SLE$_{ad}$: Electronic Tongue Data Analysis System

BRUNO BARRETO BERGAMO[1]
ROGÉRIO EDUARDO GARCIA[1]
JOSÉ ALBERTO GIACOMETTI[2]

[1]DMEC – Departamento de Matemática, Estatística e Computação
[2]DFQB – Departamento de Física, Química e Biologia
FCT – Faculdade de Ciência e Tecnologia
UNESP - Universidade Estadual Paulista
Rua Roberto Simonsen,305
Presidente Prudente (SP) – CEP 19060-900
{brunobergamo, rogerio, giacometti}@fct.unesp.br

Abstract. It was developed an "electronic tongue" – Electronic Tongue System (ETS), in Portuguese **Sistema de Língua Eletrônica** (SLE) –, instrument employing an array of capacitive sensors. Experimental data collected from several sensors are analyzed by Principal Component Analysis (PCA) and Artificial Neural Networks (ANN) techniques using different tools that support each technique separately. This work aims mainly to describe a software tool to perform the data acquisition and the computational tasks involved in data analysis. Software modules were implemented for the PCA and ANN analysis and they were integrated to the SLE instrument. Results obtained here are compared to the ones obtained using other tools, such as WEKA and MatLab, in order to validate the software developed by us.

Keywords: Data Analysis, Pattern Recognition, Artificial Neural Networks, Principal Component Analysis and Electronic Tongue

(Received November 20, 2007 / Accepted December 17, 2007)

1 Introduction

Several industries – chemical, pharmaceutical, agricultural and food – have interest in developing an efficient, low cost instrument to fast analyze and classify complex chemical solutions. Brazilians researchers developed an "electronic tongue" - *e-tongue* hereafter - having a very high sensitivity and the Brazilian Agricultural Research Corporation (EMBRAPA) applied an international patent on it. The e-tongue system comprises an array of capacitive sensors having different responses when immersed in different chemical solutions. The measured values of electric capacitance for each unit represent the variation on physical-chemical characteristics of sampled solutions. The e-tongue name refers to the human tongue, which contain receptor molecules that trigger nerve signals when they encounter taste-imparting molecules thus detecting different tastes (sweet, salty, sour and bitter) [14].

There is a vast range of applications for the e-tongue, for instance: continuous control on product quality; detection of pollutants in water (environmental applications); detection of analytes in low concentration solutions – difficult to be distinguished by human being or even impossible [5]. It is worth to mention that the e-tongue sensors have no specific discrimination of particular substance and it is based on the subjectivity of the analysis miming the human tongue.

The e-tongue system (ETS) uses several sensors fabricated from nanostructured thin films of different polymers that are deposited on the top of an interdigitated micro electrode. The e-tongue instrument is based on the values of the electric capacitance of the sensor units measured by the impedance technique [9, 15]. Measurements over the frequency range from 10 Hz to 10 kHz allow detecting traces of tastants and inorganic contaminants in liquids. Also, several statistical methods have been applied to identify the sampled liquids.

The Principal component analysis (PCA) is a multivariate method that consists in reducing the number of variables and is mostly used due to its simplicity and efficiency. PCA can be used to reduce the dimensionality of a data set retaining the characteristics that most contribute to its variance, i.e., keeping only the lower-order principal components and ignoring the higher-order ones. Such low-order components often contain the "most important" aspects of the data, since the higher-order components represent the same event (high correlation) [11]. The PCA analysis is suitable for identification of different groups based on characteristics that are linearly separable. More sophisticated tools such as multi-layer Artificial Neural Networks [12] should be used to analyze samples with a great complexity. For instance, it was shown that is possible to identify wines considering vintage, producer and type of grape.

Studies applying such techniques on samples obtained with ETS to pattern recognition have been conducted using specific tools. Previous studies applying PCA to data obtained from the ETS has been conducted using MatLab, and Artificial Neural Networks (ANN) by means of Weka (*Waikato Environment for Knowledge Analysis*)[1] and SNNS (*Stuttgart Neural Network Simulator*)[2]. Such tools require input files organized adequately following a pre-defined format. Therefore, the use of external tools introduces an additional step on data analysis: data should have a specific format. Such task could be incorporated on pre-processing stage but different investigations would require different types of files impairing the process of analysis.

This work aims to create a computational solution to collect data – using the ETS – and analyze the data sets, thus: an example of the symbiosis of Computer Science with another domain of knowledge [4], where Computer Science is the main base of this technological advance. Previous studies allowed us to choose the techniques for data analysis and to define the application requirements. This paper is organized as follows: Section 2 presents the e-tongue system; Section 3 presents a brief description of the Principal Component Analysis and Artificial Neural Networks; Section 4 describes the software data analyses modules and highlighting their functionalities and also presenting the results; and to conclude, Section 5 presents conclusions and suggestions for further work.

[1] See www.cs.waikato.ac.nz/ml/weka/

[2] See www-ra.informatik.uni-tuebingen.de/SNNS

2 Electronic Tongue

The name "electronic tongue" refers to an array of sensors that are immersed in liquids, in order to identify their different physical-chemical characteristics, for example, "tastes". The e-tongue can be used in many sectors and it is has widespread application in food and beverage industries and any other trades to monitor the quality of products. In beverage industries, for instance, tests for the taste evaluation is carried out by human tasters, thus they can be assisted by the e-tongue allowing continuous and precise measurements. The advantage of the e-tongue is that there is no decrease on sensitivity during a long period of exposition which does not occurs with human being. A possible application in the pharmaceutical industry would be the search for new compositions that neutralizes the bitter taste in medicines. As pointed out, one can notice the importance of a system like the ETS since it was developed a portable and compact which allows to perform measurements at place. The e-tongue allows to prevent the exposition of human beings to toxic substances or to awkward tastes [12].

The e-tongue system uses an array of sensors made of ultrathin films of polymers such as Langmuir-Blodgett (LB) films of 16-mer polyaniline, polypyrrole (PPy), stearic acid (SA) and composite films of several polymers [14]. Such films were deposited on top of a glass substrate that holds interdigitated microelectrodes. Sensors prepared from different materials produce different electric responses and the their variation is desired since allows a "fingerprint" of the samples [14].

2.1 Electronic Tongue System

The e-tongue system is composed by hardware and software components. The hardware is used for the capacitance measurements of sensorial units of and the software controls the data acquisition, perform the calculations and analyze the electrical signals.

The main hardware components are: signal generator; signal amplifier; multiplexer; data acquisition board and a lap-top computer. The ETS was developed by Cabral [3] and the hardware was conceived to deal with up to eight arrays of sensors, each one comprising up to eight sensor units, thus up to 64 sensor units can be handled simultaneously. The software component deal with electrical signals and provides the capacitance values which are stored into files using a pre-defined format. The software interface, partially presented in Figure 1, allows the definition of parameters to control the data acquisition[3]. The following parameters can be de-

[3] The controls represent the parameters for each array

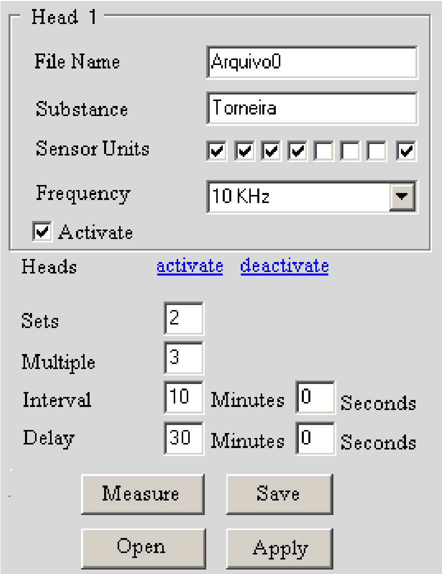


Figure 1: Partial view of ETS acquisition data interface

fined by the user: *file name* – , defining the file name to save data from each array; *substance* under analysis; *sensor units*, to define the sensor units to be considered on measurements; *frequency*, the frequency to be used on measurements – the values can be 10, 100 Hz, 1 and 10 kHz; also, the software indicates the units from which the data acquisition is performed (or not) when an experiment is started for each array used in the experiment *activate*.

As shown at lower part of Figure 1, the software displays the general parameters for the measurement, as follows: *Heads*, to *activate* or *deactivate* all arrays simultaneously; *sets*, to define the number of measures to be performed on each array; *multiple*, to define how many series of measurements will be done; *interval*, allows specifying the time interval (*minutes* and *seconds*) between a series of measurements defined by *multiple* parameter; and *delay*, specify the period to delay an acquisition. In addition, an option in the Menu – *Start All* – allows to start the measurements of all arrays without setting up the parameters – in this case, the previous parameters used in the last measurement are employed.

By combining the response of the sensor units it is possible to obtain enough data to determine which sort of substance is under analysis. For instance, it is possible to differentiate substances with similar taste, like distinct mineral water, coffee and wine. It is important to notice that the good distinction of tastants is made at low level of concentrations, below the human threshold [1, 13, 14]. For the discrimination of the liquids, analyses are conducted using the techniques described in the following.

3 Data Analysis Techniques Applied

The practical use of SLE requires the appropriate analysis of measurements and depends on characteristics observed in the measured group of liquids under analysis. The results of measurements have been analyzed using two techniques: Principal Component Analysis and Artificial Neural Networks.

3.1 Principal Component Analysis

The Principal Component Analysis (PCA) is a multivariate statistical method mostly used for pattern recognition due to its simplicity and efficiency. The PCA techqnique consists in reducing the number of variables on a system which has correlated data – probably the redundancy represents measurements of the same event. By eliminating correlated data only the required data to identify groups are kept [7].

The PCA method consists in rewriting the coordinates of a data group in another system of axes, called Principal Components, making them more convenient to be analyzed. Such new coordinates are obtained by linear combinations of original variables and they are represented on orthogonal axes, in decreasing rank-order of variance. Linear combinations are accomplished in a manner that data can be represented by a smaller number of descriptive factors, reducing the dimension of the group. The total number of principal components is the same of the total number of variables and they show the same statistical information. Usually, the first principal components have more than 90% of the statistical information contained in the original data [10].

The data for the PCA analysis are presented in matrix format, where the lines represent the observations (analyzed samples) and the columns represent the variables. Usually, the First Principal Component (PC1) is able to describe data with largest percentage, the Second Principal Component (PC2) represents the second largest portion of the variability, and so on [8].

3.2 Artificial Neural Networks

Artificial Neural Networks (ANN) are distributed parallel systems composed by nodes (*neurons*), which are simple processing units performing mathematical functions – they are essentially simple mathematical models defining a function $f : X \longrightarrow Y$. The nodes are arranged in layers and linked through connections (*synapses*), usually unidirectional. Weights are associated to the connections and such values are used to weigh the input of each neuron, as a parameter to the mathematical model to be used [2].

An artificial neural network does not have to be adaptive by itself since in order to produce a desired result it requires algorithms that are designed to alter the weight of the network connections. In this sense, the ANN must be trained before its effective use. Therefore, a learning procedure is conducted: a group of examples is submitted to the network in order to prepare it for analyzing phase. The network "extracts" the characteristics necessary to represent the information present in the data set as training. The knowledge remaining in the ANN, represented by weights associated to the connections among nodes, keeps the characteristics used to represent the solutions for the future problems to be analyzed [2, 6].

The ANN is able to generalize an input data set – used for training – to produce results whenever another input data submitted to it, which is the great advantage of this technique. It is possible to notice that an ANN can deal with the identification of groups not linearly separable. However, the disadvantage is the need of training and consequently, a data set for its accomplishment should be known.

3.2.1 ANN Architecture

The definition of the ANN architecture is an important parameter – it restricts the type of problem that can be treated since the structure chosen and the learning algorithm used for training the network are dependent [2]. The following parameters define ANN architecture:

- Number of layers: on networks with only one layer, the input layer projects itself on the output layer. The input layer is not considered, since no computation is accomplished by those neurons – only the input values are attributed. The ANN with multiple layers has more than one neuron connecting the input to the output. The neurons that compose the hidden layers mediate the external input and the output (results obtained) [6];

- Number of nodes in each layer: The amount of neurons contained in each layer;

- Type of the connection among nodes: The network can be classified as non-fully connected and fully connected network. In the connected network, each node is connected to all nodes to the following layer. If one connection is missing, the network is named non-fully connected [2];

- Topology: the topology is defined as how the neurons are connected. The nodes can have *feedforward* connections – the connections between nodes do not form cycles, always is connecting a neuron output to the input of any other neuron in same or in previous layer – or *feedback* connections – the output of a neuron in the i_{th} layer is used as input on nodes belonging to j_{th} layer, where $j \leq i$ [2].

3.2.2 Learning Process

An ANN requires the training before its use. During such process, the ANN extracts information about patterns presented to it: the learning consists in adjusting iteratively the weights associated to the connections. The values of weights represent the knowledge acquired by the ANN, allowing the network recognizing the patterns [2]. The *learning algorithm* consists of well-defined procedures to adjust the weights. There are several learning algorithms, each one using a different heuristic to adjust the connection weights [6]. The main approaches developed are: supervised and unsupervised learning, and their particular cases – reinforcement and learning by competition.

Most of the training algorithms work well in the training using a particular fixed dataset with the correct parameters. However, selecting and tuning a training algorithm on unseen data requires a significant amount of experimentation. Previous studies were conducted to investigate how suitable an algorithm is. Among the learning algorithms, it was chosen the *back-propagation* algorithm which is used on multiple layer networks. This algorithm proposes a way to define the nodes error on intermediate layers allowing weights adjustment. These adjustments are accomplished using the gradient method [2]. For learning, it is necessary to minimize the error function, defined by the sum of the quadratic error and represented as Equation 2:

$$E = \frac{1}{2}\sum_{i=1}^{k}(d_i^p - y_i^p)^2 \tag{1}$$

where E is the total error, k the number of output units, d_i the output expected and y_i is the i_{th} output obtained. This equation defines the total error accomplished by the ANN. The delta rule requires that the activation functions be semi-linear. The activation value is obtained by the Equation 2.

$$y_p^j = f_j(net_j^p) \tag{2}$$

where

$$net_j^p = \sum_{i=1}^{n} x_i^p w_{ji} \tag{3}$$

The constant n represents the number of input connections on node j, and w_{ji} represents the weight of connections between the input x_i^p and the node j.

The training happens in two phases: *forward* (used to define the network output for an input data) and *backward* (use the expected output and the output obtained to update the weights of their connections). In addition, it is possible to use the momentum term to accelerate the training process and to avoid local minima. The use of the momentum term reduces the instability and increases the learning rate [6].

3.2.3 Activation Function

Another important factor to define an ANN is the activation function, which is responsible for converting the linear combination of input values and weights, correctly calculating the neuron output. The mathematical functions mostly used are: Threshold, Linear, Ramp, Signum and Logistics. The Signum function is semi-linear and a limited function used in Artificial Neural Network [6]. It is represented by equation 4:

$$y = \frac{1}{1 + e\frac{-x}{T}} \tag{4}$$

where T represents how soften the curve should be.

4 Data Analysis Modules to SLE$_{ad}$

As extension to the SLE system the software modules for analyzing data were implemented, one for each technique previously presented – both developed in C++, using Borland C++ Builder®. By using the PCA module a straightforward analysis of the samples is possible, just after finishing the acquisition. The PCA module allows choosing the sensors to be considered on the analysis – notice that, by using external tools, the selection of a subgroup of sensors requires an extra step in the pre-processing to exclude data from discarded sensors. Additionally, it is possible to define the matrix to be used on the Principal Component calculus: covariance matrix or correlation matrix.

Figure 2 depicts the class diagram of the implemented PCA module: the component *Graphics* is used to present graphically the results (see Figure 3). In the example, three samples were analyzed – distilled water, salt diluted and sucrose diluted in water, all measured with six sensors - using the covariance matrix. Results are shown graphically: the graphic at left shows that the first principal component is capable to represent 93,93% of sampled data and second one 5,92%; and the another graphic on the right shows the projection of the PCA data using the two first components.

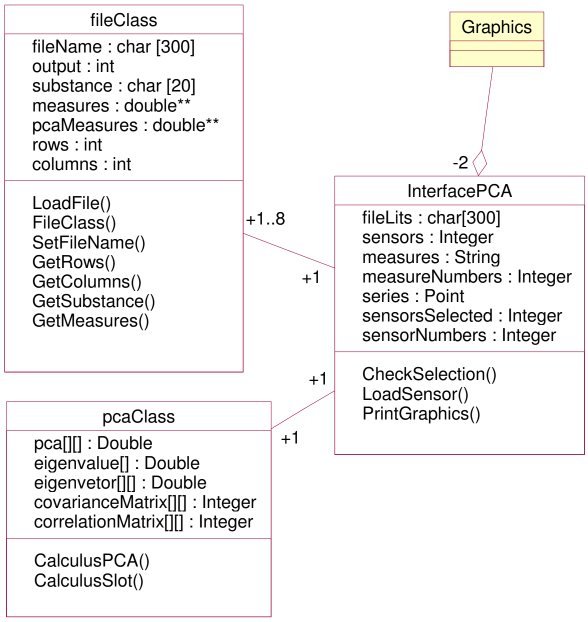


Figure 2: Class diagram of Principal Component Analysis module

The interface of the ANN module was divided in three parts: *Modeling Area*, *Analysis Area* and *Utility Bar*. In the first one, the architecture of the ANN is defined: one creates the neurons, organizes them in layers, and defines the synaptic connections between them. The neuron is represented by a rectangle subdivided vertically in two parts: the right side represents the output and the left side the input. Considering that a neuron might have multiple inputs, the left side is subdivided horizontally to represent them. Straight line segments between neurons represent the synaptic connections. The *Modeling Area* is also used to carry out the ANN tests. In the *Analysis Area* is possible to follow the ANN error during the training process and the results obtained. In *Utility Bar* are given the options to construct, to train and to test an ANN – parameters as maximum error, learning rate, number of iterations and to enter the *momentum*.

The ANN module is composed by three classes representing the neuron, the neural network and the modeling area, named *Neuronio*, *RedeNeural* and *RNAreaModelagem*, respectively. The *arquivoClass*, the same used in PCA module, is used to read the input values from files. *Neuronio* class is created from *TImage* class and it is used to show the neurons in the *Modeling Area*. Besides the attributes from *TImage* class, the *Neuronio* class has attributes to represent an artificial neuron, as: number of inputs, weight of each one, activation function value, error value and a flag to identify the neuron

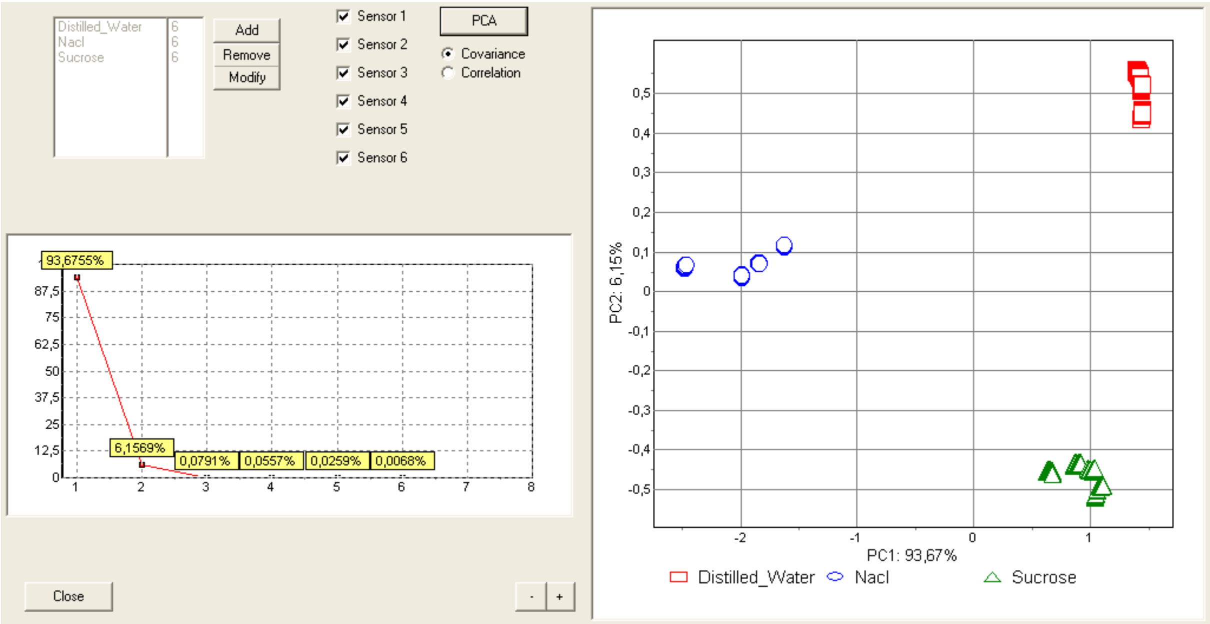


Figure 3: Interface of Principal Component Analysis module

type (input, output or hidden layer neuron). The *RedeNeural* class has network attributes and it is referred when a new network is created or when a network is restored from files. Figure 4 depicts the ANN class diagram.

As in the PCA module, each training file is responsible to show an output (samples for an expected result), and each file has the number of sensors sampled representing the amount of input values for the ANN. Such parameters are used to create *Neuronio* objects, both to input and output neurons. After defining the ANN architecture the training phase can be initiated. To conduct the validation test, the input values are read from files following the format established in the data acquisition system. After chosen the neuron output, the ANN test is conducted using the same standard input, the *net* function calculation and activation function used in training phase.

4.1 Results Obtained

To validate the software modules implemented, the results are compared to the results obtained with different tools, and presented in the following.

4.1.1 Principal Component Analysis

To obtain the data from sensorial units immersed in a liquid the SLE was programmed to carry out measurements for each sensorial unit of one e-tongue after 30 minutes of immersion required to the stabilization of capacitance readings. This interval is necessary for stabilization of the double electric layer formed in the interface between the electrode and the liquid under analysis [16].

Sucrose solutions in different concentrations were tested – $1\mu M$, $1nM$, $1pM$ and $1fM$ –, taking a series of 100 measurements using eight sensorial units which were also previously immersed in the solution during 30 minutes, for the stabilization of the capacitance readings. Results were analyzed using the PCA module from the MatLab software and the PCA module implemented, and the principal components obtained by them were identical. Also, it is possible to observe on the graphics presented in Figure 5 the same distribution of different samples of sucrose in well defined groups.

In another test, samples with different concentrations of caffeine ($1mM$, 1μ, $1n$) in MiliQ water were prepared and five sensorial units were used in the experiment. For each concentration twenty measurements were performed for each solution of caffeine and five measurements for the MiliQ water. The results for the components percentages (see Figure 7) are relatively low, not allowing the samples to be distinguished, i.e., the displayed PCA data do not show any grouping (see Figure 6).

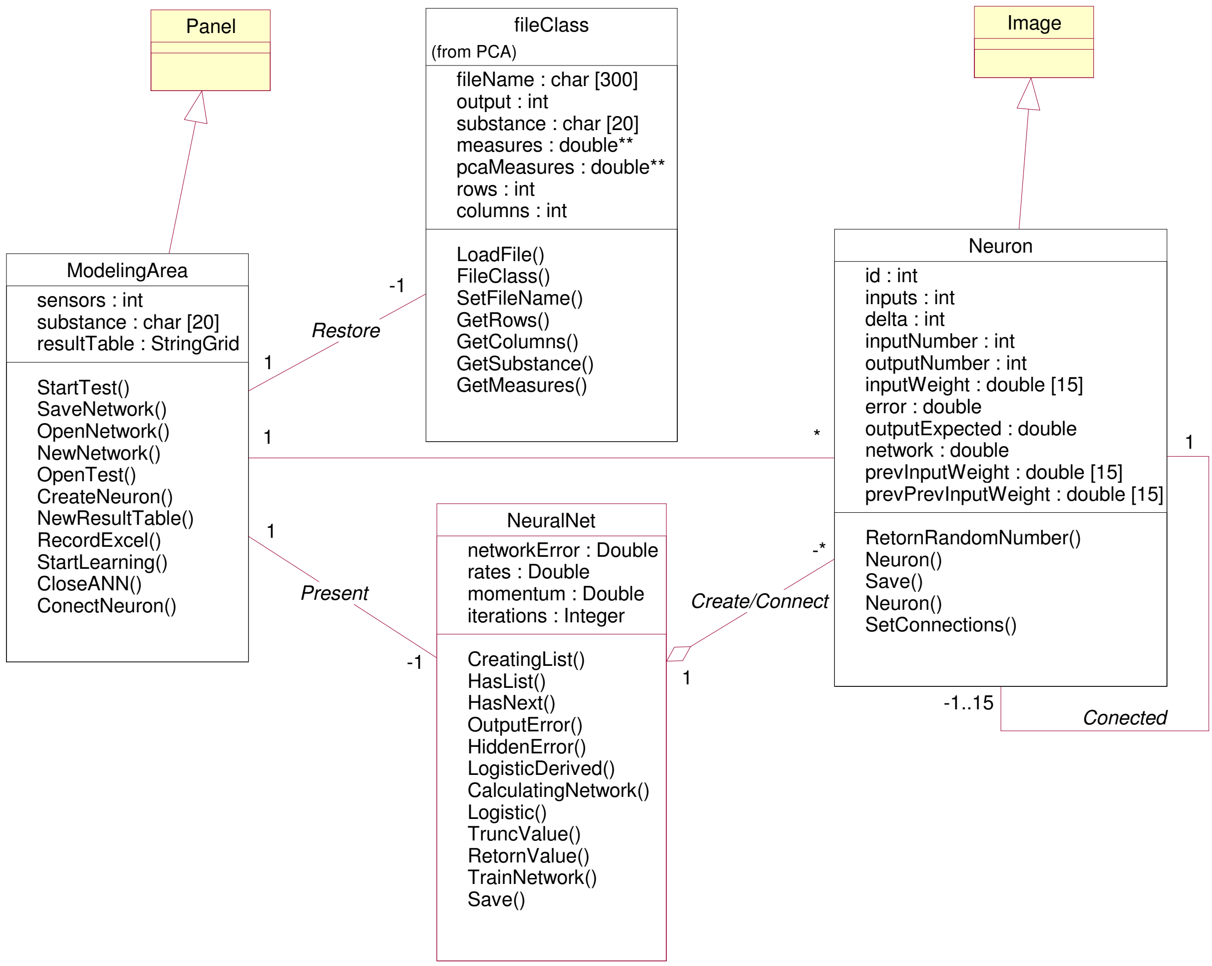


Figure 4: Class Diagram of Artificial Neural Network module

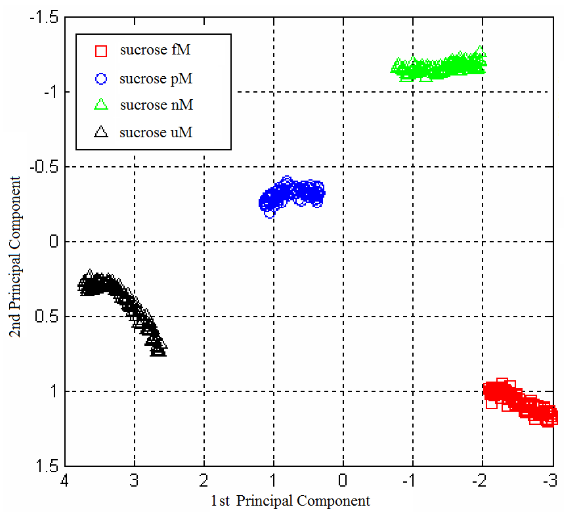
4.1.2 Artificial Neural Network

Using the ANN module and the software *Weka* the training of several different neural networks were conducted to validate the software module. Selected tests presented here are named *Experiment 1* and *Experiment 2*, and both used an artificial neural network completely connected with only one hidden layer that was trained using *back-propagation* algorithm. Moreover, trainings using different learning rates and different number of iterations were also conducted.

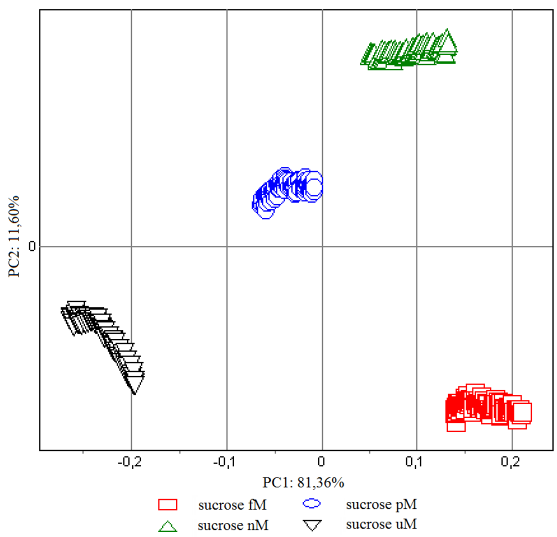
For the *Experiment 1* three linearly separable data sets were used that represent samples of water distilled – with 300 elements –, NaCl and sucrose diluted in water – samples with 400 elements each – obtained from six sensorial units. Each set was randomly separate in two subsets: one with 60% used during training phase and second with 40% used on the test phase [2]. The architecture of this net presented in the Figure 8 is composed by six inputs – referring to the unit sensors sampled –, three output neurons – one for each sample – and four neurons in the hidden layer – arithmetic mean values of the input and output neurons [2]. Four training were carried out using different values for the parameters *learning rate* and *number of iterations* – 0,2/800, 0,15/1000, 0,15/600 and 0,09/1000, respectively. Results obtained with ANN module and the software *Weka* were the same (see Table 1).

In the *Experiment 2* different samples of three wines from different producers and harvests were used, obtained with eight sensor units. It has been taken into account: 160 samples of *AC 2000*, 150 samples of *ACS 2000* and 150 samples of *CH 2000*[4]. It was used 67% of samples in the training phase and 33% had been used in the test phase. The ANN architecture was defined with eight neurons in the input layer – referring to the

[4]The wines identification was omitted

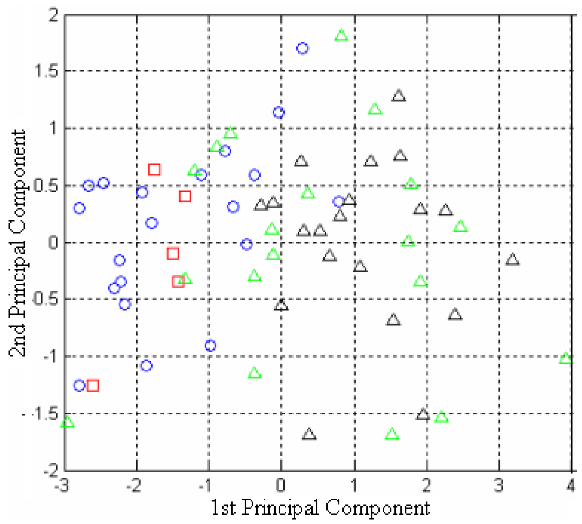


(a) MatLab

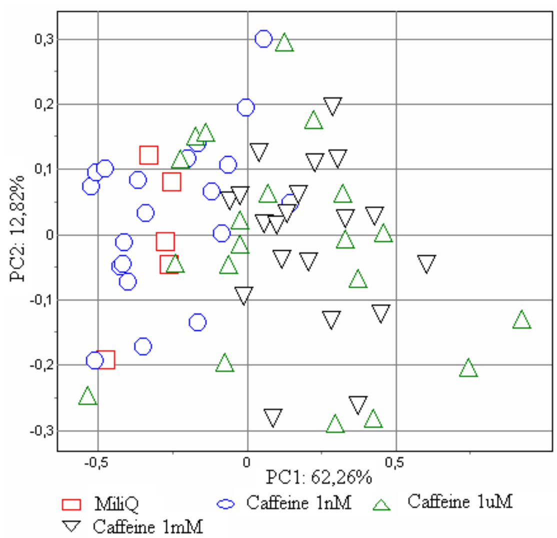


(b) PCA module

Figure 5: Principal Component graphics using sucrose samples



(a) MatLab



(b) PCA module

Figure 6: Principal Component graphics using samples of caffeine and MiliQ water

number of sensors –, three output neurons and five in the hidden layer. The results obtained (correctness rate) were the same for both – software *Weka* and the ANN module –, as presented in Table 2.

4.1.3 Remarks on results obtained using Artificial Neural Network module

The reduction of number of variables carried through the PCA allows to find if the results from the substances under analyses can be linearly separable or not. For samples not linearly separable, the ANN technique should be applied. Figure 9 presents the PCA graphic of *Experiment 2* showing that no grouping was found and should be analyzed by the Artificial Neural Networks. The results obtained by applying PCA, presented in Figure 9,

Table 1: Results obtained with samples of distilled water, *NaCl* and sucrose

	Training		Weka (%)		ANN Module (%)	
T	Rate	Iteration	Error	Correct	Error	Correct
1	0.2	800	0	100	0	100
2	0.15	1000	0	100	0	100
3	0.15	600	0	100	0	100
4	0.09	1000	0	100	0	100

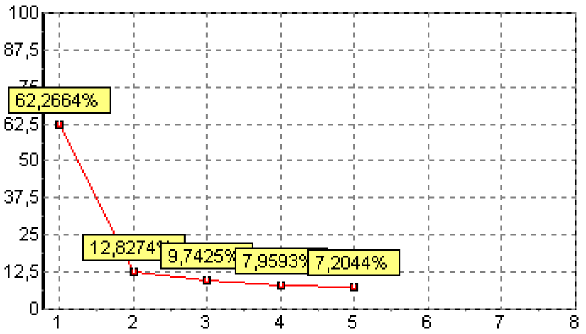


Figure 7: Graphic of principal component percentage variation using samples of caffeine and *MiliQ* water

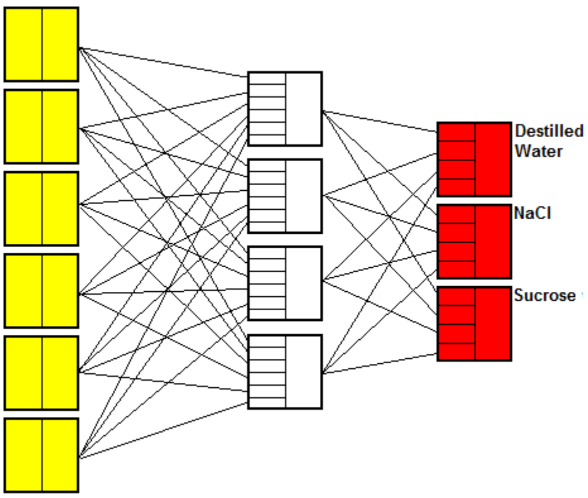


Figure 8: ANN architecture used with samples of distilled water, *NaCl* and sucrose

Table 2: Results obtained by using software WEKA and Module ANN for three different varieties/brands of wines

	Training		Weka (%)		ANN Module (%)	
T	Rate	Iteration	Error	Correct	Error	Correct
1	0.08	3000	0	100	0	100
2	0.1	3000	0	100	0	100
3	0.15	3000	0	100	0	100
4	0.2	2000	0	100	0	100

shows that the samples of *CH 2000* are projected separately. Results for ANN applied to the wine samples *ACS 2000* are found at center of the figure, while the wine samples *AC 2000* are found at the left and at below the wine *ACS 2000*. Therefore, the ANN module permits the identification of all samples.

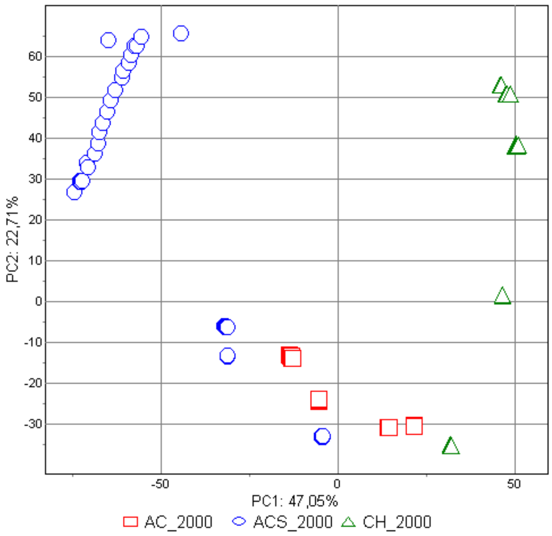


Figure 9: PCA obtained from data sets used on *Experiment 2*

5 Conclusion and Further Works

The computational tools described here are part of the research conducted in the Department of Physics, Chemical and Biology (DFQB), Faculty of Science and Technology of São Paulo State University (FCT-Unesp). The integration of the analysis modules to the ETS instrument which performs the data acquisition aims to facilitate the scientific research. The PCA module has been used by students and researchers of the DFQB supporting their research activities. It is worth to mention that the data preparation task was eliminated from the data analysis. The software allows the selection of the results obtained from several liquid samples – excluding samples from discarded sensors on the analysis – aiding the researchers since it is not necessary to generate several files data, each one having different combination of sensors to be analyzed.

The Artificial Neural Network (ANN) module has a user-friendly graphical interface to facilitate the definition of the artificial neural network. The use of the ANN module on data analysis allows one to speed the ANN tasks: no restriction on the number of hidden layers and numbers of nodes in each layer; it does not require artificial neural network completely connected; and the artificial neural network can be stored for posterior use with their weights saved. In addition, the input files for the training process and for analyzing, are read from the ETS files – data acquisition module – using the same file format.

Despite of the software is coupled to the ETS instrument it is possible to use the PCA and ANN modules for analyzing data obtained with several experimental setups. In this case it is necessary to use files formatted according to the standards defined for the ETS. In addition, the analysis of the data is easier since no external software is needed.

As further work new improvements are suggested to the softwate – for instance, 3D graphical presentation for PCA, creation of “bias” bound to input files on ANN, the implementation of learning algorithms derived from *back-propagation* as the *RProp* and *QuickProp*.

References

[1] Borato, C. E., Riul Jr, A., Carvalho, A. C. P. L. F., Fonseca, F. J., and Mattoso, L. H. C. Electronic tongue – science mimicking nature. *First World Congress on Synthetic Receptors*, 2003.

[2] Braga, A. P., de Carvalho, A. C. P. L. F., and Ludermir, T. B. *Redes Neurais Artificiais: Teoria e Aplicações*. Ed. LTC, Rio de Janeiro, 2000.

[3] Cabral, F. P. A. *Desenvolvimento de instrumentação para uso em “Língua Eletrônica”*. PhD thesis, Instituto de Física de São Carlos, Universidade de São Paulo, 2006.

[4] de Carvalho, A. C. P. L. F., Brayner, A., Loureiro, A., Furtado, A. L., von Staa, A., de Lucena, C. J. P., de Souza, C. S., Medeiros, C. M. B., Lucchesi, C. L., e Silva, E. S., Wagner, F. R., Simon, I., Wainer, J., Maldonado, J. C., de Oliveira, J. P. M., Ribeiro, L., Velho, L., Gonçalves, M. A., Baranauskas, M. C. C., Mattoso, M., Ziviani, N., Navaux, P. O. A., da Silva Torres, R., Almeida, V. A. F., Jr, W. M., and Kohayakawa, Y. Grandes desafios da pesquisa em computação no Brasil – 2006-2016, 2006. disponível a partir de `www.sbc.org.br`, acessado em Maio/2007.

[5] Ferreira, M., Riul Jr, A., Wohnrath, K., Fonseca, F. J., Osvaldo N. Oliveira, J., and Mattoso, L. H. C. High-performance taste sensor made from langmuir-blodgett films of conducting polymers and a ruthenium complex. *Analitical Chemistry*, 75(4):953–955, 2003.

[6] Haykin, S. S. *Redes Neurais – Princípios e Prática*. Editora Bookman, Porto Alegre, 2001.

[7] Johnson, R. A. and Wichern, D. W. *Applied regression analysis and other multivariat*. Prentice Hall, New Jersey, 1978.

[8] Jolliffe, I. T. *Principal Component Analysis*. Springer-Verlag, 2nd edition, 2002. ISBN 0387954422.

[9] Mattoso, L. H. C., Venâncio, E. C., Fonseca, F. J., Mello, S., Riul Jr, A., and Taylor, D. M. Patente internacional (PI: 0103502-9), 2002.

[10] Moita Neto, J. M. and Moita, G. C. Uma introdução à análise exploratória de dados multivariados. *Química Nova*, 21:467–469, 07 1998.

[11] Pla, L. E. *Analisis Multivariado: Metodo de Componentes Principales*. Secretaria Geral da Organização dos Estados Americanos, 1986.

[12] Riul Jr, A. Ciência imitando o corpo humano. *Revista Physicae*, 3:39–46, 2002.

[13] Riul Jr, A., dos Santos, D. S., Wohnrath, K., Tommazo, R. D., Carvalho, A. C. P. L. F., Fonseca, F. J., Oliveira, O. N. J., Taylor, D. M., and Mattoso, L. H. C. Artificial taste sensor: efficient combination of sensors made from langmuir-blodgett films of conducting polymers and a ruthenium complex and self-assembled films of an azobenzene-containing polymer. *Langmuir*, 18(1):239–245, 2002.

[14] Riul Jr, A., Malmegrim, R. R., Fonseca, F. J., and Mattoso, L. H. C. An artificial taste sensor based on conducting polymers. *Biosensors and Bioelectronics*, 18(11):1365–1369, October 2003.

[15] Riul Jr, A., Mattoso, L. H. C., Fonseca, F. J., Taylor, D. M., Mello, S., and Venâncio, E. C. Patente nacional (PI 0103502-9) depositada no INPI, 2001.

[16] Taylor, D. M. and Macdonald, A. G. Ac admittance of the metal/insulator/electrolyte interface. *Journal of Physics D: Applied Physics*, 20:1277–1283, 1987.