Automatic Neonatal Sleep EEG Recognition with Social Impact Based Feature Selection

Martin Macaš, Václav Gerla, and Lenka Lhotská

Czech Technical University, Technicka 2, 166 27 Prague 6, Czech Republic
macas.martin@fel.cvut.cz

Abstract. The paper presents an application of Simplified Social Impact Theory based Optimization on feature subset selection for automated neonatal sleep EEG recognition. The target classifier is 3-Nearest Neighbor classifier. We also propose a novel initialization of iterative population based optimization heuristics, which is suitable for feature subset selection, because it reduces the computational complexity of whole feature selection process and can help to prevent overfitting problems. Our methods leads to a significant reduction of the original dimensionality while simultaneously reduce the classification error.

1 Introduction

During the last century, many of natural social phenomena were modeled by ethologists, social psychologist, economists and others. Examples are agent-based models of ant behavior, models of swarming, or models of opinion formation. In last two decades, these models of natural optimization processes are modified and "forced" to solve mathematical optimization problems. Thus, methods like Ant Colony Optimization or Particle Swarm Optimization (PSO) [1] are being invented and still more and more intensively applied to real-world problems. This paper presents an application of Simplified Social Impact Theory based Optimization (SSITO) [2] inspired by opinion formation models on Neonatal Sleep EEG Recognition.

In this study we focus primarily on differentiating between two important neonatal sleep stages: quite sleep and active sleep. In clinical practice, the proportion of these states is a significant indicator for the maturity of the newborn brain [3]. Manual evaluation of EEG is a very tedious operation, and an electroencephalographer can easily make a mistake. Therefore, the classification process is being automatized in terms of feature based pattern classification. In most cases of automatic neonatal EEG classification, large amounts of EEG data must be processed. It is also complicated by the fact that various additional channels must also be processed. It is therefore necessary to compress the calculated features using a sophisticated technique. In this paper, we deal with the reducing of number of appropriate features used for the automatic classification of neonatal EEG in terms of selection of a proper subset of features.

2 Neonatal Sleep EEG Recognition

The data used in this study was provided by the Institute for Care of Mother and Child in Prague. We have 11 full-term healthy newborn records (37 - 40 weeks gestation;

V. Snasel et al. (Eds.): SOCO Models in Industrial & Environmental Appl., AISC 188, pp. 389–398.
springerlink.com

5 minutes of quiet sleep and 5 minutes of active sleep for each record; no artifacts; clearly defined sleep states). All data was recorded from eight referential derivations, positioned under the 10-20 system, namely FP1, FP2, T3, T4, C3, C4, O1, and O2. The sampling frequency was 128 Hz. The reference derivation (R) used linked ear electrodes. In addition, the following polysomnographic signals were used: EOG, EMG, ECG, and PNG. All channels were measured against ground. The EOG signal was recorded from two electrodes placed slightly above and to the outside of the right eye and below and to the outside of the left eye. Two EMG electrodes were placed on the chin and at the left corner of the mouth. ECG was recorded using two electrodes, one placed over the sternum and the other in the medial axillary line. The respiratory effort was measured using a tensometer placed on the abdomen. Examples of signals in quiet and active sleep are shown in Figure 1. To eliminate power-line noise, we used a notch filter. This rejects a narrow frequency band and leaves the rest of the spectrum almost undistorted.

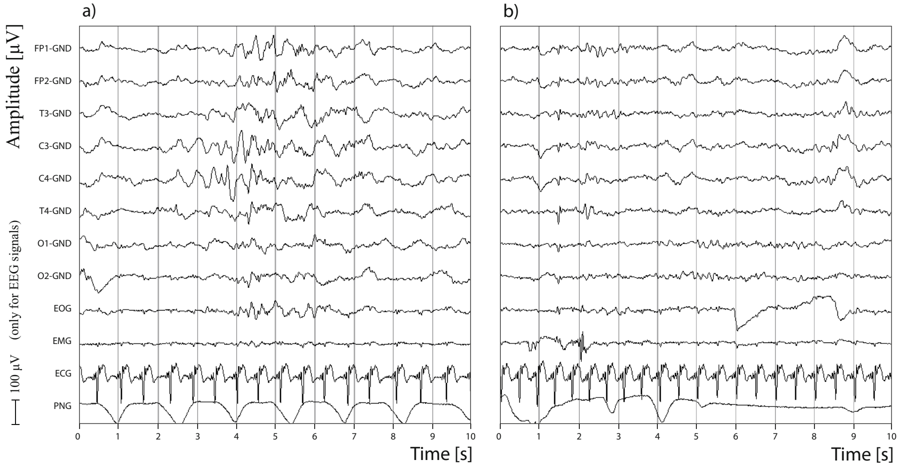

Fig. 1. Example of neonatal PSG recording; a) quiet sleep, b) active sleep. Signals: FP1, FP2, C3, C4, O1, O2, T3, T4, EOG, EMG, ECG, and PNG.

For the subsequent processing methods it is necessary to divide the signals into almost stationary segments. In this study we use constant segmentation into 1 second segments. All features listed below were calculated from these segments. Some of these (auto-correlation, cross-correlation and coherence features) were calculated from sliding window (length $20s$, shift $1s$) and were then used in such a way as to match the $1s$ segments. In this way it was obtained a total of 2087 features that were used for neonatal data classification. The following list summarizes the used feature extraction techniques and give their brief description.

- *Statistical description.* EEG signal can be characterized by the distribution of amplitude and its moments [4].
- *Interval and period analysis.* The intervals between zero and other level crossings, or intervals between maxima and minima were measured and moments of their distribution were used as features [5].

- *Application of derivatives.* Statistical features are extracted also for the first and the second derivative of EEG signals.
- *Hjorth parameters.* The Hjorth parameters are simple measures of signal complexity. These measures have been used in the analysis EEG and they are clinically useful tools for the quantitative description of an EEG [4].
- *Power spectral analysis.* We compute the mean value of absolute and relative power spectra over the common frequency bands (delta, theta, alpha, beta) [4].
- *Entropy-based features.* Entropy is thought to be a measure of EEG signal complexity and so it is potentially useful feature for our purposes [4].
- *Nonlinear energy operator.* Another features were based on the nonlinear energy operator [4].
- *Auto-correlation and cross-correlation.* Cross-correlation is a measure of similarity of two signals and auto-correlation is the cross-correlation of a signal with itself [6]. We compute the maximum positive amplitude and mean value from auto- and cross-correlation function (for selected polygraphic signals).
- *Coherence analysis.* The inter- and intra-hemispheric coherence are also calculated from the EEG signal [6].

In addition, we have also used the information extracted from the other polysomnographic channels (heart rate variability from ECG signal, regularity of respiration from PNG signal, presence of eye movements from EOG signal and body movements from EMG signal, see [7] for more details).

To make the number of input features suitable for wrapper methods, we first preselect the features by evaluating features individually using inter-intra class distance filter criterion [8] and taking only 500 best features. Thus, there is 4400 data instances of dimension 500 obtained from 11 subjects (400 instances from each).

3 Feature Selection

A feature selection process usually consists of two main components - a evaluation criterion, which evaluates potential feature subsets, and a search method, which seeks for a minimum of the criterion. Here, we use the wrapper approach to feature selection, i.e. use performance of the target classifier as evaluation criterion. The criterion is minimized using the SSITO method. Both components are described in the next sections.

The classification is performed using 3-Nearest Neighbor classifier (3NN). It simply finds 3 training data instances that are most similar to the testing instance and assigns the instance into the most common class amongst the 3 nearest neighbors. We use the Euclidean distance for similarity quantification, because it was observed to lead to good classification accuracies while keeping reasonable computational requirements. The nearest neighbor classifiers are still widely used in pattern classification, because of its simplicity, high performance (especially, but not only in large sample limit, and robustness to noisy learning data [9]. Many more sophisticated classifiers need much more time for training and testing and the wrapper approach is not suitable for them because of computational complexity reasons. Moreover, in our preliminary experiments with full feature set, 3NN outperformed quadratic Bayes classifier (assuming normally distributed classes with different covariance matrices) and CART decision tree.

Our particular feature selection criterion is the 2-fold cross-validation estimate of 3NN's error, because it was observed in some preliminary experiments with different datasets [10] to perform better than 10-fold setting or leave-one-out technique. This corresponds to some other studies, e.g. [11]. In 2-fold cross-validation, the data set is partitioned into 2 disjunctive folds of similar size. For each of two iterations, one fold is used for testing and remaining fold is used for training. The 2-fold cross-validation error estimate is the testing error averaged over the two folds.

Let D be the number of features and $\mathscr{A} \subseteq \{1,\dots,D\}$ be a subset of feature indices, which represents a feature subset. The optimization techniques assume the following encoding of a feature subset $\mathscr{A}$: $\mathbf{s} = \{0,1\}^D$, where ath component $s^a = 1$ means that the feature with index a is selected (i.e. $a \in \mathscr{A}$) and $s^a = 0$ means that the feature with index a is removed (i.e. $a \notin \mathscr{A}$).

Thus, the feature selection is defined here as a minimization of the cost function $f(\mathbf{s})$ defined as 2-fold cross-validation error estimate of 3-Nearest Neighbor classifier trained with features represented by $\mathbf{s}$. The optimization method used for the minimization is described below.

The approach described here tries to take a model from social psychology, adapt it, and use it in the area of parameter optimization. It is an attempt to use simulated people to make a decisions about solutions of an optimization problem. The simulation is based on simple opinion formation models widely used in computational psychology and commonly analyzed by tools of statistical physics. We present application of relatively novel population-based optimization methods, in which the candidate solutions influence each other and try to converge into a "good" consensus. The method called Simplified Social Impact Theory based Optimizer (SSITO) is applied here to the feature subset selection problem known from pattern recognition.

Many opinion formation models combine the social information using the notion of social impact function that numerically characterizes the total influence of social neighborhood of a particular individual. We use the analogy with the Nowak-Szamrej-Latané models [12]. Let $\{\mathbf{s}_1(t),\dots,\mathbf{s}_L(t)\}$ be a set of L candidate solutions of the feature selection problem at iteration t. Here, the population size L is 25 individuals. Each candidate solution is influenced by its social neighborhood. Here, the neighborhood simply consists of 5 randomly selected individuals. The neighbors with higher strength value have higher influence on the impact value. The strength can be associated with pair of individuals. One possible choice is – the social strength q_{ji} by which a candidate solution j affects candidate solution i depends on their cost values according to the following formula:

$$q_{ji}(t) = \max[f(\mathbf{s}_i(t)) - f(\mathbf{s}_j(t)), 0], \tag{1}$$

where $f(\mathbf{s}_i(t))$ and $f(\mathbf{s}_j(t))$ are the cost values of the candidate solutions i and j, respectively. This equation means that fitter individual have a non-zero influence on less fitter individual and is not influenced by it. Obviously, there is an infinite number of possible cost–strength mappings. Some of them can lead to much better optimization abilities.

Considering component a of candidate solution $\mathbf{s}_i(t)$, the impact function depends on the component a of candidate solutions from i's neighborhood and on strength values of these solutions. It characterizes the total impact on individual i. A positive impact value leads to preference of ath component inversion. Contrary, the negative value have supportive character and leads to preference of keeping the component value. A particular form of the impact function will follow the opinion formation models described in [12]. At each iteration and for each component a, the neighbors of i are divided into two disjoint subsets, persuaders $\mathscr{P}_i^a(t)$ with opinion opposite to $s_i^a(t)$ and supporters $\mathscr{S}_i^a(t)$ with the same value of the opinion. The impact is defined as:

$$I_i^a(t) = \frac{1}{|\mathscr{P}_i^a(t)|} \sum_{j \in \mathscr{P}_i^a(t)} q_{ji}(t) - \frac{1}{|\mathscr{S}_i^a(t)|} \sum_{j \in \mathscr{S}_i^a(t)} q_{ji}(t). \tag{2}$$

Moreover, we define $I_{Si}(t) = 0$ if $\mathscr{S}_i^a(t) = \emptyset$ and $I_{Pi}(t) = 0$ if $\mathscr{P}_i^a(t) = \emptyset$.

The update rule further uses the value of impact function to generate new state for s_i^a. The simplest deterministic update rule uses the analogy to [12] - individual changes its opinion if the impact function takes a positive value:

$$s_i^a(t+1) = \begin{cases} 1 - s_i^a(t), & \text{if } I_i^a > 0; \\ s_i^a(t), & \text{otherwise.} \end{cases} \tag{3}$$

The algorithm described above ignores the aspect of individual decision processes (e.g. experience, memory, inferring mechanisms) and of many unknown processes. These can be partly modeled by a random noise. Moreover, randomness is an essential part of any optimization metaheuristic. Hence the random noise is added in our optimizers. The simplest way to add the random element is to mutate all s_i^a with probability of spontaneous opinion inversion (mutation rate) $\kappa << 1$. This can keep the diversity and avoid a premature convergence.

The pseudocode is in Algorithm 1. First, the initial population $\{\mathbf{s}_i(0)\}_{i=1\ldots L}$ is created randomly and all cost and strength values are computed. At each iteration, vector $\mathbf{s}_i(t)$ is transformed into a new vector $\mathbf{s}_i(t+1)$ using an update rule. It updates the a-th bit of the vector $\mathbf{s}_i$ according to its value, the values of a-th bit of vectors positioned in i's neighborhood and according to their strength values. After the update of all $\mathbf{s}_i$ vectors, new values of cost and strength can be computed and the next iteration is performed.

Algorithm 1. Pseudocode for SSITOmean algorithm

```
initialize all $\mathbf{s}_i(0)$
while stop condition not met do
    for all $i$ do
        evaluate $\mathbf{s}_i(t)$ by computing $f(\mathbf{s}_i(t))$
    end for
    for all $i,j$ do
        compute strength values $q_{ji}(t)$ using equation 1
    end for
    for all $i,a$ do
        compute $I_i^a(t)$ using equation 2
    end for
    for all $i,a$ do
        compute $s_i^a(t+1)$ using equation 3 and random mutation
    end for
    t:=t+1
end while
```

The whole iterative process can lead to a search strategy that samples the binary space using the set of candidate solutions $\mathscr{Q}$. Here, the stopping condition is the reach of a maximum number of cost evaluations or maximum number of iterations, but it can be any other criterion known from the area of population based metaheuristics (cost increase, diversity degradation, etc.)

4 Results

The main purpose of our feature selection technique is to find a feature subset, that minimizes classification error of the classifier. To estimate the true benefits of the feature selection, we use an outer-loop estimate for testing. We divided the dataset into 11 folds according to the membership of instances to particular subjects (neonates). Further, we used the cross-validation algorithm on these folds. This special type of cross-validation is a fair approach that reduces a positive bias caused by ignoring of an inter-personal variability.

To show a competitiveness of our approach, we perform the same feature selection using Binary Particle Swarm Optimization algorithm (BPSO) [1], which is also based on social-psychological metaphor and is often applied to feature selection. We used the following settings: inertia weight $\omega = 1$, weights of individual and social knowledge $\varphi_1 = \varphi_2 = 2$, maximum velocity $\mathbf{v}_{max} = 5$, and the ring topology with neighborhood radius 2.

Moreover, we implemented two modifications. The first is a so-called reduced initialization and the second is the modification of the criterion called combined criterion. In the original SSITO algorithm (and in most population based methods), the initial population is created randomly, i.e. each bit is set to 1 with probability 0.5. In average, each candidate solution corresponds to $D/2$ selected features. This can be a disadvantage if the optimal solution contains a small number of features. Furthermore, it makes the search more computationally complex. Therefore, we reduce the number of features to a minimum. The main requirement is that there must be exactly one occurrence of each feature in the initial population (exactly one candidate solution contains the feature). Thus, in average each candidate solution corresponds to D/L selected features. This can significantly reduce the temporal complexity of cost evaluation (error estimation). The reduced initialization thus means that the candidate solutions are initialized randomly under the condition described above.

The second modification is referred here as the *combined* criterion. It is not novel, many papers use this approach. It simply combines the error estimate with the number of features, which leads to a more intensive dimensionality reduction. We use a simple linear combination. Let $e(\mathbf{s}(t))$ be the error estimate. Instead of using $f(\mathbf{s}(t)) = e(\mathbf{s}(t))$, the combined criterion uses $f(\mathbf{s}(t)) = e(\mathbf{s}(t)) + \alpha d/D$, where α parameter weights the relative importance of the dimensionality reduction and its optimal value depends on the depends on the classifier and on the data set. In our experiment we show the behavior of the selection for $\alpha = 1$.

A comparison for our approaches is depicted on Fig. 2, 3 and 4, where the temporal evolution of three variables averaged over 11 cross-validation runs is depicted. The variables are measured for the best-so-far solution corresponding to the minimum cost value

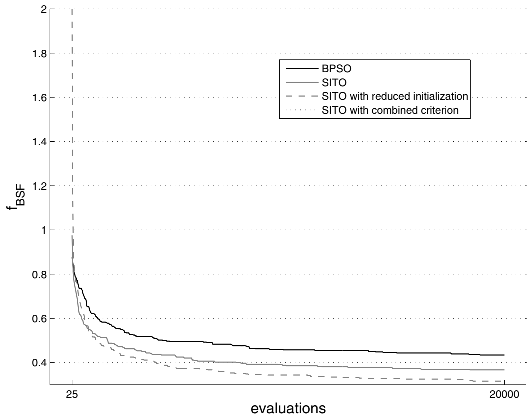

Fig. 2. The criterion value of the best-so-far solution. The combined criterion is not included in the graph, because its values is not comparable to the values of other criteria.

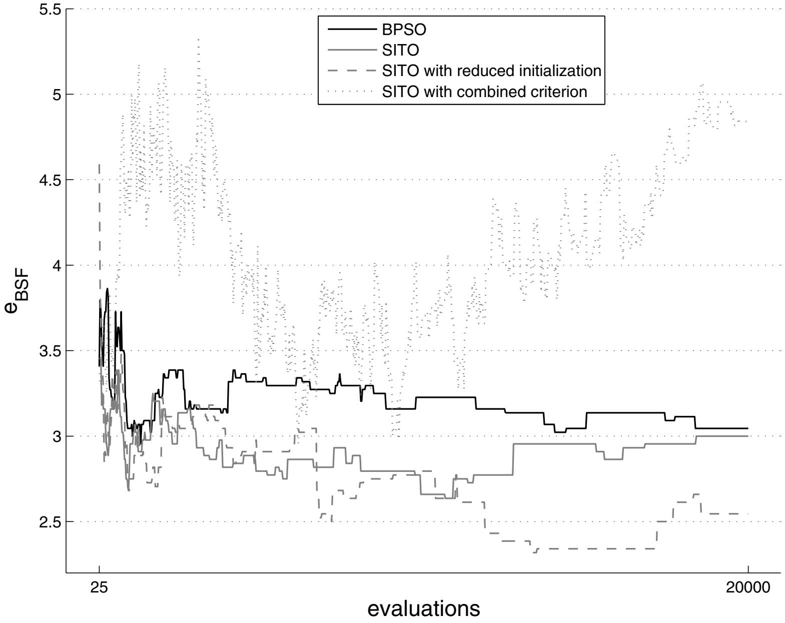

Fig. 3. The testing error value of the best-so-far solution (in %)

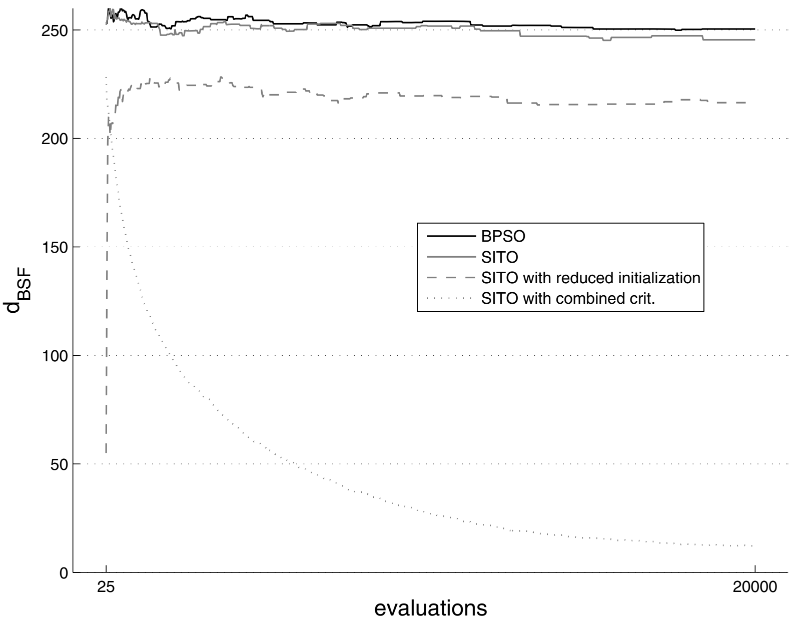

Fig. 4. The number of features in the subset represented by the best-so-far solution

found so far. The average best-so-far cost value is depicted in Fig. 2. The curve for the combined criterion is omitted, because it takes different and incomparable values. One can observe that the SSITO algorithm with normal random initialization outperforms the BPSO method. The reduced initialization, although leads to much worse initial cost values, enables the algorithm to find fitter feature subsets after a small number of iterations. However, the f_{BSF} measure evaluates only the optimization capabilities of the algorithm that are not so important in our application. A much more important measure is the testing error of the best-so-far solution, which is usually higher than the estimated error value used to guide the search mechanism.

This phenomenon can be seen in Fig. 3. The difference between final cost values and the final testing error values is approximately 2.5%. The testing error for full set of 500 features was 3.4%. It is transparent that all methods that reduce the error estimate also lead to a reduction of the testing error. On the other hand, the use of combined criterion with our setting of α does not lead to a reduction of the testing error. The benefit of the combined criterion is shown in Fig. 4, where the output number of features d_{BSF} in the best-so-far solution is depicted. One can see that all the three instances of SSITO method perform differently from the dimensionality reduction point of view. Although the normal SSITO with normal initialization leads to only slightly smaller number of features than the BPSO method and eliminates only 50% of features, the situation is different for SSITO with reduced initialization and SSITO with combined criterion. The reduced initialization leads to much smaller dimensionality of the best-so-far candidate solution (and also of all candidate solutions) and finally converges into a solution, which corresponds to less features, small cost value and the best testing error. Thus, the SSITO with reduced initialization is the best from the error minimization point of view. However, it is a frequent case, that physicians need to somehow interpret the results and compare them with their current medical knowledge. For such a case, there is a strong need for obtaining a very small number of features. For such a case, the combined criterion can be very practical. As one can see in Fig. 4, the combined criterion reduced the dimensionality from 500 features to 12 features, while it kept the testing error in reasonable bounds. Obviously, the requirement for setting of α corresponds to a need of some preliminary experiments, which can be a disadvantage of the approach.

5 Conclusions

In the paper, we describe an application of relatively novel soft computing method to a biomedical signal processing problem. Particularly we show, how the socially inspired SSITO algorithm can be useful in wrapper–based feature subset selection. All presented approaches significantly reduce the dimensionality and compared to the full set of 500 features, all methods also lead to a reduction of the testing error of 3NN classifier in about $0-1\%$. The presented SSITO method is very simple and outperforms the commonly used BPSO algorithm. We also propose a novel reduce initialization of the SSITO methods which can be directly applied to any optimization metaheuristic. The reduced initialization leads to a significant reduction of computational requirements and helps to reduce problems related to overfitting and feature selection bias. For the case of a need of a good interpretability of results, very small subset of features can be selected using the combined criterion, which also considers the relative dimensionality of the candidate feature subset.

References

1. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: IEEE International Conference on Systems, Man, and Cybernetics, vol. 5, pp. 4104–4108 (1997)
2. Macaš, M., Lhotská, L.: Simplified Social Impact Theory Based Optimizer in Feature Subset Selection. In: NICSO, pp. 133–147 (2011)
3. Lofhede, J., Degerman, J., Lofgren, N., Thordstein, M., Flisberg, A., Kjellmer, I., Lindecrantz, K.: Comparing a supervised and an unsupervised classification method for burst detection in neonatal EEG. In: Conf. Proc. IEEE Eng. Med. Biol. Soc., pp. 3836–3839 (2008)
4. Greene, B.R., Faul, S., Marnane, W.P., Lightbody, G., Korotchikova, I., Boylan, G.B.: A comparison of quantitative eeg features for neonatal seizure detection. Clin. Neurophysiol. 119, 1248–1261
5. Niedermeyer, E., da Silva, F.H.L.: Electroencephalography, basic principles, clinical applications, and related fields. In: Urban & Schwarzenberg (1982)
6. Tong, S., Thakor, N.V.: Quantitative EEG analysis methods and clinical applications. In: Engineering in Medicine & Biology. Artech House (2009)
7. Gerla, V., Lhotská, L., Krajča, V., Paul, K.: Multichannel analysis of the newborn EEG data. In: International Special Topics Conference on Information Technology in Biomedicine, Piscataway. IEEE (2006)

8. van der Heijden, F., Duin, R., de Ridder, D., Tax, D.M.J.: Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB. John Wiley and Sons (2004)
9. Bhatia, N.: Survey of nearest neighbor techniques. Journal of Computer Science 8(2) (2010)
10. Macaš, M., Lhotská, L., Bakstein, E., Novák, D., Wild, J., Sieger, T., Vostatek, P., Jech, R.: Wrapper feature selection for small sample size data driven by complete error estimates. Computer Methods and Programs in Biomedicine (to appear, 2012)
11. Weiss, S.M.: Small sample error rate estimation for k-NN classifiers. IEEE Trans. Pattern Anal. Mach. Intell. 13, 285–289 (1991)
12. Nowak, A., Lewenstein, M.: Modeling Social Change with Cellular Automata. In: Modeling and Simulation in the Social Sciences from the Philosophy of Science Point of View, pp. 249–285. Kluwer Academic (1996)