Adaptive Fuzzy Systems

Background

As of now the "smartness" of our fuzzy machines are dependent on the rules given. The greater the number of rules, the "smarter" the machine gets. However, this means that the performance of the fuzzy machines is restricted by the capabilities of the human brain. Therefore, how do we make the machines think for themselves and come up with rules of its own?

Consider the way the human beings learn. We all learn through experience and through experience we become smarter. Whether, it is the smell of lime, or the picture of our mother, we remember things as it is given to us. With memory, we improve on our actions or thoughts and by defintion become smarter. Fuzzy logic can be applied the same way. Instead, of depending on humans to put specifc fuzzy rules to deal with every situation, the machine should be able to produce its own rules through experience. This can be done with the *Data In Rules Out*(DIRO) method.

Data In Rule Out

The DIRO method is simple, put data in through a black box and rules come out as shown in the figure below.

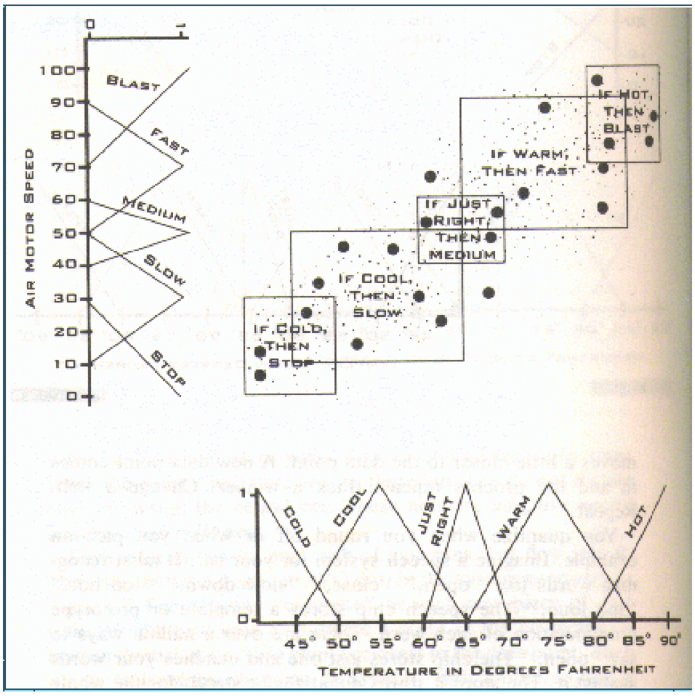
Data In Rules Out

Nueral networks, which fill the black box is beyond the scope of this article. However, it is important to note that it is neural networks, which acts like the eyes and ears of an adaptive fuzzy system whose rule changes with experience. The adaptive fuzzy system tunes its rules as it samples new data. At first the rules change fast. This lets the fuzzy system find a working set of fuzzy rules. Then with more samples, the rules change and fine tunes itself and as the saying goes practice makes perfect.

Simple Case Study: An Adaptive Fuzzy Air Conditioner

As an example, consider an air-conditioner controller, which controls the speed of its motor according the the temperature of the surrounding. Fuzzy subsets and member functions are written for this application in the similar way as before. The two subsets used are the temperature of the surrounding as inputs and speed of the motor as outputs. As usual the two sets overlap to produce fuzzy patches. Because at first, five rules are defined, there are five patches. Each patch will cover data which are represented here as points. Using an expert, we can place logical data points on each patch of the system. The expert adjusts the speed of the motor according to what temperature of the room she or he feels appropriate.

The nueral nets used in this example are called the *adaptive vector quantizers* (AVQ) which are suppose to come up with rules by itself. Each web of nueron in the system is defined as AVQ points. As data comes in, the AVQ point tries to move closer to it. The nuerons then "compete" with each other and "wins" if its AVQ point is closest to the data. How the AVQ points move about is again beyond the scope of this report.
The whole system can be seen in the figure below. The big black dots represent the AVQ point and the small dots represent the data points.



Adaptive Fuzzy System

The number of rules taken now, simply depends on the number of AVQ points in the rule cells. If a cell has an AVQ point, the *that* rule is added add to our list of rules. Some cells which have more than one rule in it, are used more often making them more inportant. Note: It is more feasible usually to have not more than two rules per cell as this cuts down on bad rules. With more data the expert is no longer needed as the system starts generating its own rules. With practise the system may enventually win over the expert.

Conclusions

This method is just one of the many methods used to generate rules in an adaptive sytem. Though a simple example was used to illustrate the system, it is important to realize how powerful this system is. In Japan, research is currently being made to use adaptive systems to model events in politics, history, medicine and even military planning. Imagine in the future, we may even have a Fuzzy British Prime Minister !!