Fuzzy Logic

There are two main advantages of applying fuzzy logic to the analysis of biological patterns and protein function (Zadeh, 1965). First, fuzzy logic inherently accounts for noise in the data because it extracts trends, not precise values. And a lot of noise is what can be found when searching for biological function, a quite loosely defined term that only makes sense in a context. Second, in contrast to other automated decision making algorithms, such as neural networks, algorithms in fuzzy logic are cast in the same language used in day-to-day conversation which makes fuzzy logic predictions easily interpretable. Third, there is no learning phase of the training set; that is, the system is no forced trying to learn the rules. The rules came from the knowledge of the system. Nevertheless, Neural networks can be used as a tuning method for the FAM phase of the controller. Finally, fuzzy logic algorithms are computationally efficient and can be scaled to include an unlimited number of components. Thus they are able to recognize biologically but not clear-cut important patterns (Woolf and Wang, 2000; and Perez-Iratxeta *et al.*,2002)

The fuzzy logic algorithm takes a decision as folows. The sequence properties previously calculated, the numerical data are classified into two main categories: "Profile Parameters" and "Numerical Parameters". These data are converted to fuzzy data in a process called "fuzzyfication" converting numerical parameters into linguistic values, which can be viewed as fuzzy sets. In our case, from a protein sequence fuzzy logic integrates several features from sequence data into single rules in order to include as much as possible protein information from a remote homology search. For example, in a pairwise alignment from a PSI-BLAST output, several remote homologue protein sequences can show different "degrees of membership" to loosely defined sets of variables (Length, Flexibility, Amino acid composition and Hydropathic profile). Let us to consider "Length". The Fuzzy system defines each numerical input variable (the respective protein sequence lengths) in the form of three fuzzy sets ("Low", "Medium" and "High"). The system give numerical values between 0 and 1 (degree of membership) to each one of the protein sequences found in the pairwise alignment.

Then, three fuzzy sets are determined: LOW length MEDIUM length and HIGH length. Consider two proteins, A of 500 amino acids and B of 70 amino acids, sharing some degree of similarity (Figure 1b). The protein A could present a degree of membership close to *1* in the fuzzy set HIGH length, and close to *0* to in the fuzzy set LOW length. For protein B the situation is the contrary to A: protein B would present a degree of membership close to *0* in the fuzzy set HIGH length and close to *1* in the fuzzy set LOW length. Next, a serie of fuzzy rules are determined. The control law is described by a knowledge-based algorithm consisting of IF … THEN rules with vague predicates and a fuzzy logic inference mechanism. The base-rule is formed by a group of logical rules that describes the relationship between the input and the output of the controller.

The knowledge encoded in the base-rule is derived from the theoretical and practical understanding of the dynamics of the controlled object (Yager and Filev, 1994), then, in our case, the base rule is derived from the implication of amino acid composition, segment length and KD and flexibility profiles contributed to the protein function. However, some rules may contribute little, or never contribute to the controller performance and could be excluded from the rule base.

Each of the rules of the FLC is characterized by an IF part, called “premise”, and a THEN part, called the “consequent”. The premise of a rule contains a set of conditions, the consequent contains a conclusion. If the conditions of the premise are satisfied, then the conclusions of the consequent apply. Then, a decision matrix is constructed with fuzzy rules in order to compare the data included in these rules. This is process is called the “decision-making logic”, which simulates human decision-making and which infer fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic. Finally, in a process called “defuzzification”, the qualitative fuzzy data is transformed to numerical value (in a scale from 0 to 100) that is usable to reorder the PSI-BLAST output in terms of biological function.

Finally, what can we consider an accurate functional prediction for a query protein sequence? The definition of protein function has become someway complex. Following the GOC suggestions, function can be related to a biological process (i.e. cell cycle), a molecular event (i.e. hydrolysis of a specific substrate), a cellular components (i.e. membrane; mitochondrion), etc. In the present work we have taken, as positive matches, only those describing a molecular function, in a narrow or broader description (i.e., Na+ transporter or ion transporter), reported on the annotation of the target protein found by PSI-BLAST.

ALGORITHM

The algorithm and program by default proceeds as follows: Upon running PSI-BLAST each of the output sequences, is analysed according to their properties used to feed the Fuzzy Logic Controller (FLC) (Amino acid composition, Segment length, Hydropathic profile and Flexibility profile). Then, this properties are introduced as inputs in the FLC.

In order to use the amino acid composition of a protein in a numerical value, we used the Euclidian distance as a classification tool. Euclidian distance is a dissimilarity index. It evaluates the difference in terms of amino acid composition between the query protein and each subject protein from the database.

The amino acid composition of a protein is defined by the vector:

$$F(x)=[f1(x), f2(x), f3(x), \ldots, f20(x)]$$

The proteins are classified in a 20-dimensional space (amino acids). The amino acid frequency of one amino acid *i* in a protein is defined by:

$$f_i(x)=\frac{n_i}{N}(i=1,2,3,4,\ldots,20)$$

Where n_i= frequency of *i* amino acid and *N*= number of amino acids in this protein. Euclidian distance is defined by:

$$d_{ij}=\sum_{k=1}^{p}(f(subject)_{ik}-f(query)_{jk})^2$$

Segment lengths from each (query and subject) protein are named as ***m*** and ***n*** respectively.

Profiles uses two analyses: the hydropathic profile according to the algorithm of Kyte and Doolittle (Kyte and Doolittle, 1982), and the flexibility profile according to the algorithm of Karplus and Schultz, (1985).

Fuzzyfication step

The numerical input data from sequence calculation are converted to fuzzy data. Fuzzyfing is transform numerical, first normalizing from 0 to 1, then the normalized values is broken up into various membership functions. For example, "HI", "MED", "LO" as a function of the normalized value.

The membership function in each fuzzy binary relation is Q(x,y)

$$\mu_Q(x,y)=\frac{|X\ intersect\ Y|}{|X\cup Y|}$$

The decision making logic step

The FLC can be looked upon as a system that, like its inputs, has the variables that are included in the antecedents of the rules x_i (**Profiles parameters** and **Numerical parameters**), and like an output (*y)*, the variable that is included in the consequents (**Protein function**). It is formed by a family of logical rules that describes the relationship between the input and the output of the controller.

IF x_i is B_{i1} AND x_n is B_{in}, THEN y is D_i

In general, there are no systematic tools for forming the rule base of the controller (Zadeh, 1965). Usually is used the intuitive knowledge and the experience because the controller is designed as a simple expert system. In our case, the rules are based in the aggregation of the factors with maximum value gave the rule with maximum value

IF *Sequence length* is *HIGH* **AND** *KD profile* is *HIGH* **AND** *Flexibility profile* is *HIGH* **AND** *Amino acid composition* is *HIGH*, **THEN** *Function* is *HIGH*

And the rule with minimum value is

IF *Sequence length* is *LOW* **AND** *KD profile* is *LOW* **AND** *Flexibility profile* is *LOW* **AND** *Amino acid composition* is *LOW*, **THEN** *Function* is *LOW*

The combination of the all the possible situations of the inputs (That is, all input with all combinations of values LOW, MED and HIGH), after several simulation runs, gives a Fuzzy Associative Matrix. (FAM). It's a Fuzzy truth table that shows all possible outputs for all possible inputs.

No training set are needed because our controller is a Self organizing controller and uses the rules of maximum value and of minimum value to make the combinations of them and automatically give the weights to each one. The rules that contribute little or nothing to the final value were eliminated from the system.

The next step is the determination of the individual rule output, which has been symbolised as **F**. The Mamdani method has been used .

Defuzzification step

The output inferred by the base-rule cannot be used directly, because we need a numerical number, not a fuzzy number. The process of selecting a single optimal point from the domain of a fuzzy membership function is called defuzzification. The Maximum height method has been used in this program.

Finally, the fuzzy logic values, numerical data, lead to a rearrangement of the initial PSI-BLAST profile, and sequences, sometimes from the bottom of the initial PSI-BLAST profile, BYPASS those with better Fuzzy scores climbing to the lead positions, the top in most cases. They suggest a putative function for the query protein in terms of fuzzy logic not of E-value or mathematical scoring.