Learning and Nonlinear Models (L&NLM) – Journal of the Brazilian Neural Network Society, Vol. 8, Iss. 3, pp. 125-134, 2010.
© Sociedade Brasileira de Redes Neurais (SBRN)

FEATURE SELECTION VIA GENETIC ALGORITHMS IN THE CLASSIFICATION OF ANTI-SNAKE VENOM MEDICINAL PLANTS

Lariza Laura de Oliveira[1], Gabriela Felix Persinoti[2], Silvana Giuliatti[2] and Renato Tinós[1]

[1]Grupo de Informática Biomédica, Departamento de Física e Matemática, Faculdade de Filosofia, Ciência e Letras de Ribeirão Preto, Universidade de São Paulo
[2]Departamento de Genética, Faculdade de Medicina de Ribeirão Preto, Universidade de São Paulo

{larizalaura@usp.br, gabi.felix@gmail.com, silvana@rge.fmrp.usp.br, rtinos@ffclrp.usp.br}

Abstract- In this work, Genetic Algorithm (GA) is employed in feature selection for the classification of medicinal plants with snake venom-neutralizing properties. The classification is performed using an Artificial Neural Network (ANN), which indicates the medicinal plants with anti-snake venom action as output when an amino acid sequence of snake venom is presented in its input. GAs and ANNs are Artificial Intelligence techniques and have been used in several similar optimization and classification problems. Here, the feature selection system is implemented using the classification error rate of the training set and the number of attributes as the fitness of each individual of the GA. The validation results for the classification system indicate that ANNs can be used to aid the selection of medicinal plants with snake venom-neutralizing properties. Also, feature selection based on GAs can help researches to select amino acids sequences of the snake venoms which can be important to the interaction with medicinal plants compounds.

Key words- Bioinformatics, Genetic Algorithms, Artificial Neural Networks, Artificial Intelligence, Snake venom, Medicinal plants.

1 Introduction

Snake bites envenomation is considered a serious public health problem, not only in Brazil, but in all Latin America [1]. Snake venoms are complex combinations of proteins including: phospholipase A2 (PLA2), proteolytic enzymes, and others. Frequently, envenomation by snake bites is treated by antiophidic serum administration. However, many times, patients do not have fast access to the serum since accidents usually occur in remote places. Furthermore, the local damage induced by snake venoms, as myonecrosis for example, can be sometimes irreversible [2].

In this sense, some fast procedures can help the patient until the usual treatment with serum can be administrated. One of these procedures is the use of medicinal plants extracts, which can be found close to the accident local. In the Brazilian popular medicine, many plants are employed in snake bites treatment, however few have their effects scientifically investigated [1].

Artificial Intelligence (AI) techniques have been employed with success in different bioinformatics problems as sequences analysis, protein structure prediction, and sequences alignment [3] and can be useful to this problem too. In this paper, feature selection based on Genetic Algorithms (GAs), which are an AI technique employed in optimization problems, is used to select attributes for classification of medicinal plants with snake venom-neutralizing properties. The classification is performed by another AI technique, Artificial Neural Networks (ANNs), which should indicate a medicinal plant with anti-snake venom in its output when an snake venom amino acid sequence is presented in its input.

Two are the main objectives of this work. The first one is to generate a classification system (software) that relates a medicinal plant with a venom protein. It can aid researches to explore new relations and associate plants which were not studied yet to other plants with known anti-venom properties. The second main objective is to select important features of the input data in order to improve classification. Then, the feature selection could indicate which subsets of amino acids present in the snake venoms proteins are the most important in the interaction with the medicinal plants. Such information is important to better understand the snake venom-neutralizing properties of medicinal plants compounds.

In the next section, the materials and methods used in this work are described. Then the experimental results with two medicinal plants are described. It is important to observe that the AI system proposed here can be extended in order to cope with more plants and snakes venom. This paper intends to show that the construction of an AI system to classify and select features in the problem of relating medicinal plants and snake venoms is possible.

2 Material and Methods

2.1 Data

The medicinal plants and snake venom amino acid sequences were selected based on [4] and [1]. Table 1 shows the relation between the selected medicinal plants and snake species and some antiophidian properties of the selected plants described in [1]. Two medicinal plants were used: *Casearia sylvestris*, popularly known in Brazil as "*Guaçatonga*" or "*Erva-de-bugre*" [5] and *Eclipta prostrata*, popularly known as "*Erva-botão*" [6].

Table 1: Relation between Medicinal Plants and Snake species [1].

Plant species	Antiophidian properties	Snake species
Casearia sylvestris	Anti-PLA$_2$ Antiedema Antimyotoxic Antihemorrhagic Antilethality	*B. alternatus, B. jararacussu, B. moojeni, B. neuwiedi, B. pirajai, C. d. terrificus*
Eclipta prostrata	Anti-PLA$_2$ Antihemorrhagic Antiproteolytic Antimyotoxic	*C. rhodostoma, B. Jararaca, B. jararacussu, L. muta*

The selected snake species were: *B. alternatus, B. jararaca, B. moojeni, L. muta, B. neuwiedi, B. pirajai, C. rhodostoma,* and *C. d. terrificus*. The species *B. Jararacussu* was not employed in this work because both plants (*Casearia sylvestris* and *Eclipta prostrata*) have antiophidian properties that interact with the compounds of its snake venom. The amino acid sequences of venoms' proteins were acquired from the public database maintained by National Center for Biotechnology Information (NCBI). A total of 94 protein sequences of the selected snake species were obtained in FASTA format.

In the classification system, one of the two selected medicinal plants should be indicated according to an amino acid sequence presented in the input of the ANN. In this way, the classifier (ANN) has two outputs, one for each medicinal plant. The direct use of the protein sequence as classification engine input is not suitable because the sequences have different length. The technique used to encode the protein sequence was based on the *n-gram* method [7]. Here, the *n-gram* value of each amino acid subsequence consists in computing occurrences of *n* consecutive amino acids in a protein sequence. In the original method, the *n-gram* value consists in dividing the number of occurrences by the protein length, but in this paper the absolute values of each occurrence are used as classification engine inputs. For example, the *2-gram* values for the amino acid sequence *ACLVAC* is: 2 for *AC*, 1 for *CL*, 1 for *LV*, 1 for *VA,* and 0 for the remaining subsequences of two amino acids. Thus, there are, respectively, 20, 400 (20^2), and 8000 (20^3) different values for *1-gram, 2-gram,* and *3-gram* codification, if 20 amino acids are considered. Hence, the size of the classifier input is the same for every protein.

The values of *1-gram, 2-gram,* and *3-gram* codification were used as classifier (ANN) inputs, i.e., the *n-gram* values consist in computing the occurrences of one, two, and three consecutive amino acids in a protein sequence. Consequently, the total number of subsequences with one, two, and three amino acids that can be used as inputs (possible features) for the classifier is 8420. In order to reduce the number of attributes for the classification task and to identify the subsets of subsequences that are most important for the classifier, feature selection is applied to the data.

2.2 Feature Selection

The simplest method to select a feature subset is to employ exhaustive search. This method test one classifier for each feature subset and the performance obtained for each one is compared. Considering that *N* is the total number of possible features in the data, exhaustive search implies in testing 2^N combinations. When *N* is large, like in this problem, the number of combinations is too large, what makes prohibitive the use of exhaustive search. In this way, other methods for feature selection have been proposed.

The most known methods to extract features are probabilistic methods [8]. Nevertheless, these methods need knowledge about the probability distributions and are computationally complex. Alternative methods evaluate features according to their importance in distinguishing objects and/or computing the correlation between each pair of features [9]. Another approach involves the use of Principal Component Analysis (PCA) for feature selection [10]. However, the use of correlation analysis requires knowledge about the probability distributions and large sample databases.

Feature selection can be seen as an optimization problem, where an optimal subset of attributes should be found. GAs have been successfully employed in similar optimization problems. The use of GAs to select attributes for ANNs is not new and has been reported in the literature [11], [12], [13] and [14]. In such problems, GA individuals represent a subset of features. For each individual, a classifier is constructed, trained, and tested. The classification error rate over the training or test set is then used by the GA as fitness function to evaluate the individuals.

2.3 System with GA and ANNs

The GA based system was implemented in Java language. Each GA individual represents a subset of features used as the set of attributes for the ANN.

Each GA individual is encoded as a binary array, where each array's position represents the presence or absence of a feature. Thus, each individual is represented by a binary array with 8420 positions. An "1" in the *n*-th position of the array of individual *i* indicates that the *n*-th subsequence of amino acids are present in the ANN *i*.

The employed ANNs are Multi Layer Perceptrons (MLPs), trained with backpropagation algorithm [15]. The MLP is composed by a set of neurons, each one representing a nonlinear function, distributed in layers. In general, training an MLP means to find a set of weights (represented by real numbers) that connect the neurons of two consecutive layers. The utilized MLP has only one hidden layer with a fixed number of neurons and one output layer with two neurons (one for each class). Figure 1 shows an example of an MLP built for a given individual.

In order to evaluate the GA individual, i.e., the quality of the features subset encoded in the array of the individual, one different MLP is built, trained, and evaluated. Two different strategies to compute the fitness (evaluation) of an individual were tested in this work. In the first strategy, the fitness of the individual *i* of the GA population is given only by the classification error rate over the training set for the *i*-th MLP, built with the inputs specified by the array of individual *i*. In the second strategy, the fitness of the individual *i* is given by a linear composition of the classification error rate over the training set for the *i*-th MLP and of the number of attributes. The second strategy was adopted to further reduce the number of selected attributes.

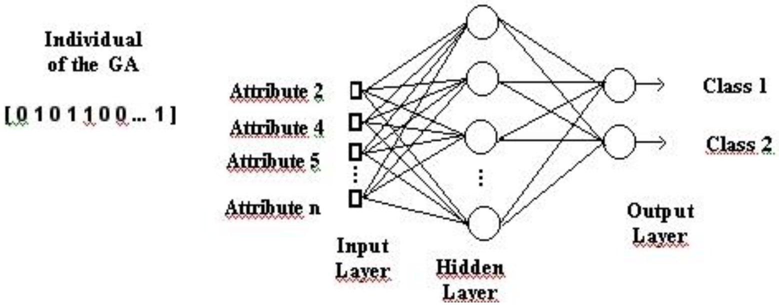


Figure 1. Example of the codification. The *i*-th MLP is built according to the array of individual *i*.

After the evaluation of the individuals in a population, the solutions (individuals) are selected to compose the next population according to their fitness. Higher fitness implies higher probability of selection. The employed selection methods are tournament selection and elitism. Elitism consists in the selection of the best individual of the population, which is not modified. In this way, the individual with the best fitness value will always survive. Tournament selection is then used to select the remaining individuals of the population. For each new individual, a group of individual of the population is randomly chosen and the individual with the best fitness is selected.

Then, individuals selected by tournament are modified according to transformation operators. Here, crossover and mutation are used. In the crossover, parts of the arrays of two selected individuals are exchanged according to the crossover rate. In the mutation, each selected individual's gene (element of the array) is mutated with probability given by the mutation rate. When a gene is mutated, its value is changed, i.e., if the stored value is 1, it goes to 0, and vice versa.

Figure 2 shows the GA based feature selection system. Initially, a population of individuals (problem solutions, i.e., subsequences of features) is randomly generated. The maximum size of each individual was defined as 20. In this way, the number of features of each individual is randomly chosen between 1 and 20 and then the features were randomly chosen in the binary array with 8420 positions, but considering the same probability into the sets of 1-gram, 2-gram, and 3-gram.

Each individual of the population is then evaluated by computing the classification error of the MLP built and trained with the attributes specified by the individual. In a second time, to generate the next population, individuals of the population are selected by elitism and tournament, and are modified according to crossover and mutation. These steps are repeated until a maximum number of iteration is reached.

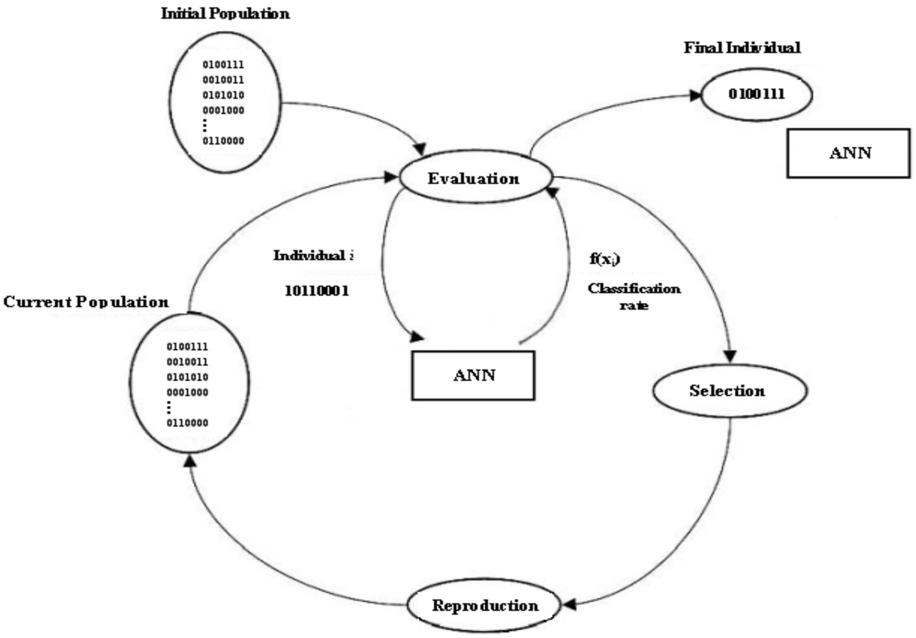


Figure 2. GA based Feature selection system.

3 Results

The results of two experiments sets are presented in this section. In the first experiments set (experiments with fitness function 1), the fitness function of individual *i* is given by the classification error rate over the training set of the MLP built and trained with the attributes (subsets of sequences with one, two, or three amino acids) specified by the *i*-th individual. In the second experiments set (experiments with fitness function 2), the fitness function is given by the weighted sum of the same classification error rate employed in the first experiments set and the number of features presented in the individual.

In the following experiments, the individuals of the current population were selected using tournament and elitism. Crossover with rate 0.6 and flip mutation with rate 3/*m*, where *m* is the length of the chromosomes, were employed. Each GA was executed for 100 generations, and the number of individuals in each population was equal to 100, except for the experiments to define fitness function parameters, where the population size was equal to 50.

Each MLP used is built according to:

- Input layer: the total of inputs was equal to the number of attributes for the tested individual, i.e., the number of inputs was equal to the number of ones in the chromosome of the individual.
- Hidden layer: only one hidden layer was used. According to previous tests, the number of hidden neurons was set to 20.
- Output layer: the number of output neurons was equal to 2, one for each class (*Casearia Sylvestris* and *Eclipta prostrata*).

Also, the MLP was trained during 100 iterations. The learning and momentum rates employed were respectively equal to 0.01 and 0.9. The initial weights were randomly generated.

The 94 selected sequences were divided in two sets. The first one, with 70% of the examples, is the training set, which was used to train the MLP and to compose the fitness function. The second one is the test set, with the remaining examples, which were presented to the MLP built according to the array of the best individual found in the last generation of the GA in order to test the generalization of the classification.

3.1 Experiments with fitness function 1

The first experiments set was executed considering the fitness function as the percentage of the incorrectly classified patterns of the training set (classification error rate) by the MLP built and trained with the attributes specified by individual *i*. In this way, the fitness function, which should be minimized during the evolutionary process of the GA, for individual *i* is given by:

$$f(i) = e(i) \qquad (1)$$

where *e* (*i*) is the classification error rate over the training set for the MLP specified by individual *i*. As the initial population and the initial weights of the MLP are random, each experiment was repeated 9 times with different random seeds. Table 2 shows the results for each random seed for the best individual in the last generation of the GA. The classification rate (number of correctly classified examples) for the training and test sets and the number of attributes related to the best individual are presented in Table 2. The median, mean, and standard deviation (std) are also presented.

Table 2: Results of the experiments with fitness function 1.

	Classification rate for the training set (%)	**Classification rate for the test set (%)**	**Number of Attributes**
seed 0	89.394	71.429	18
seed 1	89.394	71.429	18
seed 2	89.394	71.000	7
seed 3	92.424	67.857	19
seed 4	90.909	82.143	7
seed 5	93.939	71.429	14
seed 6	89.394	75.000	17
seed 7	90.909	64.286	19
seed 8	90.909	78.571	12
median	90.909	71.429	17
mean	90.741	72.572	14.556
std	1.597	5.371	4.876

The best results for the classification rate for the training set, which is inversely related to the fitness of the best individual, was obtained in the experiment with random seed 5 (Table 2). However, one can observe that the best classification rate for the test set was obtained for seed 4. In this way, it is possible to observe that the best classification rate for the training set does not necessarily means the best classification rate for the test set.

Table 3 shows the amino acids subsets selected for the best individual found in each execution.

Table 3: Selected attributes in the experiments with fitness function 1.

Seed	**Selected attributes**
0	M, Q, R, WM, QA, MQ, LG, IE, HQ, EH, CM, AW, AR, WAA, VYV, TPW, QEN, MHT
1	M, N, Q, R, S, W, Y, WE, NR, IN, HW, FW, DS, AY, YDL, QNF, LNW, IFA
2	VV, PK, NQ, HY, WSA, VTW, PKS
3	H, K, N, YR, WN, WD, MC, FW, ET, EC, CC, CA, YDG, WSL, VRG, SIY, NTW, KYF, GKC
4	H, K, Y, RE, NR, CH, WQY
5	E, G, K, L, R, V, IL, GL, FW, DT, CL, YRP, TWF, RKR, LLV, GDM
6	H, I, L, W, Y, TC, SR, RF, MG, IV, VAM, SLP, EFQ
7	H, L, M, P, Q, T, W, Y, MH, IS, FC, EC, CC, AW, AA, WVF, KEE, HHP, ENQ
8	H, N, S, PI, IN, GQ, FA, CM, CL, AY, SGK, DKL

3.2 Experiments with fitness function 2

In order to reduce the attributes in the MLP, the fitness function was changed by adding a term to punish individuals with many attributes. In this way, the new fitness function for individual *i* is given by a linear function composed by the classification error rate over the training set, *e*(*i*), for the *i*-th MLP and number of attributes (number of ones in the array), *l*(*i*), for individual *i*. Thus, the fitness of the individual *i* is given by:

$$f(i) = ae(i) + bl(i) \qquad (2)$$

where the parameters *a* and *b* are non-negative real numbers.

Some experiments were initially done in order to define the best parameters of the fitness function given by Eq. (2). Nine combinations of *a* and *b* values were tested. Table 4 shows, for the best individual in each execution, the classification rate for the training set and the number of attributes specified by the individual in these experiments, which were executed with 50 individuals in each population.

Table 4: Results of the experiments to determine the parameters *a* and *b* for fitness function 2.

A	***B***	**Classification rate for the training set (%)**	**Number of Attributes**
0.1	0.9	57.576	1
0.2	0.8	72.727	3
0.3	0.7	80.303	3
0.4	0.6	77.273	5
0.5	0.5	75.757	3
0.6	0.4	80.303	4
0.7	0.3	83.333	5
0.8	0.2	84.848	9
0.9	0.1	87.879	15

One can observe that the attributes number generally increases when the value of *b* decreases. The values *a*=0.8 and *b*= 0.2 were chosen in the remaining experiments presented in this section because they correspond to a good compromise between a high classification rate and a small number of attributes. Table 5 shows the experimental results for each random seed for the best individual of the GA for fitness function 2 with *a*=0.8 and *b*= 0.2. The classification rate (the number of correctly classified examples) for the training and test sets and the attributes number related to the best individual are presented in Table 5. The median, mean, and standard deviation (std) are also presented.

Table 5: Results of the experiments with fitness function 2 with *a*=0.8 and *b*= 0.2.

	Classification rate for the training set (%)	**Classification rate for the test set (%)**	**Number of Attributes**
seed 0	92.424	71.429	19
seed 1	87.879	71.429	3
seed 2	95.454	64.286	12
seed 3	90.909	75.000	5
seed 4	89.394	64.286	4
seed 5	89.394	64.286	4
seed 6	87.879	78.571	8
seed 7	89.394	64.286	9
seed 8	89.394	64.286	8
median	89.394	64.286	8
mean	90.236	68.651	8
Std	2.409	5.584	5.050

Table 6 shows the amino acids subsets selected for the best individual of the population for each execution.

Table 6: Selected attributes in the experiments with fitness function 2 and *a*=0.8 and *b*= 0.2.

Seed	Selected attributes
0	H, L, M, T, YN, TS, SH, RE, QE, PK, LW, FG, EW, ET, EA, CD, YCV, DWS, AGQ
1	VA, RW, MH, DR, DF, TMH
2	W, QV, PH, PE, LC, GK, ET, EI, STK, FQP, DKE, DDV
3	H, IV, CH, YSA, NDK
4	MV, DR, CC, KWK
5	SN, RA, QC, PV, MQ, MD, KH, GK, CF, SAS, QLE, GQC, CWM
6	H, V, RF, IN, GD, CP, ITC, DSG
7	M, N, R, VE, SF, PK, ME, MYF, HDW
8	W, YD, SR, YVV, LGI, FGF, EQK, APE

4 Analysis of the Results and Conclusions

One can observe that, according to tables 2 and 5, the mean number of attributes is smaller for fitness function 2. This fact occurs because the attributes number information is used in fitness function 2, enabling the GA search for solutions with high classification rate and small attributes number. The use of the number of attributes as a criterion for the optimization is important to this problem because, as can be observed in the experimental results, different subsets of subsequences of one, two, and three amino acids (tables 3 and 6) can generate classifiers with good performance. This occurs because the number of possible subsets of subsequences of amino acids is much larger than the number of examples. In this way, the classifier can use different sequences in order to reach good performance and the number of attributes reduction can indicate a smaller subset of subsequences to be analyzed.

According to tables 3 and 6, some attributes were selected (isolated or in a subsequence of two or three amino acids) by both experiments. Histidine, Triptophan, Methionine, Lysine, and Tyrosine residues were frequently selected in the *1-gram*, *2-gram* and *3- gram* subsets. In the proteins set used in this work, there are mainly Phospolipases (PLA_2), Serin proteinases, and Myotoxins. Studies involving PLA_2 have been increasing mainly due to the interest in myonecrosis research [1], [16].

In [17], where *Bothrops moojeni* venom was used to study PLA_2, chemical modifications were introduced in residues in the follows positions: Histidine 48, Triptophan 77, Tyrosine 119, Tyrosine 52, Lysine 49, and Methionine 8. These modifications affected toxic and pharmacological effects of PLA_2s. Similar results were obtained using different snake venoms. The chemical modification (carboxymethylation) of the Methionine 8 (M8) residue, which is highly conserved in PLA_2s of snake venoms [17], isolated in *Pseudechis Australis* PLA_2s [18] decreased the lethality and the enzymatic effects. The presence of 9 conserved residues of Tyrosine in PLA_2s suggests that they perform some important function in the molecule [16], [19]. The modification of Tyrosine 52 (Y52) and Tyrosine 119 (Y119) showed that these residues could be related to mytotoxic and neurotoxic activities. The chemical modification of Triptophan 77 (W77) suggests this residue may have participation in neurotoxic activities [17]. In [20] and [19], tests using *Bothrops moojeni* venom showed a possible relation of Lysine 36 (K36) and Lysine 38 (K38) residues with neurotoxic, myotoxic, cytotoxic, and bactericidal effects.

Among 18 PLA_2s sequences used in this work, 3 sequences have the H48 residue conserved and 10 have the M8 residue conserved. Neither of PLA_2s sequences used here have Y52, but 4 sequences have Y119, while only one PLA_2 sequence used here has W77. The analysis of PLA_2s used here showed that 8 sequences have K38, but none K36.

In this work, some amino acids selected by the GA were reported in literature as important to the interactions between medicinal plants and snake venom. The amino acids selected by GA with importance recognized are: Histidine, Lysine, Metionine, Triptophan, Tyrosine. Also, the amino acids Asparagine and Phenylalanine, reported in the literature as important, were selected in the *2-gram* and *3-gram* subsets. However, it is important to observe that, in general, the attributes selected by the GA are those which can help to discriminate between the two subsets of venoms sequences, i.e. the subset of sequences classified as *Casearia Sylvestris* or *Eclipta Prostrata*, and, many times, these subsequences do not coincide with the subsequences of amino acids with recognized importance in the interaction between the venom and the substances in the plants.

In order to investigate why the attributes shown in tables 3 and 6 were selected by the GA, a multiple alignment was performed with all protein sequences employed here, using ClustalW2 software [21] available at EBI (European Bioinformatics Institute) homepage. The default software's parameters were used. The obtained alignment tree (Figure 3) shows that the sequences employed in this work can be arranged in groups. In Figure 3, the sequences classified as *Casearia Sylvestris* are shown in red, while the sequences classified as *Eclipta Prostrata* are shown in black.

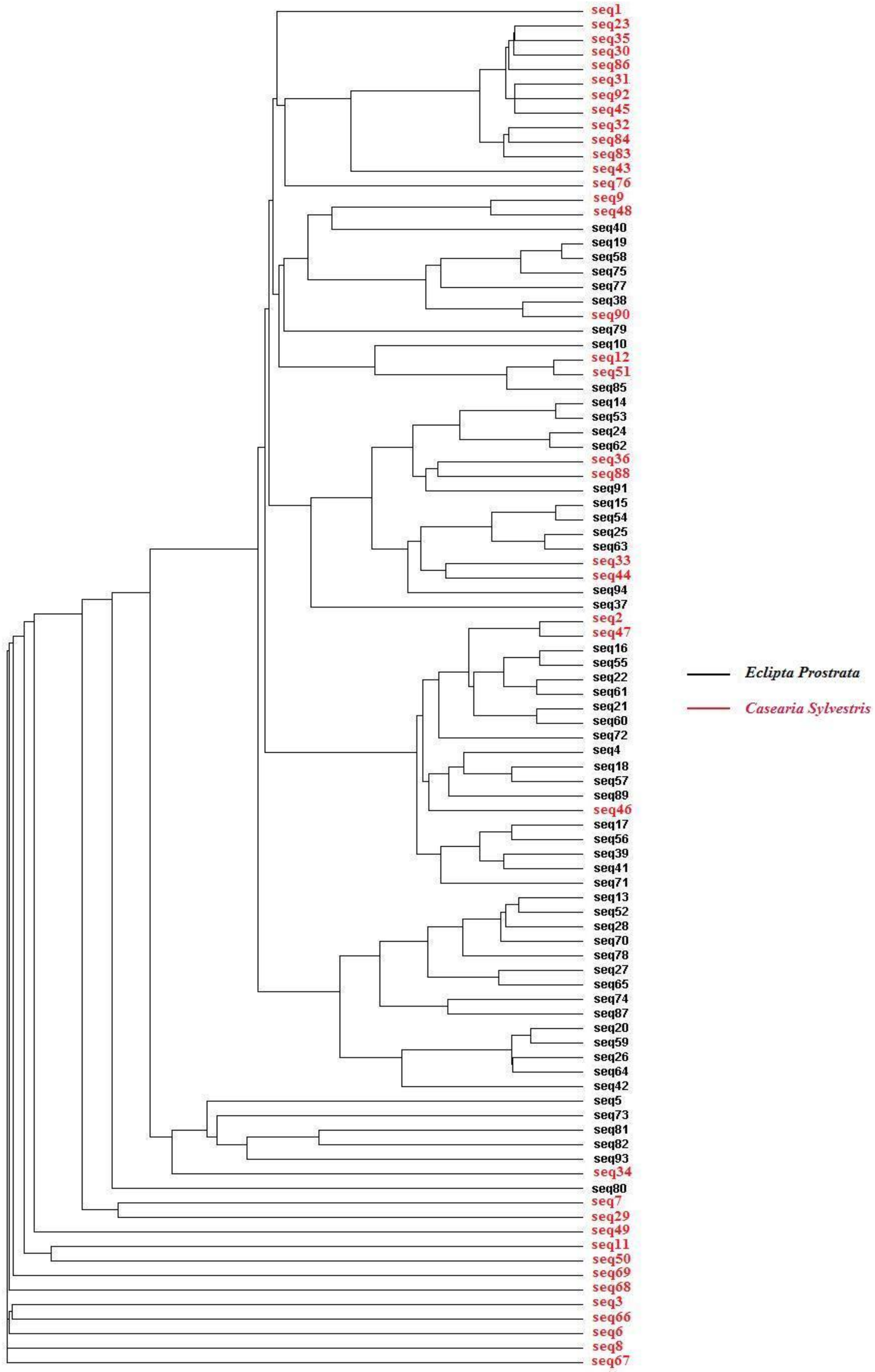
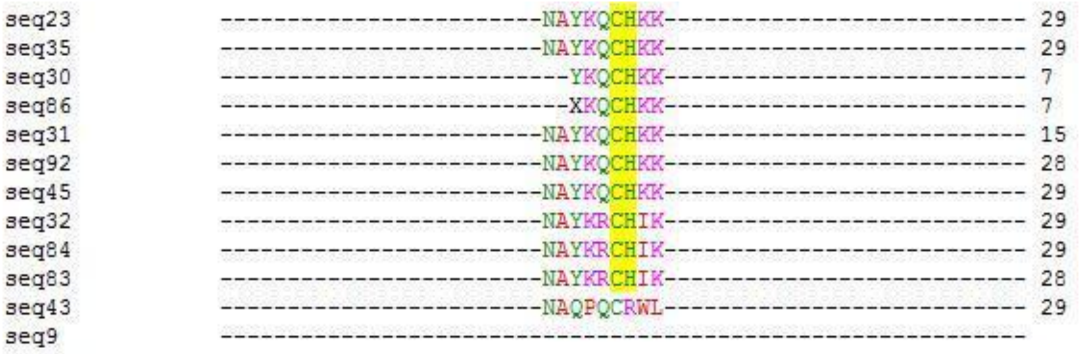


Figure 3. Alignment tree.



```
seq23           ------------------------NAYKQCHKK--------------------------- 29
seq35           ------------------------NAYKQCHKK--------------------------- 29
seq30           --------------------------YKQCHKK--------------------------- 7
seq86           --------------------------XKQCHKK--------------------------- 7
seq31           ------------------------NAYKQCHKK--------------------------- 15
seq92           ------------------------NAYKQCHKK--------------------------- 28
seq45           ------------------------NAYKQCHKK--------------------------- 29
seq32           ------------------------NAYKRCHIK--------------------------- 29
seq84           ------------------------NAYKRCHIK--------------------------- 29
seq83           ------------------------NAYKRCHIK--------------------------- 28
seq43           ------------------------NAQPQCRWL--------------------------- 29
seq9            ------------------------------------------------------------
```

Figure 4. Alignment fragment.

The alignment indicates that some amino acids are conserved in some snake venoms' sequences and can be used to classify the sequences related to *Eclipta Prostrata* and *Casearia Sylvestris* classes. Some of these amino acids shown in the alignment coincide with those selected by the GA as attributes for the classification. It is an expected result because the GA selects amino acids subsequences important to discriminate sequences related to the two classes considered in this work. For example, the alignment indicates that the amino acids subsequence CH can be used to classify the sequences 23, 35, 30, 86, 31, 92, 45, 32, 84 and 83 (Figure 4 shows with more details this fragment of the alignment). These sequences clustered by the alignment belong to the *Crotamine* family, a Neurotoxin from *Crotalus Durissus Terrificus*, related to *Casearia Sylvestris* class in this work. The subsequence CH was selected by the GA for random seed 3 (Table 6), with others subsequences, as important to the data classification. Moreover, the subsequences DSG, CP, GD and IN selected by GA and shown in the alignment were found in sequences related to *Eclipta Prostrata*, while the subsequences CC were founded in sequences related to *Casearia Sylvestris*. However, it is important to observe that the alignment shown in Figure 3 is not the only way to separate the classes. Other groups of subsequences can be used to discriminate sequences related to the two plants, as can be observed in the results obtained by the GA.

The analysis of the results indicates that the system with GA and ANN is able to select features important to classify venom protein sequences in medicinal plants classes, as it can be seen in the obtained classification rates. The results still showed that some amino acids had been reported in the literature as important to the relation between medicinal plants and snake venoms.

This paper indicated that the construction of an AI system to classify and select features in the problem of relating medicinal plants and snake venoms is possible, and can help researches to select amino acids sequences of the snake venoms which can be important to the interaction with medicinal plants compounds.

A possible direction for future work is to investigate whether other amino acids selected by GA have relevance to the interaction among the medicinal plants and the snake venom's proteins studied here.

5 Acknowledgments

The authors would like to thank CNPq and FAPESP (Proc. 2007/02791-4 and 2008/07098-8) for the support to this project.

6 References

[1] Soares, A. M.; Januário, A. H.; Lourenço, M. V.; Pereira, A. M. S. and Pereira, P. S. (2004). Neutralizing effects of Brazilian plants against snake venoms. *Drugs of the Future*, 29(11): 1105-1117.

[2] Varanda, E. A. and Gianini, M. J. S. M. (1994). Bioquímica de venenos de serpentes. In: *Barravieira B (ed.), Venenos animais: uma visão integrada,* Editora de Publicações Científicas, Rio de Janeiro: 205-223.

[3] Baldi, P. and Brunnak, S. (1998). *Bioinformatics: the machine learning approach*. MIT Press, Cambridge, USA.

[4] Santos, G. F.; Tinós, R. and Giuliatti, S. (2006). Classification of snake venom-neutralizing effects of medicinal plants via artificial neural networks. *Proc. of the 14th Annual Int. Conference on Intelligence System for Molecular Biology* (ISMB'2006), Fortaleza.

[5] Silva, S. L.; Calgarotto, A. K.; Chaar, J. S. and Marangoni, S. (1998). Isolation and characterization of ellagic acid derivatives isolated from *Casearia sylvestris* SW aqueous extract with anti-PLA2 activity. *Toxicon*, 52(6): 655-666.

[6] Mors, W. B.; Nascimento, M. C.; Parente, J. P.; Silva, M. H.; Melo, P. A. and Suares-Kurtz, G (1989). Neutralization of lethal and myotoxic activities of South America rattlesnake venom by extracts and constituents of the plant *Eclipta prostrata (Asteraceae)*. *Toxicon*, 27(9): 1003-1009.

[7] Wang, J. T. L.; Ma, Q.; Shasha, D. and Wu, C. H. (2001). New techniques for extracting features from protein sequences. *IBM Systems Journal*, 40(2): 426-441.

[8] Therrien, C. W. (1989). *Decision, estimation and classification*. John Wiley and Sons, New York, USA.

[9] Looney, C. G. (1997). *Pattern recognition using neural networks*, Oxford University Press.

[10] Krzanowski, W. J. (1987). Selection of variables to preserve multivariate data structure using principal components. *Applied Statistics*, 36: 22-33.

[11] Yang, J. and Honavar, V. (1998). Feature subset selection using genetic algorithm. In: *Motoda H and Liu H (eds.), Feature extraction, construction, and subset selection: a data mining perspective*, Kluwer, New York, USA: 117-136.

[12] Oliveira, L. S.; Sabourin, R.; Bortolozzi, F. and Suen, C. Y. (2003). A methodology for feature selection using multi-objective genetic algorithms for handwritten digit string recognition. *Int. Journal on Pattern Recognition & AI*, 17: 903-929.

[13] Yao, X. (1999). Evolving artificial neural networks. *Proc. of the IEEE*, 87(9): 1423-1447.

[14] Favretto, F. O.; Tinós, R. and Carvalho, A. C. P. L. F. (2006). Selection of sensors in an artificial tongue via genetic algorithms. *Proc. of the Workshop on Computational Intelligence* (WCI'2006),Ribeirão Preto, Brazil.

[15] Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Macmillan, New York, USA.

[16] Soares, A. M. and Giglio, J. R. (2003). Chemical modifications of phospholipases A2 from snake venoms: effects on catalytic and pharmacological properties. *Toxicon*, 42: 855-868.

[17] Amui, S. F. (2006). *Do laboratório ao virtual: desenvolvimento de um banco de dados de venenos de serpentes brasileiras e análise computacional de estruturas primárias de fosfolipases A2*. Dissertação de Mestrado, Faculdade de Ciências Farmacêuticas de Ribeirão Preto, Universidade de São Paulo.

[18] Takasaki, C.; Sugama, A.; Yanagita, A.; Tamya, N.; Rowan, E. G. and Harvey, A. L. (1990). Effects of chemical modifications of Pa-11, a phospholipase A2 from the venom of Australian king brown snake (Pseudechis australis), on its biological activities. *Toxicon*, 28: 107-117.

[19] Soares, A. M., Fontes, M. R. M. and Giglio, J. R. (2004). Phospholipases A2 myotoxins from Bothrops snake venoms: structure-function relationship. *Current Organic Chemistry*, 8(17): 1677-1690.

[20] Lomonte, B.; Angulo, Y. and Calderón, L. (2003). An overview of lysine-49 phospholipase A2 myotoxins from crotalid snake venoms and their structural determinants of myotoxic action. *Toxicon*, 42: 885-901.

[21] Larkin, M. A.; Blackshields, G.; Brown, N. P.; Chenna, R.; McGettigan, P. A.; McWilliam, H.; Valentin, F.; Wallace, I. M.; Wilm, A.; Lopez, R.; Thompson, J. D.; Gibson, T. J. and Higgins, D. G. (2007). ClustalW and ClustalX version 2. *Bioinformatics* 23(21): 2947-2948.