-THEN rule of equation (1) is a fuzzy set A o $R^{(k)}$ in V whose membership function is

$$^{\mu}A^{(k)} \text{ o } R^{(k)}(\mu) = \sup\ x\ \mu U\ [\mu A^{(x)} * {}^{\mu}F_1^{(k)}\ x...x\ F_n^{(k)} \text{ --> } G^{(k)}\ (x,y)] \quad (2)$$

with * as operator such as MIN, MAX, PRODUCT, or others. μA is used to represent the membership function of a fuzzy set A.

The final output of a pure fuzzy system is a fuzzy set A o $(R(1),...,R(l)^{(k)})$ in V, a combination of the respective fuzzy set. Hence a pure fuzzy system constitutes the essential part of fuzzy systems as a general framework in which linguistic information is quantified and fuzzy principles are used to realize systematic use of linguistic information.

Takagi and Sugeno Fuzzy System

Instead considering fuzzy IF-THEN rules in form of equation (1), Takagi and Sugeno in 1985 proposed using fuzzy IF-THEN rules in the form:

$$L^{(k)}:\ IF\ x_1\ is\ F_1^{(k)}\ AND\ ...\ AND\ x_n\ is\ F_n^k\ THEN\ y^k = c_0^k + c_1^k x_1 + ... + c_n^k x_n \quad (3)$$

where $F_i^{(k)}$ are fuzzy sets, c_i are real-valued parameters, $y^{(k)}$ is the Takagi-Sugeno fuzzy system output due the rule $L^{(k)}$, and k=1,2,...m. That is, they considered rules whose IF part is fuzzy but whose THEN part is crisp. For a real-valued input vector $x=(x_1,..., x_n)^T$, the output y(x) of Takagi and Sugeno fuzzy systems is a weighted average of $y^{(k)}$:

$$y(x) = \frac{\sum_{\ell=1}^{k} w^{(k)} y^{(k)}}{\sum_{k} w^{(k)}} \qquad (4)$$

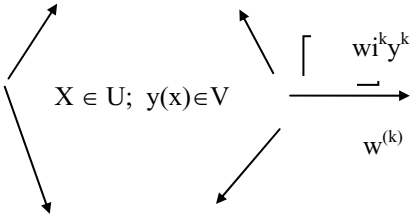
where weight $w^{(k)}$ implies the overall truth value of the premise of rule $L^{(k)}$ for the input and is calculated as

$$w^{(k)} = \prod_{i=1}^{n} \mu F_i^{(k)}(x_i) \qquad (5)$$

which is shown in the following representation.

$L^{(1)}$: IF X_1 IS F_1^1 AND...AND X_n IS F_n^1

THEN $y^1 = c_0^1 + c_1^1 x_1 + ... c_1^1 x_n$



$L^{(k)}$: IF X_1 IS F_1^k AND...AND X_n IS F_n^k

THEN $y^1 = c_0^k + c_1^k x_1 + ... c_1^k x_n$

Fuzzy Systems with Fuzzification and Defuzzification

Compared with the pure fuzzy system we may add a fuzzifier to the input and a defuzzifier to the output of the pure fuzzy system. The fuzzifier maps crisp points in U to fuzzy sets in U, and the defuzzifier maps fuzzy sets in V to crisp points in V. The fuzzy rule base as well as the fuzzy inference engine are the same as those in pure fuzzy logic systems, as shown in Figure 1.

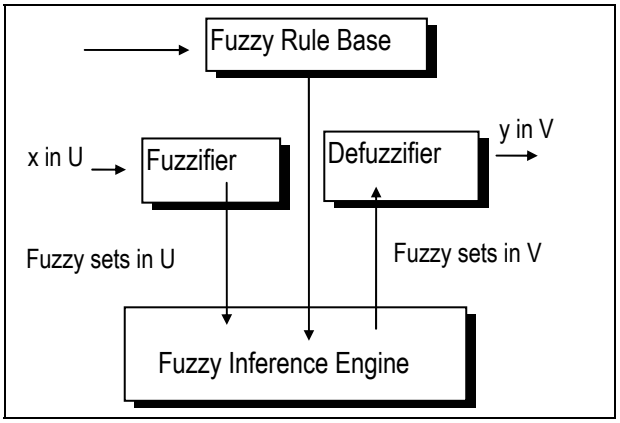


Figure 1: Fuzzy system with fuzzifier and defuzzifier

Neuronal Nets

A neural net consists of input variables and weighting factors, activation layers and output variables. The physiological pendant of the inputs are the dendrites as part of the anterior motoneurons extend for one-half to one millimeter in all directions from the neuronal soma. Therefore, these dendrites can receive signals from a fairly spatial area around the motoneuron. This provides vast opportunity for summation of signals from many separate presynaptic neurons. The weighting functions physiological pendant are the synapses. The synapse could be interpreted as the juncture between one neuron and the next, based on three major parts, the soma, which is the main body of the neuron; a single axon, which extends from the soma into the peripheral nerve; and the dendrites, which are thin projections of the soma that extend up to one millimeter, into the surrounding areas of the cord. The output has its physiological pendant in the axon, which is the central core of a nerve fiber. The biological neuron and the artificial neuron are shown in Figure 2.

From Figure 2.b one can assume that the synapses of an artificial neuron receive an activation x_i with a specific strength w_i from another artificial neuron, which will be part of the summing process of the output, the axon. The basic concept results in an input vector $\mathbf{x} = (x_1,...,x_n)^T$, a weighting vector $\mathbf{w} = (w_1,..,w_n)^T$ and the resulting activity as sum of the weighted input, which could be assigned as activity function z:

$$z(\mathbf{w},\mathbf{x}) = \sum_j \mathbf{w}_j \mathbf{x}_j = \mathbf{w}^T \mathbf{x}$$

Often there exists a threshold, which has to be passed, to activate the output. Modeling the threshold results in the relation

$$z(\mathbf{w},\mathbf{x}) = \mathbf{w}^T \mathbf{x} - T$$

where T indicates the threshold. Assuming

$$\mathbf{x} \dashrightarrow \mathbf{x} = (\mathbf{x}_1,...,\mathbf{x}_n,1)^T$$

and

$$\mathbf{w} \dashrightarrow \mathbf{w}\,(\mathbf{w}_1,...,\mathbf{w}_n,-T)^T$$

we receive the scalar product

$$z(\mathbf{w},\mathbf{x}) = \sum_j \mathbf{w}_j \mathbf{x}_j - T =$$

$$(\mathbf{w}_1,...,\mathbf{w}_n,-T)\,(\mathbf{x}_1,...,\mathbf{x}_n,1)^T = \mathbf{w}^T\mathbf{x}$$

which can be rearranged as follows

$$z = \mathbf{w}^{(0)} + \sum_i \mathbf{w}_i^{(1)} \mathbf{x}_i$$

whereby the notation$^{(k)}$ indicates the correlation's of the x components

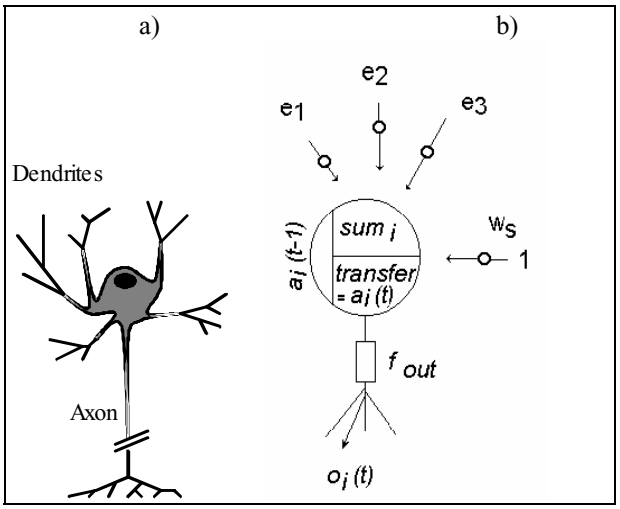


Figure 2: Biological neuron (a) and artificial neuron (b); for more details see text

Modeling high-order synapses then can be directly derived from the above equations as follows:

$$z = \mathbf{w}^{(0)} + \sum_{i} \mathbf{w}_{i}^{(1)} \mathbf{x}_i + \sum_{ijk} \mathbf{w}_{ijk}^{(2)} \mathbf{x}_j \mathbf{x}_k + \sum_{ijkl} \mathbf{w}_{ijkl}^{(3)} \mathbf{x}_j \mathbf{x}_k \mathbf{x}_l + \ldots$$

This type of artificial neurons are the so called sigma-pi-units.

The output of an artificial neuron can be described by the function S(.) depending on the internal activity z,

$$y = S(z)$$

The complete output of an artificial neuron hence can be stated as transfer function of type y

$$y = f(\mathbf{x}, \mathbf{w}, z, S).$$

THEORY OF SENSITIZATION AS PART OF PRE-PROCESSING

The idea of Sensitization can be found in cognitive psychology in the context of chunking. Chunking is more or less the adaptation of a new fact or a so far unknown situation with the help of knowledge facts or models. Only out of old facts or acting strategies one can develop new strategies for understanding of so far unknown. Transforming this idea to the handling with neural nets means, that first a net has to learn a basic concept. To prevent that the net used includes typical output ranges in its classification behavior due to the measured data for analysis, and therefore looks for measures with a high output, it is necessary to normalize the input data set by using an appropriate-processing method. To handle changes in the global state of the case study it is favorable to use a pre-processing step that results in a gradient vector that is calculated by the difference between the data when no pathological situation is present minus the present actual data.

By presenting the weaker states after the basic concept is settled it can be ensured that the net will be forced to change its classification structure slightly out of it's former structure, without destroying the older structure. This means that the net will be sensitized. Especially when a back propagation network is used the learning rules force the net to sensitize its structure in such a way that only the case study specific state representing structure is modified, as the classification results have to be the same over the whole sensitization period. If weaker and weaker states will be presented successively the classification structure will change accordingly, until the similarity of the different evolutionary state representations will be so little, that the net can not be forced to change it's structure anymore.

Figures 3.a and 3.b show the sensitized net working structure. Since the classification potential is changed locally, the net changes its classification behavior not in general by learning the evolutionary data sets, but shows the according adaptive behavior. This local change surely can lead to the unification of so far divided concepts, a fact which will open the door to a wide range of so far unknown or unnoticed intrinsic relations of the micro array data sets, representing the case study states.

Henceforth, sensitization in case of micro array data analysis can be introduced as intelligent pre-processing for clustering analysis that means normalization and filtering, that is necessary due to

Henceforth, sensitization in case of micro array data analysis can be introduced as intelligent pre-processing for clustering analysis that means normalization and filtering, that is necessary due to

- Systematic experimental errors,
- Uneven hybridization gel,
- Background variations,
- Wavelength dependency,
- Intensity dependency.
- Image processing algorithm-dependency
- Etc.

Hence, the importance of using intelligent pre-processing algorithms is really based on the hypothesis underlying micro array analysis that the measured intensities for each arrayed gene represent its relative expression level. However, before the levels can be appropriately compared, one generally performs a number of transformations on the data to eliminate questionable or low quality data, to adjust measured intensities to facilitate comparisons, and to select those genes that are significantly differentially expressed, which explains the need for a pre-processing methodology beyond.

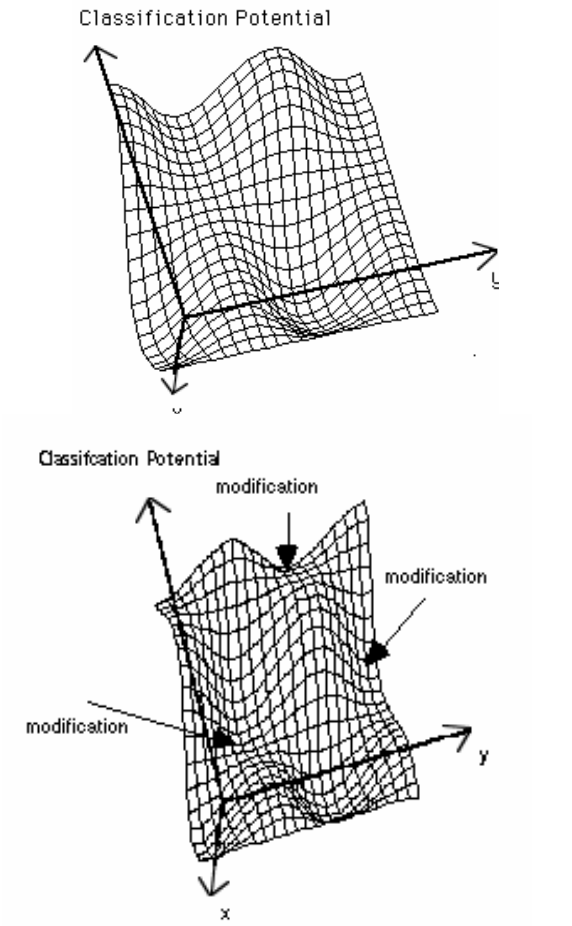


Figure 3: Classification-potential a) E before sensitization and b) E' after sensitization

The sensitized neuronal nets classifier in general is able to separate all trained states representing a powerful concept of weaker evolutionary states of different case study situations, to be trained

Figure 4 shows the time differences between an early warning of a common neural classifier and a sensitized neural network. Both nets have had the same warning criteria, setting an alarm when the probability for an pathological state is higher then 85%. It can be seen that a sensitized neural classifier is able to decrease the alarm time by a factor 5, as the net classifies the evolutionary state of the begin of an auricular fibrillation rather early.

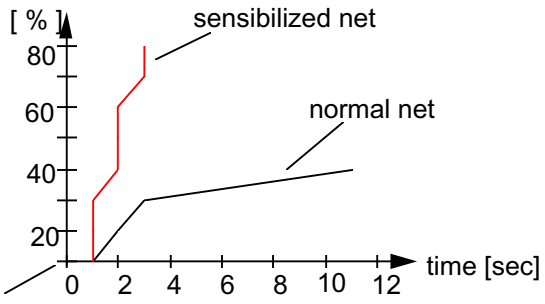


Figure 4: Difference between early warning of a sensitized and a normal neural net

In case of micro array data analysis sensitized nets can be developed for usage in

- Total Intensity normalization,
- Iterative linear regression normalization,
- Standard deviation regularization,
- Cross Slide Replicates T-test,
- Signal/Noise checking,
- Slice analysis,
- Etc.

As an measure example one can assume that two factors contributed to the gene X, the biological factors one is interested in, and experimental factors, one is not interested in. This requests for a possibility to extract the experimental factors which can be done initializing the pre-processing task of the statistical analysis. The statistical analysis behind can be

- Pre-processed local linear regression model,
- Pre-processed least squares,
- Etc.

To handle changes in the global state of cases it is favorable to use a pre-processing step that results in a gradient vector that is calculated by the difference between the data of a case study as requested when no pathological situation is present minus the present actual data.

Procedure of sensitized gene analysis in micro arrays

- Define a gene array window
- Sensitize window along log(IntensityProduct) axis
- Calculate logRatioMean and logRatioSD of gene data of the pre-processed samples
- Calculate Z-scores of each gene data point
- Z-score = (logRatio-logRatioMean)/ logRatioSD
- Trim data with Z-scores beyond interested range

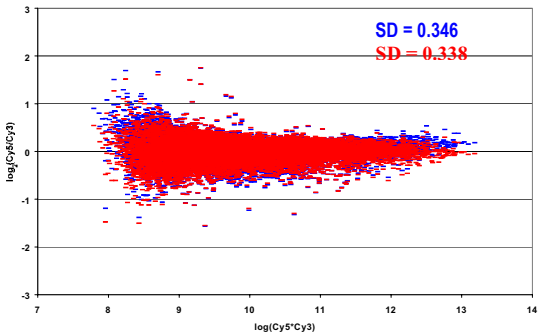


Figure 5 Sensitized Gene Expression Analysis

EXPRESSION ANALYSIS AND WORKFLOW ANALYSIS

Workflow management systems provide the foundation for defining and executing business processes. A combined workflow process is embedded to overcome the

individual influences on data. At present there are no standardised methods available that can be used as a workflow-based quality assurance system.

- data generation,
- data handling,
- data processing,
- data interpretation, manually as well as automated.

in order to coordinate the information, based on a process control view, that includes the integration of data, documents, interpretation, execution of work and computer assisted work such as statistics. This can be achieved by developing a methodological concept for data management and data analysis, which has to be workflow based, embedded in business objects. While using the method of business objects, as shown for example in Fig. 6., the multiple steps of expression analysis or CGH will be adapted and standardised. A business object is defined as a representation of a thing active in the business domain, including at least its business name and definition, attributes, behavior, relationships, rules, policies and constraints.

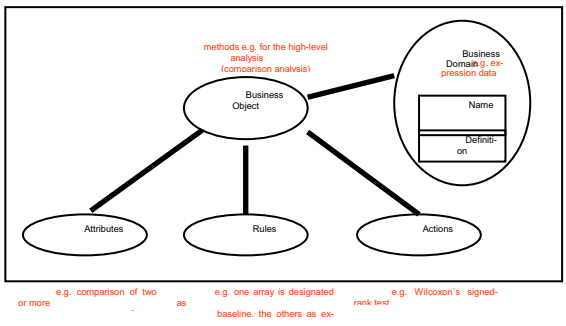


Figure 6: Business Object Concept for Gene Expression Analysis

The fundamental benefits inherent to workflow-based applications are

- Flexibility in changing the model of the underlying business process
- Integration capabilities for even disparate applications
- Reusability of activity implementations and process models
- Scalability of application development and execution

Henceforth the workflow forms the dataflow of the business objects which represent the name, definition, attributes, behavior, relationships, rules, policies and constraints of the gene expression algorithms. The fundamental benefits inherent to business objects in sensitized genes expression analysis are

- Flexibility in changing the model of the underlying business object
- Integration capabilities for even disparate applications
- Reusability of activity implementations and business objects
- Scalability of application development and execution due do the object oriented programming paradigm of business objects

CONCLUSIONS

The potential of soft computing workflow analysis as well as business objects for micro array data analysis is huge. We only scratched the surface of the complex due to a brief view insight possible medical application domains. The potentially of soft computing and preprocessing contains an incredible number of solutions to the several problem depending domains.

REFERENCES

Kaufmann, A., Gupta, M. M. *"Introduction to Fuzzy Arithmetic"*, Van Nostrand Reinhold, 2001

Kosko, B. *"Neural Networks and Fuzzy Systems"*, Prentice-Hall International, 2002

Möller, D.P.F. "Soft Computing Methods", In: *Mathematical and Computational Modeling and Simulation,* Springer Verlag, Heidelberg, 2003

Takagi, H., Sugeno M. "Fuzzy Identification of Systems and its Applications to Modelling and Control", *IEEE Trans. on Man, and Cybern.*, SMC-15(1), 1985, pp. 116-132

Wang, L. X. *"Adaptive Fuzzy Systems and Control"*, Prentice Hall International, 1994

Zadeh, L. A. „Fuzzy Sets", *Informat. Control, Vol. 8*, 1965, pp. 338-353

AUTHORS

DIETMAR P. F. MÖLLER was born in Preetz, Germany. He enrolled at the Universities of Lübeck, Bremen Mainz and Bonn, where he studied Electrical Engineering and Human Medicine. He obtained his doctoral degree in 1980. From 1985 to 1991 Dr. Möller lead the anaesthesia division of Dräger AG in Lübeck, Germany. 1991 he has been elected as Full Professor for Computer Engineering (TI) at the Technical University of Clausthal. In 1998 he has been elected as Full Professor for Computer Engineering Systems (AB TIS) at the University of Hamburg where he holds the chair for Computer Engineering. Since 1998 he also is Adjunct Professor at the California State University Chico. His E-Mail address is dmoeller@informatik.uni-hamburg.de and his Web-page can be found at http://www.informatik.uni-hamburg.de/TIS/