Sleep Stage Classification using Fuzzy Sets an Machine Learning Tecniques

Pedro Piñero[1], Pavel Garcia[1], Leticia Arco[2] , Alfredo Álvarez[3], M. Matilde García[2], Rolando Bonal[2]

[1] Universidad de las Ciencias Informáticas, Ciudad de La Habana, Cuba
{ppp, cortina}@uci.cu

[2] Centro de Estudios Informáticos, Universidad Central "Marta Abreu" de Las Villas, 54830 Villa Clara, Cuba
{leticiaa, mmgarcia, bonal}@uclv.edu.cu

[3] Clínica del sueño, Centro de Neurociencias de Cuba, 10600 Ciudad de La Habana, Cuba
alfredo@cneuro.edu.cu

Abstract. The sleep stages automatic classification is one of the major problems in neuroscience. The hypnogram (visual representation of the distribution of sleep states in patients) is determinated after a study of electrophysiologic record. In this paper we present the ISSSC system (Intelligence Systems for the Sleep Stages Classification). This system is divided into tour different modules: the first processes the electrophysiological signals and determines this signals most relevant parameters; the second module establishes (using those parameters) fuzzy rules that will use during the classification process; the third module is an inference module, it implements a zero degree Sugeno model and the last one adjust the automatic classification process. Finally the system builds the patient's hypnogram and provides us different outputs. We present the classification results obtained from applied the systems to classify patients with different sleep disorders.

Keywords: sleep stages, fuzzy rules base reasoning, fuzzy system, expert system, machine learning

1 Introduction

The sleep stages automatic classification has become a key problem in neuroinformatics and generally speaking in neuroscience. That is, to describe the patient's transition through the sleep stages (hypnogram) taking into account the study of the electrophysiologic records such as Electroencephalogram (EEG), Electrooculogram (EOG) and Electromyography (EMG). All these records together are known as Polysomnogram.

The sleep stages automatic monitoring is an important source of information that enhances the study of different diseases. The most popular technique to process the sleep stages based on rules were proposed by Rechtschaffen and Kales (R&K) in 1968 [18]. Today it is widely recognized that these rules have severe limitations which were not foreseen 30 years ago [9][12]. Nevertheless, these rules have survived all criticism raised in the past. The unbeaten advantage of these rules is that they extended the Loomis's observations[1] [13]. Other reason for the R&K continuous use is

[1] Loomis describes the sleep stages from A to E.

that these rules can be learned with minimum effort and provide a guidelines with clear directives for most situations. The limitations of these rules are discussed extensively by Himanen and Hasan [9].

In this work we will assume that the sleep stage scoring is a kind of classification problem. In this problem the human sleep stages can be classified into one of 6 discrete stages according to R&K rules. In general, the sleep stages classification process follows two different states, like in any other classification problem: first it starts from a set of observations to obtain their classes or clusters (learning process), and afterwards, it continues from the known classes to classify new observations using an inference method. According to this principle, three families of classification problems appear:

- Classification with learning (Supervised classifications).
- Classification with partial learning (Semi-supervised classification).
- Classification without learning (Unsupervised classification).

In this paper we discuss about the first family of classification problems (supervised problems) that includes the problems whose classes and objects are known during the learning process. To solve our classification problem we use of Machine Learning techniques and the fuzzy logic theory.

Machine Learning is an interdisciplinary field with connections to artificial intelligence, information theory, statistics, pattern recognition and cognitive science. Machine learning algorithms learn automatically from experience and use different forms to represent knowledge. Among the many ways to represent knowledge, if-then rules [21][5][2] have been used with other techniques to make inferences and classify some new cases. Their common use is due to their simplicity and objectiveness.

On the other hand, the fuzzy rules, an example of if-then rules, are based on the use of fuzzy logic to represent knowledge. A lot of problems need the learning of fuzzy rules, because the kinds of features need it. Fuzzy rules are essential components in the inference fuzzy systems, an earlier kind of classifier that has won popularity in recent years. General fuzzy rules can be represented as: *If* x *is* A *then* y *is* B*;* where, *B* and *A* are fuzzy sets belonging to the *Y* and X linguistic variables respectively.

Fuzzy set [7] is usually identified by membership functions. A fuzzy set μ of X is a mapping involving set X up to the unit interval: $\mu: X \rightarrow [0, 1]$. Here $\mu(x)$ is known as the x-value membership degree.

We dealt with the Intelligence Systems for the Sleep Stages Classification (ISSSC). In section 2, details on the modules that constitute the system are given. And in section 3, the results of the use of this system are validated based on the analysis of the system implementation and the elements that affect the classification process. And finally we discuss about the conclusions of this paper.

2 Intelligent System for the Sleep Stages Classification (ISSSC)

The ISSSC version 1.0, consists on four different modules. The operation of this system modules correspond to two different system's case of use. On the one hand, the ISSSC is used as part of a learning process and to generate rules of fuzzy inference that could be apply in further classifications of new patients. On the other hand, the case of use is related to the classification and study of new patients taking into account the first case of use.

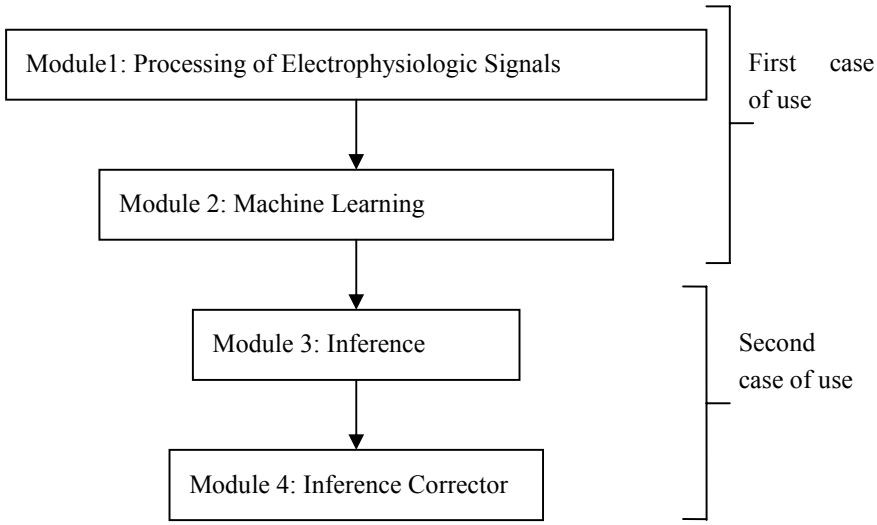


Fig. 1. Module diagram and cases of use

2.1 Module 1: Processing of Electrophysiologic Signals and computation of relevant parameters

This module processes the patient's electrophysiologic records that are stored in binary files. Then after processing those signals, the system computes the descriptors that allow us to identify the different sleep stages of given patient.

2.1.1 *Obtaining of the polysomnographic records*

We carried out this research in the Clinical Neurophysiology Offices of the Neurosciences Center of Cuba. We recorded the EEG, EOG and EMG using a Medicid -3E. In each case we work with silver-disk electrodes stuck with collodion to the skin or to the scalp. In the electrode-skin interface, we managed to achieve an impedance lower than 10 Kohms after a previous cleaning of the skin making use of alcohol and applying a conductive gel that facilities the transmition of the electric potentials.

The EEG derivations were: Fz, C3, C4, O1 and O2 of the 10/20 system. We established: frequency range from 0.5 to 30 Hz and the gain of 10600 in the amplifiers. We use two monopole channels for recording the EOG. One channel recorded the potential difference in an electrode located slightly to the side of the external angle of one eye and 1 cm above of it. The other channel recorded the potentials in an electrode located slightly to the side of the external angle of the other age and 1cm below.

We situated two electrodes under the patient's chin for recording the EMG. We established a 10-100 Hz and the gain was of 10000. The test lasted 5 ms in each case. We studied the signals taken form some of Neuroscience Center of Cuba patients. The records we had were not alike. The sample was 4 patients (MRLL, ACGBCLA, AGSCLA and ARR) suffering from different illnesses: Insomnia and depressive disorders, breathing difficulties when sleeping, Narcolepsy, etc. The manual sleep classification of the patient was made by different technicians, that are why, the results match only in a 71 %.

2.1.2 Computation of relevant descriptors

Here, the main purpose is to sum up data, trying to prevent missing important information; that is to reduce the number of variables without losing relevant information. The electrophysiologic signals divided into 30 sec periods.

This module is used by both cases of use already mentioned. In one case the module computes the necessary descriptors of patients that have been already studied, and forms a training set that could be use for learning and rules production. In the other case, the module computes the descriptors of new patients, so they could be classified (see fig 2)

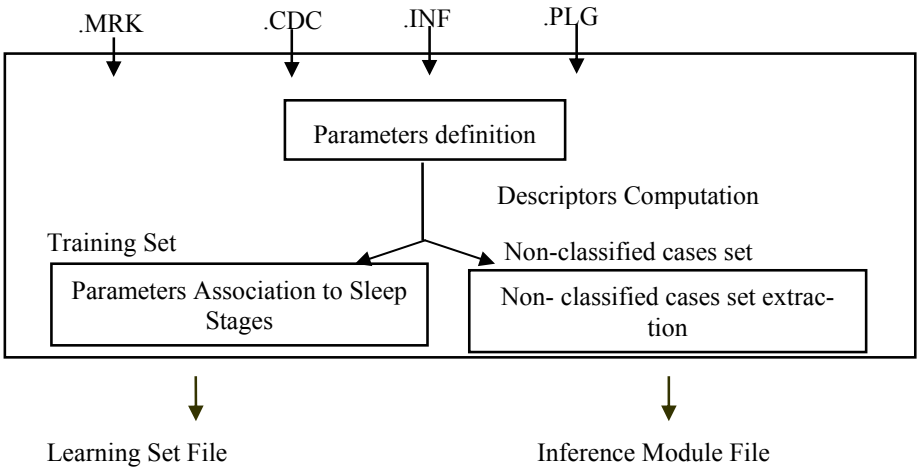


Fig. 2. Training set formation and non- classified cases set

The input files to this module are: MRK, CDC, INF and PLG. And their structure is as follows. The archives with the extension (.INF) are text archives that keep general information organized as follows:

<Context> = <N> <List>

<Context> character stream that identify a data.

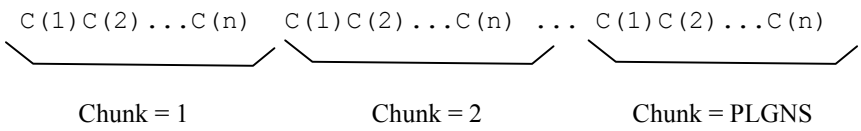
<N> integer number that indicate the amount of values associated to this data.

<List> it is the sequence of the same type of N values separated by spaces in blank. That is, there are only in a list characters and numbers.

We are only interested in the following data from all the information in a list:

`PLGNC = 1 7`	Number of channels
`PLGNS = 1 12000`	Number of simples that were registered at the same time in every channel (chunks).
`PLGMontage = 7 C3_-A12 LOG-A12 C4_-A12 ROG-A12 O1_-A12 EMG-25 O2_-A12`	Assembly derivations
`PLGSR(Hz) = 1 200.00`	Sampling frequency. The sampling period in milliseconds is calculated as follows: 1000/PLGSR.
`EpochTime = 1 30`	Time in second of the analysis epoch

The archives with extension (.PLG) keep the registered signals. These are archives of integer numbers (16 bits) with the following structure.



There are in the archive chunk sequences that have as many N sampled values as channels (n = PLGNC).

The archive with the extension (.MRK) store the sleep stages, marks and information inserted in the record once the specialists have analyzed it. These archives of characters ASCII and integer numbers (32 bits) with the following structure:

`Car(1)Pos(1) Car(2)Pos(2) ...`

Every Car(i) is the character that encodes the stage, mark, etc. Pos represents the record chunks where the character is inserted (1≤`Pos`≤PLGNS).

In this file, we were only interested in the information about the sleep stages and the record chunks where it is located.

Finally, the archives with the extension .CDC store the calibration values and the DC level of every register channel. It is an archive of floating-point values with the following structure:

`Cal(1)DC(1) Cal(2)DC(2) ... Cal(n)DC(n)`

Every Cal(i)DC(i)is the calibration value and DC level of the i channel (1≤`i`≤PLGNS).

This file is used to convert the calibration values to microvolt.

Once the input information is analyzed, it has to be processed in order to form the training set. In this step, every epoch is considered as a case and the sleep stage the patient is in, is the class that the case belongs to.

If the simple for the analysis is a patient already studied, the output in the first module will be the files (.lrn), (.data) and (.names). These files will be used in the learning process and fuzzy rules generation. If the simple is a new patient, then the output is a file (.cse).

The file (.lrn) stores a training set and there is always a file (.names) where the description of this set is stored. Now, the file (.data) combines all the information in the first two files to supply two different ways to organize the output information of this module.

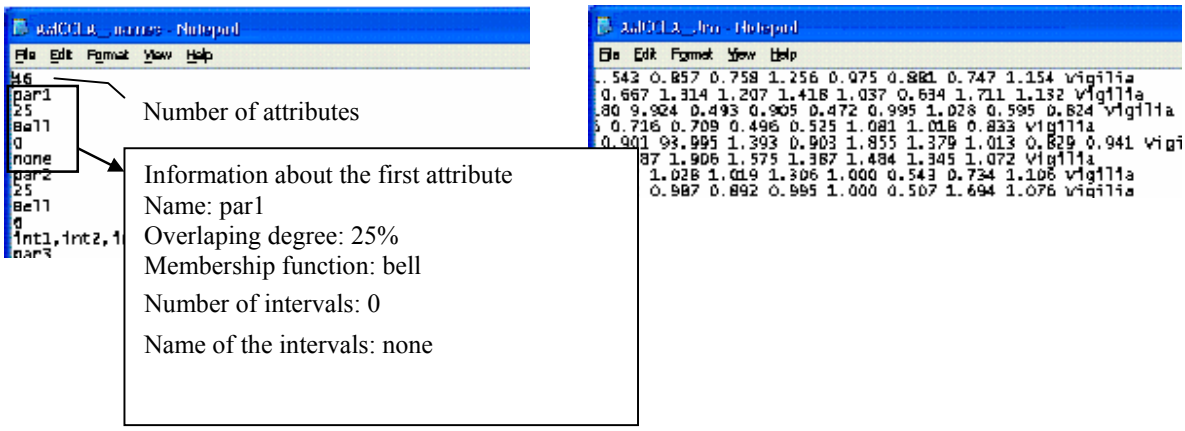


Fig 3. Structure of the files (*.names*) and (*.lrn*) of the first module. From the left to rigth .names y .lrn

2.1.1 Parameterization

We obtain the alpha, theta, delta and sigma waves taking into account the electroencephalogram signal. There are different methods to filter and process the signal, such as: wavelet transform [16], Fourier Transform [16] or using a Butterworth filter [1].

The parameterization was carried out using Matlab version 6.0 due to the advantages of this software when processing signals.

The patients sleep records have electroencephalographic signals from the following channels: (C3-A12, C4-A12, O1-A12, O2-A12), EOG signals from both eyes and EMG signals.

The sampling frequency of the signal was 250 Hz. The EEG was previously filtered with a down-step filter with a frequency of 30 Hz.

The parameters we got coincide with Butterworth filter [1]

Results of the EEG:

- There was an increment of the brain activity relative to the average
- There was a middle level of the alpha activity (8-11 Hz) in the stage relative to the EEG level
- There was a middle level of the theta activity (3.5-3.8 Hz) in the stage relative to the EEG level
- There was a middle level of the delta activity (0.5-3.5 Hz) in the stage relative to the EEG level
- There was a middle level of the sigma activity (11.5-15 Hz) related to the EEG level
- Maximum Absolute Amplitude of the EEG

Results of the EMG:

- There was an increment of the muscular activity related to the average.
- Maximum Absolute Amplitude of the EMG

Results of the EOG:

- There was an increment of the ocular activity related to the average.
- Maximum Absolute Amplitude of the EOG
- Product of alpha level and the ocular activity level.

2.2 Module 2: Machine learning

Two algorithms that generate fuzzy rules ware created for this module. They use a training set with nominal and numeric data. There is a general description of this module in figure 4.

This module is used in the first case of use of the system. The input files are (.lrn) and (.names) or (.data) files that were generated in the previous module. Two different types of processing the parameters are identified according to the information in the input files.

Method 1 *Discretization of parameters*: a specialist provides linguistic terms that correspond to the parameter. The system identifies the data and determines the intervals for each linguistic term.

Method 2 Clusterization of parameters: the linguistic terms that correspond to the parameter, are determined automatically in the Clusterization process.

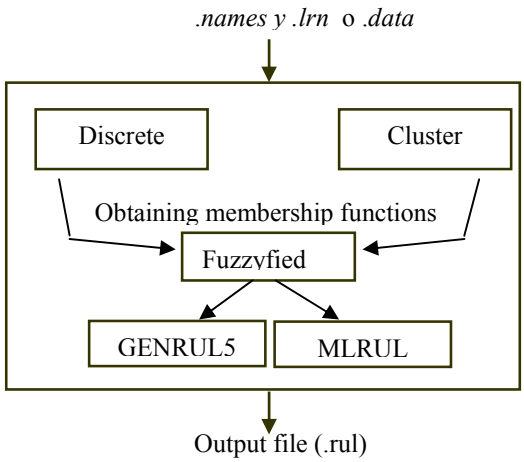


Fig 4. General bisection of the machine learning module

Once the system has recognized the numeric intervals that correspond to each linguistic term, it generates automatically the membership functions and the fuzzy sets.

GENRUL5 and MLRUL are the algorithms developed for generating the fuzzy rules using numeric and nominal data [2].

In the first of these algorithms, the system goes over the whole training set generating the necessary rules. A case is covered if and only if there is an already generated fuzzy rule that has the same consequent or class of the objective feature of the case and the parameters values is equal to a linguistic term that precedes the rule. For instance:

given $rule_p$ and $case_m$:

$rule_p$ **:** ***If*** *x is Tall* ***and*** *y is Old* ***then*** *C1*

$case_m$ **:** *(tall, old, C1)*

So $case_m$ is completed covered by $rule_p$. All the same, a rule is generated to cover every uncovered case of the training set.

In the second algorithm (MLRUL), partitions are created by consecutive arrangements of the training set. The most relevant linguistic variable in the partition is taking as the comparison criteria. This relevance is determined by the use of a mathematic that stimulates homogeneity among cases of the same class and the heterogeneity among cases of different classes. The following methods are proposed to determine the relevance of a linguistic variable:

- Mántaras Distance [14]
- Use of descriptive statistics VCramer [10] [21]
- Computation of the entropy and the acquisition of information [5]
- MLRelevance measure, this is the relevance measure we propose [2]

The algorithm ends when all partitions are closed partitions. A partition is closed if and only if all the cases have the same consequent or class. One of the advantages of this algorithm is that it enables the manipulation of irrelevant feature, absence of information and handling data with noisy information.

This module creates two files. The first one is a file .rul that will be used in the inference process; and the second one is a text file that describes the generated fuzzy rules and its main objective is to explain the inference process.

2.3 Module 3: Inference

This module is used by the system 2nd case of use and it is related to the classification and study of new patients. The process of inference uses the components package FuzzyPack. This package is a platform that enables the generation of fuzzy systems that match the zero order Sugeno model.

A fuzzy system generated using FuzzyPack operates in the first place based on the fact that there is an input file .rul that stores all the information to determine the fuzzy rule system: description of the parameters domain, its relevance information, about the membership functions of those parameters, their values, description of all the system rules including the significance level.

Second, it is based on a connection mechanism that is among components. This mechanism anables the creation of a layers net that depends on the rules file (*.rul*) given as an input parameter to this module.

The first layer of this module is a parameters layer. It has as many nodes as different membership functions.

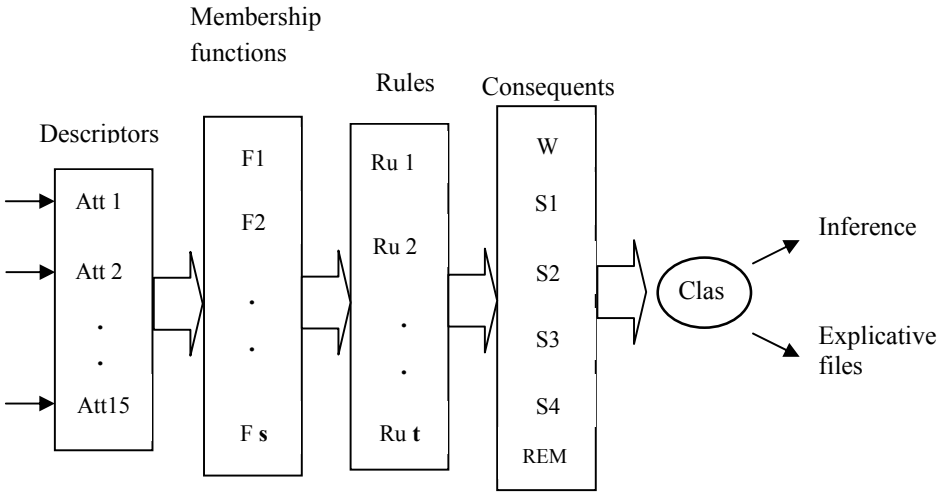


Fig. 5. Bisection of the inference module

Every node of the first layer is connected to all the membership functions link to its linguistic term. There are in the third layer as many nodes as the amount of rules in the file (.rul). The connections between nodes of the second and third layers depend on the existance of the membership functions in every rule. In the fourth layer there is one node for every different consequent in the rules file. That is, there will be 6 consequents, one for every sleep stage (waking, REM, Stage1, Stage2, Stage3, Stage4). All nodes in the third layer that have the same consequent will be connected to the node in the fourth stage that corresponds to that consequent. There is in the fifth layer only one node and all the previous layer nodes are connected to it. Its function is to give a final answer and to for the explanatory files that help to understand the inference of the system (see fig 5).

The components that are represented by the nodes that at the same time are rules, behave as t-norms. Meanwhile, the nodes in the consequents layer behave as co-norms. Many components are implemented. They represent different types of membership functions, several types of t-norms and co-norms.

2.4 Module 4: Inference Corrector

Up to this point, the process of automatic classification takes place independently in each epoch. It means the system determines the sleep stage in each 30-second interval. It does not take into account the relationship between the given stage and it's adjacent. This may alter, in some cases, the logia sleep sequence because, even though the stages are distant from each other, they may have similar physiologic characteristics. That is why, it is necessary to implement corrector module that takes into account the relationship already mentioned.

This module operation is based on a hard-rule system and a case-base reasoning system. The hard rules are supply be experts [4] and the cases are those from the Polysomnogram of some patients that were classified and studied by experts.

The hard rules are contained in the following groups: transition rules from waking stage, from the stage 1, from the stage 2, from the stage 3, from the stage 4, from the REM stage and rules for the elimination of short interruptions.

The rule syntax is: the stages in parenthesis will be correct, not[stage] will be use to state that a stage different from the one assigned to the variable stage.

Examples of rules from each of the rules groups

V, (3 o 4), not[V] => V, 2, not[V]

1, (4), not[3] and not[4] => 1, 2, not[3] *and* not[4]

2, (1), REM => 2, (REM), REM

3,(1), no[3] and not[4] => 3, 2, not[3] and not[4]

4, (3), 4 => 4, 4, 4

REM, (4), 3 => REM, 3, 3

x,x,Y,x,x => x,x,x,x,x

3 Validation

The evaluation is a necessary and important process in the development and testing of any software. The main purpose of this process is to determine the system degree of correct response according to the expectations.

Brender [6] propose some metrics and other measures that express different quality parameters of the medical knowledge. Among the metric measures used in the evaluation of the system we can mention total behavior, behavior conditioned to the class (expected and inferred values), kappa total behavior, kappa behavior conditioned to the class (inference values) and an error functional.

3.1 Elements that influence the classification process

Interexpert's agreement. The Inter.-experts agreement moves from 67% to 91 % [1]. In 10 laboratories in Japan, it moved form 67% to 75.36% [1]. In the Center of Neuroscience of Cuba (CNC) [4], it was 71% for all cases and 72% in the healthy cases. In a similar study, B. Kemp had an interexpert's agreement of 75% in all cases and 77% in healthy cases [11].

Automatic classification of healthy and sick patients: Martin [15] had an interexpert's agreement of 80.8% in a study of 5 healthy subjects, Stanus [20] had an interexpert's agreement of 75% with healthy subjects and 70 % with depressive disorders and insomnia. In the CNC the agreement between the experts and the system was 69% with all cases and 70% with healthy subjects. B. Kemp in a similar comparison, reports 70% and 75%, respectively. This demonstrates that it is easier to classify healthy subjects than subjects suffering from the sleep disease.

Homogeneity of the bases of cases: The classification results are grater when working with polysomnograms of patients suffering from similar diseases.

Analysis of each sleep stages: During the visual classification analysis of 9 healthy case studies, Smith and Karacan [19] stated a 83% of interexpert's agreement. Though during this process stages 1 and REM were analyzed as a whole. On the other hand, on [16], the net efficiency was only tested on stages 1, 2, REM and Waking having as a result a 77.6% of interexpert's agreement. It while analyzing the case studies stages are disregarded, the final results will to not be the effective ones of the classification.

Stages hard to classify: On [17] and [1] the stage one was the hardest to classify. The greatest disadvantages are found in telling apart Stages 1, 2 and REM. Stage 1 is the hardest to classify because it can be mistaken with stages 2 and REM.

3.2 Test

The electrophysiological signals resulting from the 4 patients that were taken to create the bases of the case sets to each patient were: EMG (C3-A12), EEG (O1-A12), EOG (LOG-A12), EOG (ROG-A12) and EMG-25.

Each base of case sets split, having a 70% on training and a 30 % on test (see table 1).

Table 1: Relation of cases by stages, training and test files.

Base of cases	Split	Stage1	Stage2	Stage3	Stage4	REM	Wake
MRLL	Train	11	286	74	151	96	27
	Test	5	123	33	66	42	12
ACGBCLA	Train	13	281	53	180	153	20
	Test	6	121	24	78	67	10
AGSCLA	Train	58	205	83	105	86	200
	Test	26	88	36	46	37	86
ARR	Train	2	439	88	72	100	1
	Test	2	189	39	31	44	1

Analyzing table 2, as a particular case of metrics and functional application for GENRUL5 algorithm having the base of cases ACGBCLA, we can see satisfactory results.

Table 2: Metrics and functional (ACGBCLA) using GenRul5

Metrics and funtionals	Value
Total behavior	0.8736
Total behavior of Kappa	0.7980
Functional of quality	0.8736
Functional of error	0.0081
Weight of error	0.0200

Table 3: Result form apply the corrector module with the MLRul algorithm

Bases of Cases	Without correction (%)	Corrected (%)
MRLL	70.41	79.48
ACGBCLA	81.01	87.37
AGSCLA	74.81	84.37
ARR	68.15	78.86

Table 4: Discretization and clusterization comparisons, using GenRul5 algorithm (test sets)

Bases de cases	Discretization (%)	Clusterization (%)
MRLL	62.98	67.97
ACGBCLA	61.43	71.90
AGSCLA	41.37	64.89
ARR	73.52	73.86

It is very important to analyze each stage within the context of the hypnogram, while classifying automatically the sleep stages of patients. The inference module determines each stage in an independent way but the inference corrector module uses the hypnogram, resulting from the inference module, following the stages transition rules. That is why, we can see on table 3 that corrected results surpass in approximately a 10 % the first results of the classification.

Summarizing, we can see on table 4 that the process of clusterization has higher results than those from the process of discretization, due to the intervals are not specified by the user but obtained automatically from data.

5 Conclusions

The ISSSC version 1.0 system is a helping system for medical diagnosis. This system allows us to create the hypnogram of patients taking into account the electrophysiological signals: EEG, EOG and EMG.

The system is a useful tool while detecting sleep disorders it also helps to spread the expert's expertise. The ISSSC system consist of four modules: signals preprocessing, machine learning, inference and inference corrector module.

Each module works independently from the others. This characteristic of the system allows us to develop different applications using different module combinations. The result of the medical test run on this system, were successful though a greater amount of patients must be tested with this system in order to prove the reliability of the results.

The heterogeneity of the patient's disorders use don the learning module may result in false learning and the results during the sleep analysis. For this reason, we propose a specialized learning process for each kina of sleep disorder aiming to obtain better results.

References

1.	Agarwal, R., Gotman, J.: Computer-Assisted Sleep Staging. IEEE Transactions On Biomedical Engineering, Vol 48, No. 12. (2001)
2.	Arco , L., Caballero, Y.: Aprendizaje automatizado para la generación de reglas borrosas. Trabajo de Diploma. Universidad Central "Marta Abreu" de Las Villas, Cuba (2001)
3.	Aserinsky E, Kleitman N. Regularly occurring periods of eye motility, and concomitant phenomena, during sleep. Science 1957; 118: 273–274.
4.	Barreiro , A.: Clasificación automática de etapas de sueño humano. Trabajo para optar por el título de Especialista de Primer Grado en Fisiología Normal y Patológica.Ciudad de La Habana, Cuba (1995)
5.	Bender, E.: Mathematical Methods in Artificial Intelligence. IEEE Computer Society Press. Los Alamitos, California (1996)
6.	Brender, J.: Measuring quality of medical knowledge. In Proceeding of the Twelfth International Congress of the European Federation for Medical Informatics (1994) 69-74
7.	Cox E., Taber R. and O'Hagan M., The Fuzzy Systems Handbook, Second Edition AP Professional, Paperback, 2nd Bk&Cd edition, August 1998.
8.	Dement, W., Kleitmann, N.: The relation of eye movement during sleep to dream activity; an objective method for the study of dreaming. J. Exp. Psychol, Vol. 55 (1957) 339-346
9.	Himanen SL, Hasan J. Limitations of Rechtschaffen and Kales. Sleep Medicine Reviews 2000; p 149–167.
10.	Jabson J.: Applied Multivariate Data Analisys Vol. 2: Categorical and Multivariate methods. Springer (1992)
11.	Kemp B.: A model-based monitor of human sleep stages. Biol. Cybern (1987)
12.	Kubicki S, Herrmann WM, Ho¨ ller L. Critical comments on the rules by Rechtschaffen and Kales concerning the visual evaluation of EEG records. In: Kubicki S, Herrmann WM (eds). Methods of sleep research. Stuttgart New York: Fischer, 1985: 19–35.
13.	Loomis AL, Harvey EN, Hobart GAI. Cerebral stages during sleep, as studied by human brain potentials. J Exp Psychol 1937; 21: 127–144.
14.	Mántaras, R.: A Distance-Based Attribute Selection Measure for Decision Tree Induction. Machine Learning (1991)

15.	Martin, W., Johnson, L.: Pattern recognition of EEG-EOG as a technique for all night sleep stage scoring. Electroencephalogr. Clin. Neurophisyol, Vol. 32 (1972) 417-427
16.	Oropesa, E., Hans, C., Marc, J.: Sep Stage Classification using Wavelet Transform and Neural Network. International Computer Science Institute (1999)
17.	Park, H., Park, K., Jeong, D.: Hybrid Neural-network and Rule-based Expert System for Automatic Sleep Stage Scoring. Proceedings of the 22nd Annual EMBS International Conference, Chicago IL (2000)
18.	Rechtschaffen, A., Kales, A.: A Manual of Standardized Terminology, Techniques, and Scoring System for Sleep Stages of Human Subjects. Los Angeles, CA: BIS/BRI, Univ. of California (1968)
19.	Smith, J., Karacan, I.: EEG sleep stage scoring by an automatic hybrid system. Electroencephalogr. Clin. Neurophisyol, Vol. 31, (1971) 231-237
20.	Stanus, E., Lacroix, B.: Automated sep scoring: A comparative reliability study of algorithms. Electroencephalogr. Clin. Neurophisyol, Vol. 66, (1987) 448-456
21.	Tom, M. Hill, M.: Machine Learning. (1997) http://www.cs.cmu.edu/~tom/mlbook.html