Self-Organizing Maps and its Applications in Sleep Apnea Research and Molecular Genetics

Gabriela Guimarães[1] and Wolfgang Urfer[2]

[1] CENTRIA, Universidade Nova de Lisboa,
and Department of Computer Science, Universidade de Évora
Portugal

[2] Department of Statistics, Universität Dortmund
Germany

Abstract

This paper presents the application of special unsupervised neural networks (self-organizing maps) to different domains, as sleep apnea discovery, protein sequences analysis and tumor classification. An enhancement of the original algorithm, as well as the introduction of several hierachical levels enables the discovery of complex structures as present in this type of applications. Furthermore, an integration of unsupervised neural networks with hidden markov models is proposed.

Keywords: Unsupervised Neural Networks, Hidden Markov Models, Sleep Apnea, Protein Sequences, Tumor Classification

1 Introduction

The development of more and more powerful computers in recent years has lead to a recording of a great amount of data gathered from, for example, industrial processes, medical applications, meteorological phenomena, etc. Artificial neural networks (ANNs) and methods from statistics are particularly interesting for handling such noisy and inconsistent data. The application of ANNs and statistics often refers to problems of discrimination (supervised learning) or to clustering problems (unsupervised learning).

Self-organizing Maps (SOMs) as proposed by Kohonen(1982) are well suited for the discovery of patterns in high dimensional data, i.e. clustering problems (Kohonen (1995), Kaski and Kohonen (1996)). In addition, SOMs have also been successful in applications, where temporal or sequential data are processed, for instance, in speech recognition, process control and time series analysis in medicine (Behme et al. (1993), Walter and Schulten (1993), Guimarães (2000)).

In this paper we give a review of SOMs to several application domains, such as sleep apnea, protein sequence analysis and tumor classification.

For the diagnosis of sleep apnea the temporal dynamics of physiological parameters such as respiration and heart rate, have to be recorded and evaluated. In order to perform an automated identification of sleep apnea, a simultaneous analysis of all signals is needed. Different types of sleep apnea diseases represent complex patterns in the time series that occur during one night. Those patterns may differ strongly, even for the same patient.

The main aim of statistics in bioinformatics is the development and application of methods for the analysis of genomic data in order to elucidate biological processes. Statistics and ANNs play a major role in diagnosing diseases and developing new drugs (Brunnert et al. (2000)). The most important contribution of statistics has been the development of strategies for extracting information from DNA and protein sequence databases by sequence comparison, characterization and classification.

These applications have in common that complex patterns are searched for and a hierarchical segmentation of the problem is needed introducing hierarchical SOMs. In addition to the hierarchical component, both applications demonstrate a temporal or a sequential component.

In section 2 SOMs and their possible extensions are introduced. Section 3 presents the application of extended SOMs to sleep apnea. The application of extended SOMs to protein sequence analysis and tumor classification is shown in section 4. Finally, a conclusion and the extension of the approach to other methods from statistics, as Hidden Markov Models, is presented in section 5.

2 Self-Organizing Maps for Exploratory Data Analysis

Artificial Neural Networks (ANNs) may be classified according to their learning principles mainly into two different types: ANNs with supervised learning and ANNs with unsupervised learning.

ANNs with supervised learning adapt their weights to a given input-to-output relationship, for instance, for the recognition of images representing handwritten character. Here, the character type (class) is already known, and an association of the handwritten character to the corresponding character type is searched for, in order to predict new handwritten characters. Often some kind of noise is present in the data, such that the performance of the classification system mainly depends on the chosen features and the complexity (number of free parameters) of the model. For this kind of pattern recognition problems, ANNs with supervised learning, such as Feed-forward Networks or Radial Basis Function Networks, have been widely studied in relation to their statistical properties. It is well-known that Feed-forward Networks can approximate, to arbitrary accuracy, any smooth function. In the context of classification problems, Feed-forward Networks with sigmoidal non-linear activation functions of the neurons can approximate arbitrarily any decision boundary. Such ANNs provide universal non-linear discriminant functions modeling posterior probabilities of class membership, permitting a probabilistic interpretation of the results (Bishop (1995)). Models in statistics strongly related to those ANNs are logistic discrimination functions, projection pursuit regression, and multivariate adaptive regression splines.

In this work, however, we will not focus on pattern recognition problems, where a classification is known a priori, but on pattern discovery problems, where the inherent patterns in the data are searched for. ANNs with unsupervised learning are suitable for such problems, since they adapt their internal structures (weights) to the structural properties (e.g. regularities, similarities, frequencies, etc.) of high-dimensional input data. ANNs like ART (Adaptive Resonance Theory) and Self-Organizing Maps (SOMs) belong to this type and, specially, the latter are well-suited for clustering (Kaski and Kohonen (1996), Ultsch and Siemon (1990)).

The motivation of SOMs is strongly biology-oriented, where biological principles, the generation of topographical maps in the brain through self-organization, play an important role. In the following, the learning process will be described from a more algorithmic point of view. During learning SOMs adapt their weights such that a n-dimensional input space is projected onto a m-dimensional map with $m<n$, preserving the neighborhood of the input data on the map. Usually, a two-dimensional map is chosen. The map is formed by the properties inherent to the data itself. Consequently, no previous classification of the data is needed. The input layer has n units representing the n components of an input vector $x_k = (x_{k1},..., x_{kn}), k = 1,..., N$. The output layer is a two dimensional array of units arranged on a map. Each unit in the input layer is connected to every unit in the output layer with a weight $w_i = (w_{i1},..., w_{in}), i = 1,..., p \cdot q$ associated. All weights are initialized randomly. They are adjusted according to Kohonen's learning rule

$$\Delta w_i = \eta(t) \cdot h_{ir}(t) \cdot (x_k - w_i) \tag{1}$$

that uses a distance measure $\| \mathrm{w}_\mathrm{r} - x_k \| = \min_i \| \mathrm{w}_\mathrm{i} - x_k \|$ to determine the bestmatch r and a neighborhood function

$$h_{ir}(t) = e^{-\frac{(i-r)^2}{2\sigma(t)^2}} \tag{2}$$

that realizes the lateral inhibition. The learning rate determines the strength of learning with $0 < \eta(t) < 1$. The radius $\sigma(t)$ determines the set of neurons in a neighborhood of the bestmatch that are included into the learning process. Both functions usually decrease monotonously during learning.

On the map neighboring units form regions that correspond to similar input vectors. These neighborhoods form disjoint regions, thus enabling a classification of the input vectors. However, in order to perform a classification, a visualization of the network structures is needed, since the Kohonen algorithm converges to an equal distribution of the units on the output layer (Kohonen (1995)). Therefore, a three dimensional landscape, called U-Matrix (Ultsch and Siemon (1990)), is generated representing structural properties of the high dimensional input space on the map. At each point of the grid the weights are analyzed with respect to their neighbors. The distance between the weights of two neighboring units then is displayed as height into the third dimension. A U-Matrix has valleys where the vectors on the map are close to each other and represent data that are in the same class. Hills or walls represent larger distances indicating dissimilarities of the input data (see Fig. 1). Such a visualization of a SOM can be used for clustering, since similar input vectors are close together on the map and fall into the same valley, i.e. cluster. In the last years SOMs together with the U-Matrix method have been successfully applied to a wide-ranging number of applications where a clustering of high-dimensional data is intended (Kaski and Kohonen (1996), Kohonen (1995), Ultsch et al. (1997)).

The discovery of complex patterns, for instance, in multivariate time series, protein sequences, and genes with SOMs is much more complex, since it demands an improvement and extension of the original SOM. Therefore, SOM with several hierarchical layers are introduced, in order to capture and discover structures at different abstraction levels. This is necessary, when a segmentation of complex and structured problems is needed, for instance, in such areas as image recognition (Koh et al. (1995)), temporal pattern discovery in multivariate time series (Guimarães (2000)), speech recognition (Kemke and Wichert (1993)), and protein sequences analysis (Andrade et al.

(1997)). In addition, for applications with temporal or sequential data a visualization of trajectories on a map enables the monitoring of such phenomena, for instance, for the recognition of misarticulations in speech (Mujunen et al. (1993), EEG signal monitoring (Joutsiniemi et al. (1995)) and sleep apnea detection (Guimarães and Ultsch (1999)).

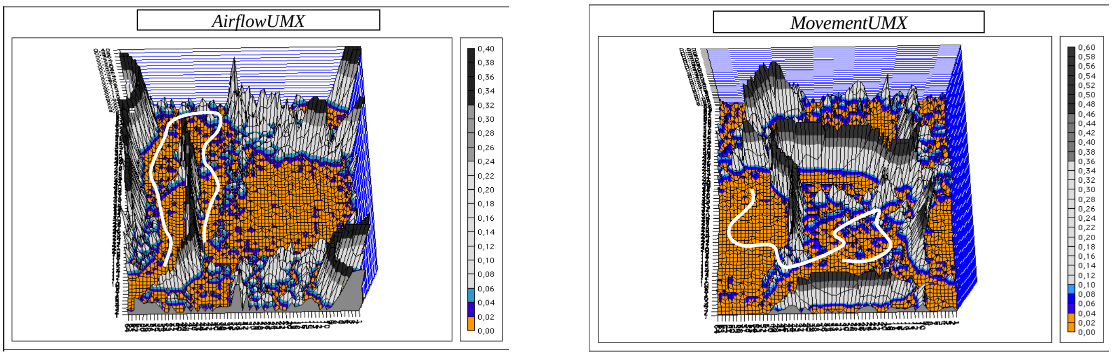

Fig. 1. U-Matrices (*AirflowUMX*: U-Matrix from features related mainly to airflow and *MovementUMX*: U-Matrix from features related mainly to respiratory movements) for Primitive Patterns obtained from multivariate time series of an application in medicine, called sleep apnea.

3 Detection of Sleep-related Breathing Disorders

In this section, we introduce a method for Temporal Knowledge Conversion, named TCon (Guimarães (1998), Guimarães (1999)), that enables the discovery of temporal patterns in multivariate time series. The main idea lies in introducing several abstraction levels, such that a step-wise and successive detection of the temporal patterns becomes possible, breaking down this highly structured and complex problem into several sub-tasks. This method also performs a transition of temporal patterns in multivariate time series into a linguistic representation form in form of temporal grammatical rules, intelligible and understandable for human beings such as domain experts.

Fig. 2 shows the main steps of the method TCon. Multivariate time series $Z = \vec{x}(t_1),...,\vec{x}(t_n)$ with $\vec{x}(t_i) \in \mathrm{R}^m, m > 1$ sampled at equal time steps $t_1,...,t_n$ gathered from signals of complex processes are the input of the system. Results are the discovered temporal patterns as well as a linguistic description of the patterns, interpretable for human beings. An overview of the method is given by Guimarães and Ultsch (1999).

Features: First of all, a pre-processing and feature extraction for all time series is a pre-requisite for further processing (Bishop (1995)). For the feature extraction one or even more than one time series $\vec{x}_S(t_i) = x_{j_1}(t_i),..., x_{j_S}(t_i)^T \in \mathrm{R}^S$ may be selected from the multivariate time series Z with $j_k \in S, k = 1,...,s$, $S \subset \{1,...,m\}, s = |S|$. A feature $m_S(t_i, l) = f(\vec{x}_S(t_i),..., \vec{x}_S(t_{i+l}))$ then is the value of a function $f : \mathrm{R}^{s \times l} \to \mathrm{R}$ at time t_i with $l \in \{1,..., n-l\}$ from selection *S*. In order to find a suitable representation of all time series, methods, for instance, from signal processing, statistics or fuzzy theory, can be used.

Primitive Patterns: Second, exploratory methods, in particular, SOMs together with the U-Matrix-method (see Fig. 1) are used for the discovery of elementary structures in the time series, named as *primitive pattern classes* p_j, $j = 1,..., k$. An element of a primitive pattern class is a primitive pattern $p_j(t_i)$, $j = 1,..., k$ that belongs to a given primitive patterns class p_j, $j=1,...,k$ and is associated to a given time point t_i. Regions on a U-Matrix that do not correspond to a specific primitive pattern class are associated to a special group, named *tacet*. We are now able to classify the whole features with primitive patterns and tacets. This will be called a primitive pattern (PP)-channel. Instead of analyzing all time series simultaneously, several selections of features are made. Consequently, several SOMs are learned (see Fig. 2). This leads to more than one PP-channels (see Fig. 3). At this level, machine learning algorithms may be used, in order to generate a rule-based description of the primitive pattern classes (see also Fig. 2).

Successions: In order to consider temporal relations among primitive patterns, succeeding identical primitive patterns $p_j(t_i),..., p_j(t_{i+k})$, $i=1,...,n-k+1$ obtained from each SOM are identified as *successions*. Since several feature selections are possible, successions from different PP-channels may occur more or less simultaneously. Each *succession* $s_j(a,e)$ is associated to a given primitive pattern class and has a starting point $a:=t_i$ an end point $e:=t_{i+l}$ and, consequently, a duration $l=e-a$. Since each primitive pattern is represented through its bestmatch on a U-Matrix, trajectories of succeeding primitive patterns (bestmatches) on a U-Matrix are used for the identification of successions.

Events: More or less simultaneous occurring successions $s_1(a_1,e_1),...,s_q(a_q,e_q)$ that occur more than once are identified as an *event* $e(l)$. Then $A=\max(a_1,...,a_q)$ is the starting point, $E=\max(e_1,...,e_q)$ the end point and $l=E-A$ the duration of the event $e(l)$. Each event belongs to a given event class. In order to reduce the great amount of information, a vague simultaneity is introduced.

In addition, the significance of events (frequency of the occurrence of events) is calculated using conditional probabilities between the occurrence of simultaneous primitive patterns on different PP-channels. Histograms over the calculated probabilities enable a differentiation between significant events (very frequent events) and less significant events (less frequent events). Rare events are omitted in the sense as they are regarded as delays between events, named as event tacets. In order to join events with different significance levels, very frequent events are associated to less significant events. Therefore, similarities among significant and less significant events will be considered counting the number of equal types of successions occurring in both events. This results in an extremely reduced number of events. Consequently, each event is described by one significant event and, possibly, one or more than one less significant events. At this level, the whole multivariate time series is described by a sequence of events $F=e_{1_j},...,e_{n_j}$, $j=1,...,m$. In order to identify events with SOMs, extended hierarchical SOMs have to be used (Guimarães (2000)). For each event a temporal grammatical rule is generated (see Fig. 2).

Sequences: Subsequences of events $e_i,...,e_k$ that occur more than once in F are identified as a sequence $sq(\min,\max)=e_i(\min_i,\max_i),...,e_k(\min_k,\max_k)$. This means that sequences are repeated subsequences of the same type of events at different time points t_i. Since events may succeed immediately or after a time delay, i.e. an event tacet, the duration of event tacets can be used for determining the starting event or/and the end event of a sequence. This is possible, if the duration of event tacets is regarded as a transition between different sequences due to larger delays between succeeding events. In addition, probabilistic automata, can be also used for the identification of sequences. Probabilistic automata describe transition probabilities between events such that paths through such an automata describe probable subsequences of events. For each sequence a temporal grammatical rule is generated (see Fig. 2).

Temporal Patterns: Finally, small variations in the events of each sequence type lead to the identification of similar sequences. Similar sequences $sq_i(\min_i, \max_i) \vee ... \vee sq_v(\min_v, \max_v)$ will be joined together to a *temporal pattern* $tp(\min, \max)$, where $\min = \min(\min_i, ..., \min_v)$ and $\max = \max(\max_i, ..., \max_v)$. Temporal patterns are abstract descriptions of the main temporal structures in multivariate time series. String exchange algorithms are suitable for the identification of temporal patterns. For each temporal pattern a temporal grammatical rule is generated (see Fig. 2).

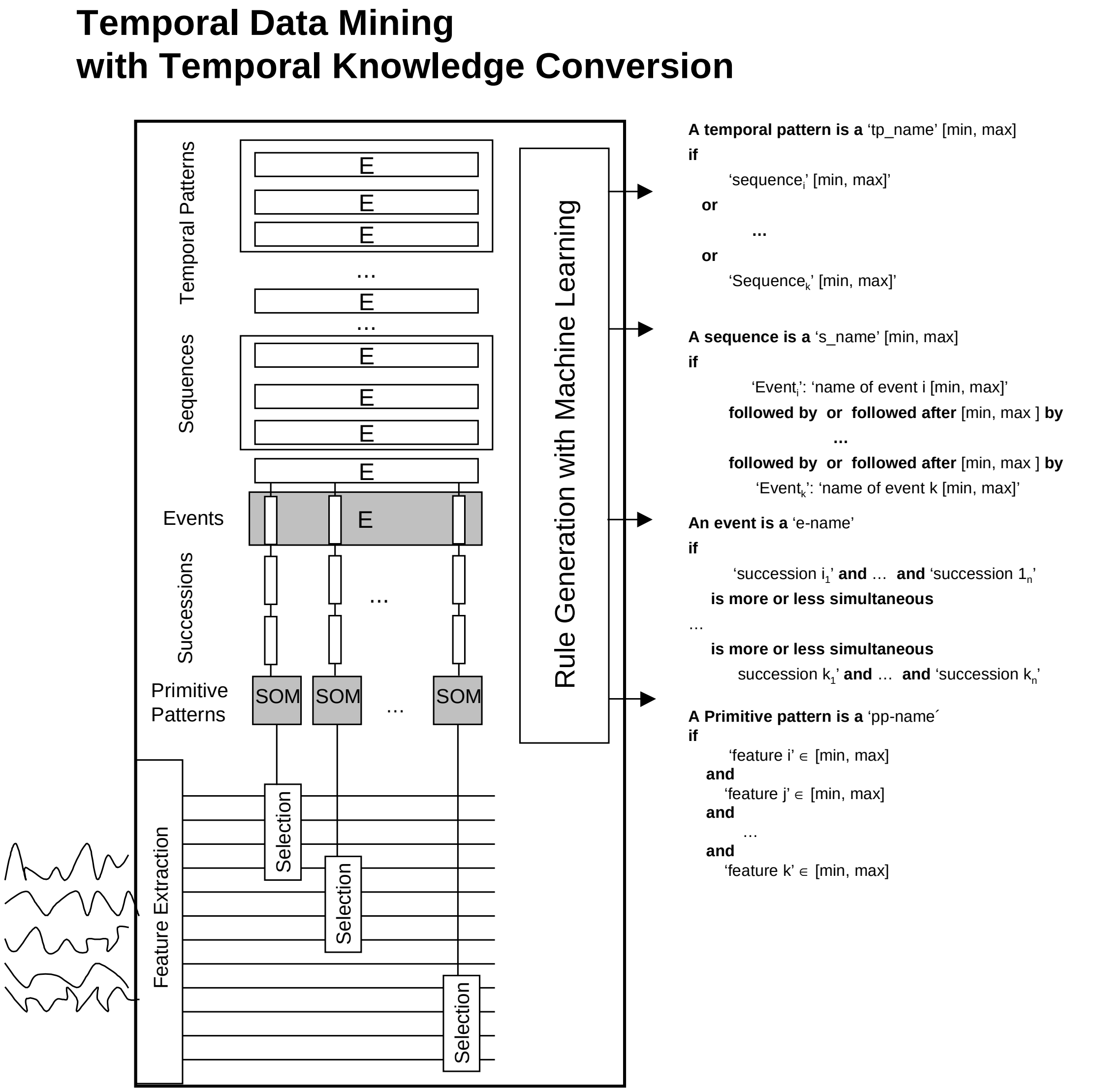

Fig. 2. Abstraction levels and steps of the method for Temporal Knowledge Conversion (TCon)

This approach was applied to an example in medicine, namely sleep-related breathing disorders (SRBDs), consisting in various types among which sleep apnea is best known (Penzel and Peter (1992)) (see Fig. 3 for a cutout from a recording of one patient). For the diagnosis of sleep apnea the temporal dynamics of physiological parameters such as respiration and heart rate, have to be recorded and evaluated. For an analysis of sleep apnea, a large number of parameters are involved, such as sleep-related signals (EEG, EOG, EMG), signals concerning the respiration (airflow, ribcage and abdominal movements, oxygen saturation, snoring) and circulation related signals (ECG, blood pressure).

For the identification of different types of SRBD, mainly apnea and hypopnea, just the signals concerning the respiration had to be considered (Peter et al. (1998)). Severity of the disorder is calculated by counting the number of apnea events per hour of sleep. The sum of the index of apneas and hypopneas is a measure for the respiratory disturbance index (RDI). It can be seen as pathological, when the RDI exceeds 20 events per hour of sleep, while patients with more than 40 events per hour of sleep have to be referred to therapy.

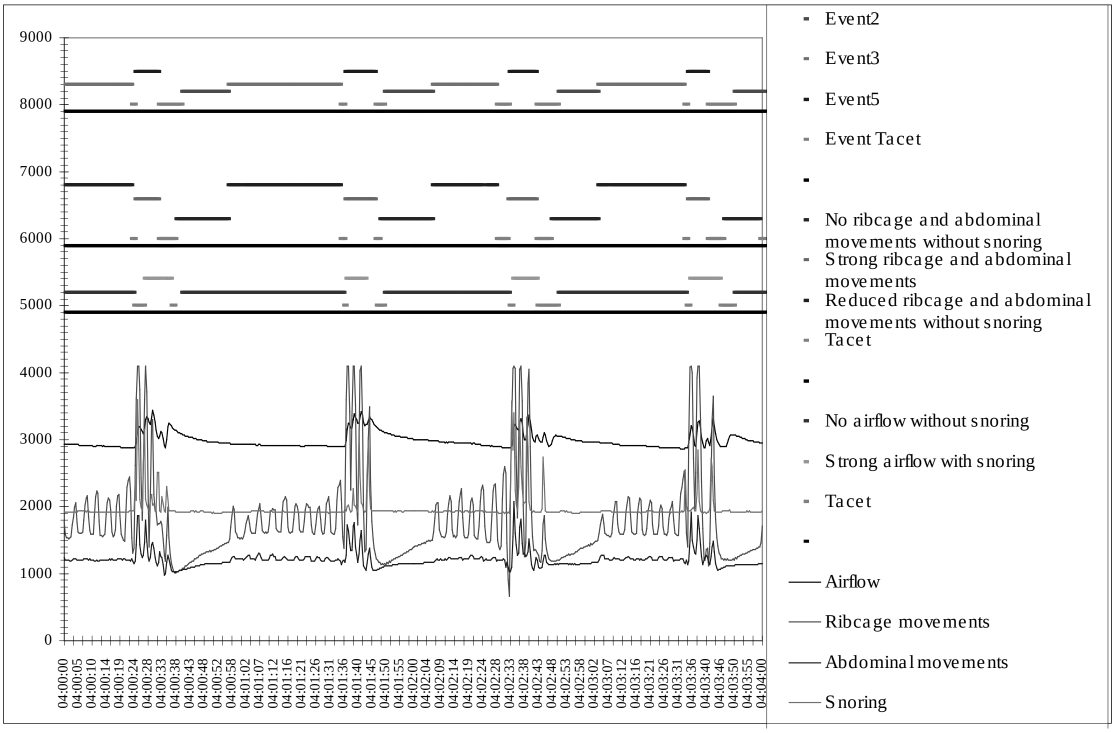

Fig. 3. Multivariate time series and resp. primitive patterns/successions from a patient with SRBDs

Technical assistants usually make the visual classification of the different types of SRBDs based on such a recording. An automatic identification of SRBDs is a quite hard task, since a simultaneous analysis of all signals is needed. In addition, quite different patterns for the same SRBD may occur, even for the same patient during the same night, and a strong variation of the duration of each event may occur, as well.

SRBDs can be subdivided into SRBDs with and SRBDs without an obstruction of the upper respiratory tracs. The different kinds of SRBDs are identified through the signals ´airflow´, ´ribcage movements´ and ´abdominal movements´, ´snoring´ and ´oxygen saturation´, where a distinction between amplitude-related and phase-related disturbances is made. Concerning the amplitude-related disturbances, we distinguish disturbances with 50% as well as disturbances with 10-20% of the baseline signal amplitude. Phase-related disturbances are characterized by a lag between ´ribcage movements´ and ´abdominal movements´. An interruption of ´snoring´ is present at most SRBDs as well as a drop in ´oxygen saturation. 25 Hz sampled data from three patients having the most frequent SRBDs (altogether 27 patterns) have been used. No additional information was provided from the medical experts, since the main aim is to discover inherent structures in multivariate time series using unsupervised methods, such as SOMs.

A structured and complete evaluation of the discovered temporal knowledge at the different abstraction levels was made using a questionnaire. All events (six) and temporal patterns (four) consisting in six different sequences (see Fig. 4) presented to the medical expert described the main properties of SRBD as, for instance, ´hyperpnoe´, ´obstructive snoring´, ´obstructive apnoe´ or ´hypopnoe´. The generated temporal grammatical rules described very well the domain knowledge. An evaluation of the rules at this level lead to an overall sensitivity of 0,762 and a specificity of 0,758. ´Event5´ was correctly identified as a special event, called ´hyperpnea´. SRBDs always end up with a ´hyperpnea’. In some cases the duration of ´Event5´ was too short. The duration of all other events were in a valid range. For one of them even previously unknown knowledge was discovered. This temporal pattern was named by the expert as ´mixed obstructive apnoe´, distinguished into a ´mixed obstructive apnoe´ with an interruption and snoring having a ´central´ and an ´obstructive´ part and a ´mixed obstructive apnoe´ without an interruption and without snoring ending in an ´hypoventilation´.

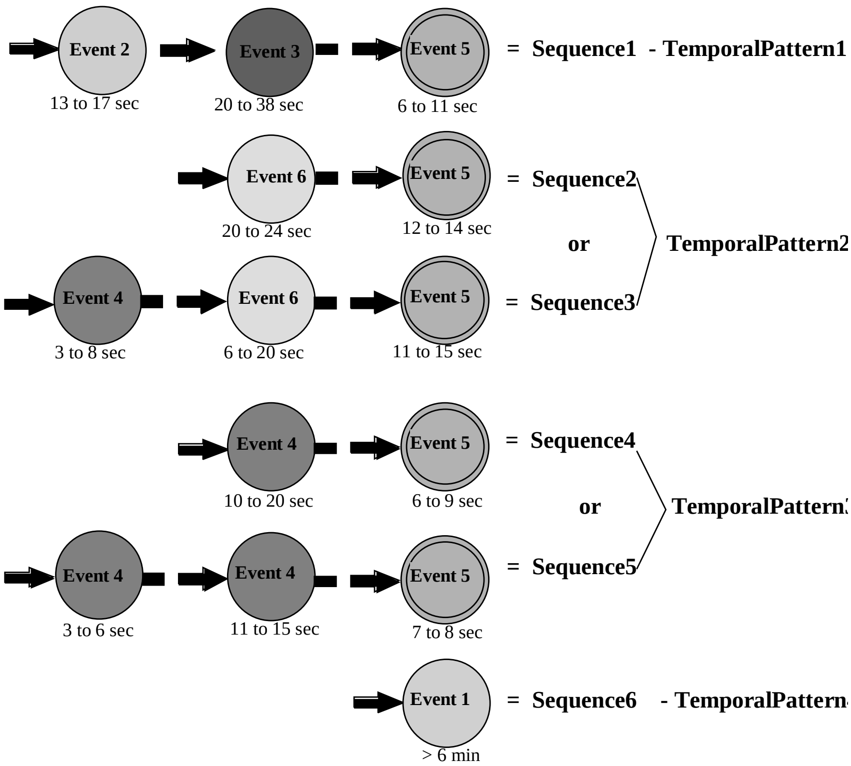

Fig. 4. Temporal Patterns with corresponding sequences and events for all SRBDs

4 Classification of Protein Sequences and Tumors

A self-organizing map (SOM) can be used to classify sequences within a protein family (ras-p 21 family) into subgroups that correspond to biological subcategories. Andrade et al. (1997) present a modified SOM-algorithm and use the rab family of small guanosine-triphosphate-ases to illustrate the performance of the method.

In their approach each of N protein sequences is binary coded as a sequence vector (input vector) $f_k = (f_{k1},..., f_{kn}), n = 20 \cdot L, k = 1,..., N$. Each position of the sequence is described by 20 components corresponding to all possible 20 amino acids *A,C,D,E,...,S,T,V,W,Y*. The component corresponding to the amino acid type at this position is coded by ‚one‘, and the rest of the components are set to zero. The resulting sequence vector f_k has length $20 \cdot L$, where L is the length of the sequence alignment.

The SOM is a two-dimensional layer of *pxq* units with one weight $w_i = (w_{i1},...,w_{in}), n = 20 \cdot L, i = 1,..., pxq$ for each of the *pxq* units. The weights have the same number of components as the sequence vectors f_k and their components take real values between zero and one. At zero time the weights w_i, $i=1,\ldots,p \cdot q$ are the mean of all sequence vectors. The distance $\delta_{i,k}$ from weight w_i to the sequence vector f_k is given by

$$\delta_{i,k} = \sqrt{\sum_{n=1}^{20 \cdot L} \left| f_{k,n} - w_{j,k} \right|^2} \tag{3}$$

where $n=1,\ldots,20 \cdot L$ is the index of the vector components.

The bestmatch is identified by having the smallest distance to the sequence vector. This vector is updated with a linear combination of its previous value with the presented sequence vector as follows:

$$w_i(t+1) = (1-\alpha^o)\, w_i(t) + \alpha^o f_k \tag{4}$$

where α^o is a factor that sets the weight given to the example sequence in the updating step. The update makes the weight more closer to the example presented to the network. The examples are presented to the system in random order, once for each training cycle. Then the time devoted to all cycles is $s \cdot N$, where N is the number of examples (protein sequences) and s is the number of learning epochs. These procedure adds noise to the dynamics of the weight evolution, which helps the system to avoid non-optimal classification.

However, a single SOM just leads to the clustering of the family at a definite resolution level. Only several SOMs with several resolutions enable the identification of a sequence relationship not existant in a single map, that means a hierarchy of sequences in the family. Therefore, a set of experiments with SOMs having different sizes are arranged in a tree-like fashion through a linkage of the clusters that contains the same sequences at successive levels. Such a tree representation can be compared with phylogenetic trees that try to accommodate the evolutionary relationsships of a group of sequences in a tree according to their sequence homology.

Andrade et al. (1997) analyzed 42 proteins of the rab family and showed the power of the SOM to obtain a reliable classification that agrees with the classifications obtained by phylogenetic trees.

A recent application is given by the molecular classification of tumors using SOMs. Specific cancer treatments (chemotherapy) try to maximize efficacy and minimize toxicity. Therefore, improvements in cancer classification are important to advances in cancer treatment therapies. Golub et al. (1999) described a statistical approach to cancer classification based on gene expression monitoring by DNA micro-arrays. Cancer classification is divided into class discovery and class prediction. Class discovery refers to defining previously unrecognized tumor subtypes and entails two issues:

1. developing algorithms to cluster tumors by gene expression and
2. determining whether putative classes produced by such clustering algorithms reflect the true structure in the data.

Golub et al. (1999) applied a two-cluster SOM to group 38 initial leukemia samples into two classes on the basis of the expression pattern of 6817 genes. This class discovery technique can be used to identify fundamental subtypes of any cancer.

5 Conclusions

In this paper, a survey of the application of extended SOMs to sleep apnea, protein sequence analysis and tumor classification was given. Therefore, a hierarchical segmentation of the problem using hierarchical SOMs is needed. In addition, both types of applications demonstrate a temporal or a sequential component, that could be investigated with statistical methods, such as hidden markov models (HMMs).

HMMs and the EM-algorithm are alternative statistical methods for the characterisation of a protein family. Following Krogh et al. (1994) we consider a family of protein sequences that all have the same three-dimensional structure. An HMM is thought to be able to generate protein sequences by a random process. The core of the HMM consists of M so called match states in corresponding to positions in a protein. Each of these match states m_k, $k=1, ..., M$ generates an amino acid x from the 20-letter amino acid alphabet according to the distribution $P(x|m_k)$, $k=1, 2,...., M$. For each match state m_k there is a delete state d_k that is used to skip m_k. There are also a total of $M+1$ insert states i_k which generate amino acids according to probability distributions $P(x|i_k)$. From each state, there are three possible transitions to other states. The transition probability from state q to state r is denoted by $T(r|q)$. We can generate a sequence of amino acids $x_1, x_2, ..., x_L$ by following a path of states $q_0, q_1, ..., q_N, q_{N+1}$, where $q_0=m_0$ is the begin state and $q_{N+1}=m_{N+1}$ is the end state. If q_i is a match or insert state, we define $l(i)$ to be the index in the sequence $x_1, x_2, ..., x_L$ produced in state q_i.

The probability of the event that the path $q_0, q_1, ..., q_{N+1}$ is taken and the *sequence* $x_1, x_2, ..., x_L$ is generated is given by

$$P(x_1, x_2,..., x_L, q_0, q_1,..., q_{N+1} \mid \text{model}) = T(m_{N+1} \mid q_N) \prod_{i=1}^{N} T(q_i \mid q_{i-1}) P(x_{l(i)} \mid q_i)$$

where $P(x_{l(i)} \mid q_i) = 1$ if q_i is a delete state.

The probability of any sequence $x_1, x_2, ..., x_L$ is a sum over all possible paths that could produce that sequence. So we get

$$P(x_1, x_2,...., x_L \mid \text{model}) = \sum_{paths\ q_0, q_1,..., q_{N+1}} P(x_1, x_2,..., x_L, q_0, q_1,...., q_{N+1} \mid \text{model}).$$

The second equation defines a probability distribution on the space of amino acid sequences. The goal of our analysis is to find a model that describes a family of proteins by assigning large probabilities to amino acid sequences in this family. Liu et al. (1999) write the basic form of an HMM as

$$y_t \sim f_t(y \mid h_t) \text{ and } h_t \sim g_t(h \mid h_{t-1}).$$

Here f_t and g_t are probability distributions, the y_t are observations and the h_t form an unobservable Markov-chain. The dynamic linear model or state space model in time series analysis used by Schmitz and Urfer (1997) is a special case of this model.

There are several algorithms that given an arbitrary starting point for the parameters find a local maximum in such a way that the likelihood increases in each iteration. The well known EM-algorithm can be used to estimate transition probabilities and the amino acid distributions. This algorithm is often used in statistics in quite different applications such as toxicology (Selinski et al. (2000), Gilberg et al. (1999)) and plant genetics (Emrich and Urfer (1999)).

Aknowledgements

We would like to thank Prof. Dr. J. H. Peter and Dr. T. Penzel, Medizinische Poliklinik, Philipps-University of Marburg for providing the sleep apnea time series.

This research was supported by the German Research Council (DFG) through the Graduate College and the Collaborative Research Center at the University of Dortmund (SFB 475): Reduction of complexity for multivariate data structures.

References

Andrade, M.A., Casari, G., Sander, C., Valencia, A. (1997): *Classification of protein families and detection of the determinant residues with an improved self-organizing map*, Biological Cybernetics, 76, 441-450.

Behme, H., Brandt, W.D., Strube, H.W. (1993): *Speech Recognition by Hierarchical Segment Classification*, in: S. Gielen, B. Kappen (Eds.): Proc. Intl. Conf. on Aritificial Neural Networks (ICANN 93), Amsterdam, Springer Verlag, London, 416-419.

Bishop, C.M.(1995): *Neural Networks for Pattern Recognition*, Oxford, Clarendon Press.

Brunnert, M., Müller, O. and Urfer, W. (2000): *Genetical and statistical aspects of polymerase chain reactions*, Technical Report 6/2000, University of Dortmund.

Emrich, K. and Urfer, W. (1999): *Estimation of genetic parameters using molecular markers and EM-algorithm*, Technical Report 48/1999, University of Dortmund.

Gilberg, F., Edler, L. and Urfer, W. (1999): *Heteroscedastic Nonlinear Regression Models with Random Effects and Their Application to Enzyme Kinetic Data*, Biometrical Journal, 41, 543-557.

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S. (1999): *Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring*, Science, Vol. 286, October, 531-537.

Guimarães, G. (2000): *Temporal Knowledge Discovery for Multivariate Time Series with Enhanced Self-Organizing Maps*, To appear in: IEEE-INNS-ENNS Intl. Joint Conf. on Neural Networks (IJCNN'2000), Como, 24-27 July, Italy.

Guimarães, G. (1999): *Temporal Knowledge Conversion - The Extraction of Temporal Knowledge from Multivariate Time Series*, in: Procs of the 2nd Intl. Workshop for the Extraction of Knowledge from Databases (EKDB), associated with Intl. Conf. EPIA99, 65-79.

Guimarães, G. (1998): *Eine Methode zur Entdeckung von komplexen Mustern in Zeitreihen mit Neuronalen Netzen and deren Überführung in eine symbolische Wissenrepräsentation*, PhD Dissertation, University of Marburg, Germany.

Guimarães, G., Ultsch, A. (1999): *A Method for Temporal Knowledge Conversion*, Procs. of IDA99, The Third Symposium on Intelligent Data Analysis, August 9-11, Amsterdam, Netherlands, Lecture Notes in Computer Science, Springer Verlag, 369-380.

Joutsiniemi, S.L., Kaski, S., Larsen, T.A., (1995): *Self-Organizing Map in Recognition of Topographic Patterns of EEG Spectra*, IEEE Transactions on Biomedical Engineering, Vol. 42, No. 11, 1062-1068.

Kaski, S., Kohonen, T., (1996): *Exploratory Data Analysis by Self-Organizing Map: Structures of Welfare and Poverty in the World*, in: A.P.N Refenes, Y. Abu-Mostafa, J. Moody, A. Weigend (Eds.): Neural Networks in Financial Engineering. Proc. of the Intl. Conf. on Neural Networks in the Capital Markets, London, England, 11-13 October, 1995, Singapore, 498 – 507.

Kemke, C.,Wichert, A., (1993): *Hierarchical Self-Organizing Feature Maps for Speech Recognition*, Proc. of the World Congress on Neural Networks (WCNN 93), Hillsdale, Vol. III, 45-47.

Koh, J., Suk, M., Bhandarkar, S.M., (1995*): A Multilayer Self-Organizing Feature Map for Range Image Segmentation*, Neural Networks, Vol.8, No. 1, Elsevier Science Publisher, 67-86.

Kohonen, T. (1995): *Self-Organizing Maps*, Springer, New York.

Kohonen, T. (1982): *Self-organized formation of topologically correct feature maps*, Biological Cybernetics 43, 141-152.

Krogh, A., Brown, M., Mian, I.S., Sjölander, K. and Haussler, D. (1994): *Hidden Markov Models in Computational Biology - Applications to Protein Modeling*, Journal of Molecular Biology, 235, 1501-1531.

Liu, S. L., Neuwald, A.F. and Lawrence, C.E. (1999): *Markovian Structures in Biological Sequence Alignments*, Journal of the American Statistical Association, 94, 1-15.

Mujunen, R.,Leinonen, L, Kangas, J., Torkkola, K., (1993): *Acoustic Pattern Recognition of /s/ Misarticulation by the Self-Organizing Map*, Folia Phoniatr., 45, 135 - 144.

Penzel, T., Peter, J.H.: *Design of an Ambulatory Sleep Apnea Recorder*, in: H.T. Nagle, W.J. Tompkins (Eds.): Case Studies in Medical Instrument Design, IEEE, New York, 1992, 171-179.

Peter, J.H., Becker, H., Brandenburg, U., Cassel, W., Conradt, R., Hochban, W., Knaack, L., Mayer, G., Penzel, T. (1998): *Investigation and diagnosis of sleep apnoea syndrome*, in: McNicholas, W.T. (ed.): Respiratory Disorders during Sleep. European Respiratory Society Journals, Sheffield, 106-143.

Schmitz, N. and Urfer, W. (1997): *State-dependent time series models for heart rate dynamics data and their application to psychophysiology*, Informatik, Biometrie und Epidemiologie in Medizin und Biologie, 28, 169-184.

Selinski, S., Golka, K., Bolt, H.M. and Urfer, W. (2000): *Estimation of toxicokinetic parameters in population models for inhalation studies with ethylene*, (Environmetrics, accepted).

Ultsch, A., Kleine, T.O, Korus, D., Farsch, S., Guimarães, G., Pietzuch, W., Simon, J. (1997), *Evaluation of Automatic and Manual Knowledge Acquisition for Cerebrospinal Fluid (CSF)*, in: E. Keravnu et al. (Eds.): Artificial Intelligence in Medicine, Lecture Notes in Artificial Intelligence 1211, Vol. 934, Springer Verlag, 110-121.

Ultsch, A., Siemon, H.P. (1990): *Kohonen´s Self-Organizing Neural Networks for Exploratory Data Analysis*, Proc. Intl. Neural Network Conf. INNC90, Paris, Kluwer Academic, 305-308.

Walter, J.A., Schulten, K.J. (1993): *Implementation of Self-Organizing Neural Networks for Visual-Motor Control of an Industrial Robot*, IEEE Transactions on Neural Networks, Vol. 4, No.1, January 86-95.