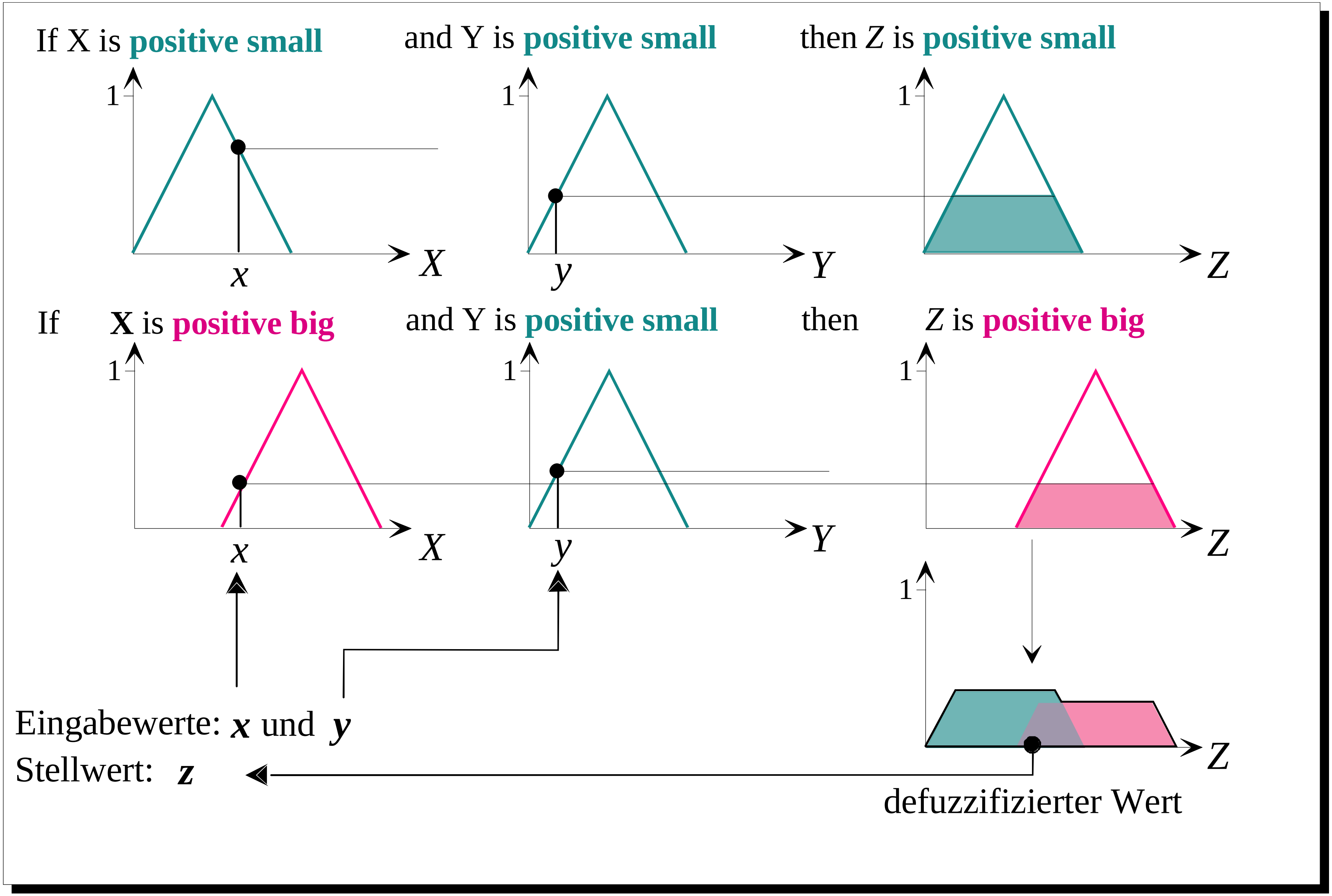
Mamdani Control



Fuzzy inference can be defined as a process of mapping from a given input to an output, using the theory of fuzzy sets.

Mamdani-style inference

Rule: 1
IF *x is A3*
OR *y is B1*
THEN z C1

Rule: 2
IF *x is A1*
AND *y is B2*
THEN z is
C2

Rule: 3
IF *x is A1*
THEN z isC3

Rule: 1
IF *projectfunding is adequate*
OR *projectjstaffing is small*
THEN *risk is low*

Rule: 2
IF *projectfunding is marginal*
AND *project_staffing is large*
THEN *risk is normal*

Rule: 3
IF *projectfunding is inadequate*
THEN *risk is high*

where *x, y and z {project funding, project staffing and risk) are linguistic variables; Al, A2 and A3 (inadequate, marginal and adequate) are linguistic values* determined by fuzzy sets on universe of discourse *X (project funding); Bl and B2 (small and large) are linguistic values determined by fuzzy sets on universe of* discourse *Y (project staffing); Cl, C2and C3 (low, normal and high) are linguistic*values determined by fuzzy sets on universe of discourse *Z (risk).*

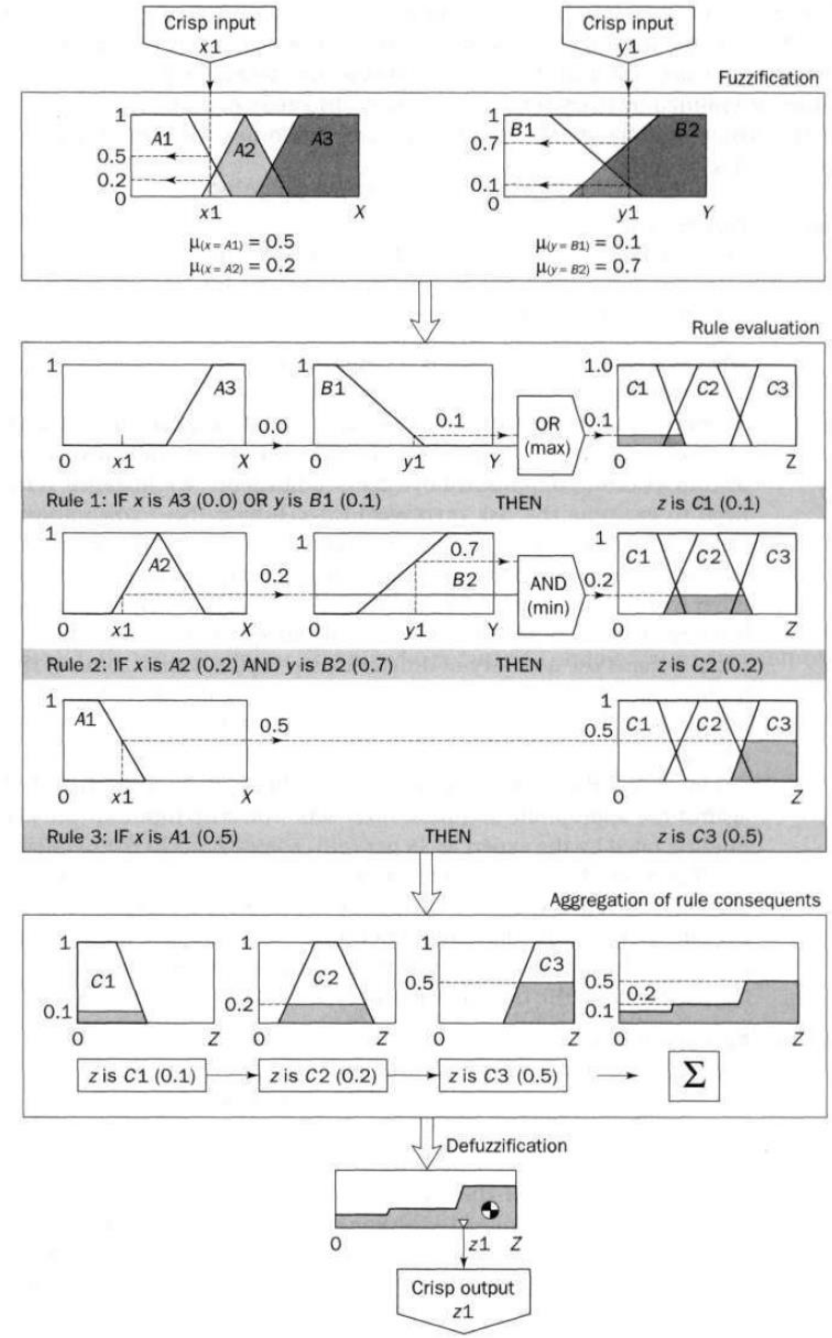


Figure 4.10 The basic structure of Mamdani-style fuzzy inference

Step 1: *Fuzzification*

Step *2: Rule evaluation*

Step 3: ***Aggregation of the rule outputs***

Step 4: ***Defuzzification***

By centroid technique. It finds the point where a vertical line would slice the aggregate set into two equal masses. Mathematically this **centre of gravity (COG) can be expressed as**

$$\mathrm{COG} = \frac{\int_a^b \mu_A(x)\,x\,dx}{\int_a^b \mu_A(x)\,dx}$$

NEURO FUZZY SYSTEMS

Neural Networks

- neural networks are low-level computational structures that perform well when dealing with raw data
- although neural networks can learn, they are opaque to the user.

Fuzzy Systems

- fuzzy logic deals with reasoning on a higher level, using linguistic information acquired from domain experts.
- fuzzy systems lack the ability to learn and cannot adjust themselves to a new environment.

Integrated neuro-fuzzy systems can combine the parallel computation and learning abilities of neural networks with the humanlike knowledge representation and explanation abilities of fuzzy systems. As a result, neural networks become more transparent, while fuzzy systems become capable of learning.

How does a neuro-fuzzy system look?

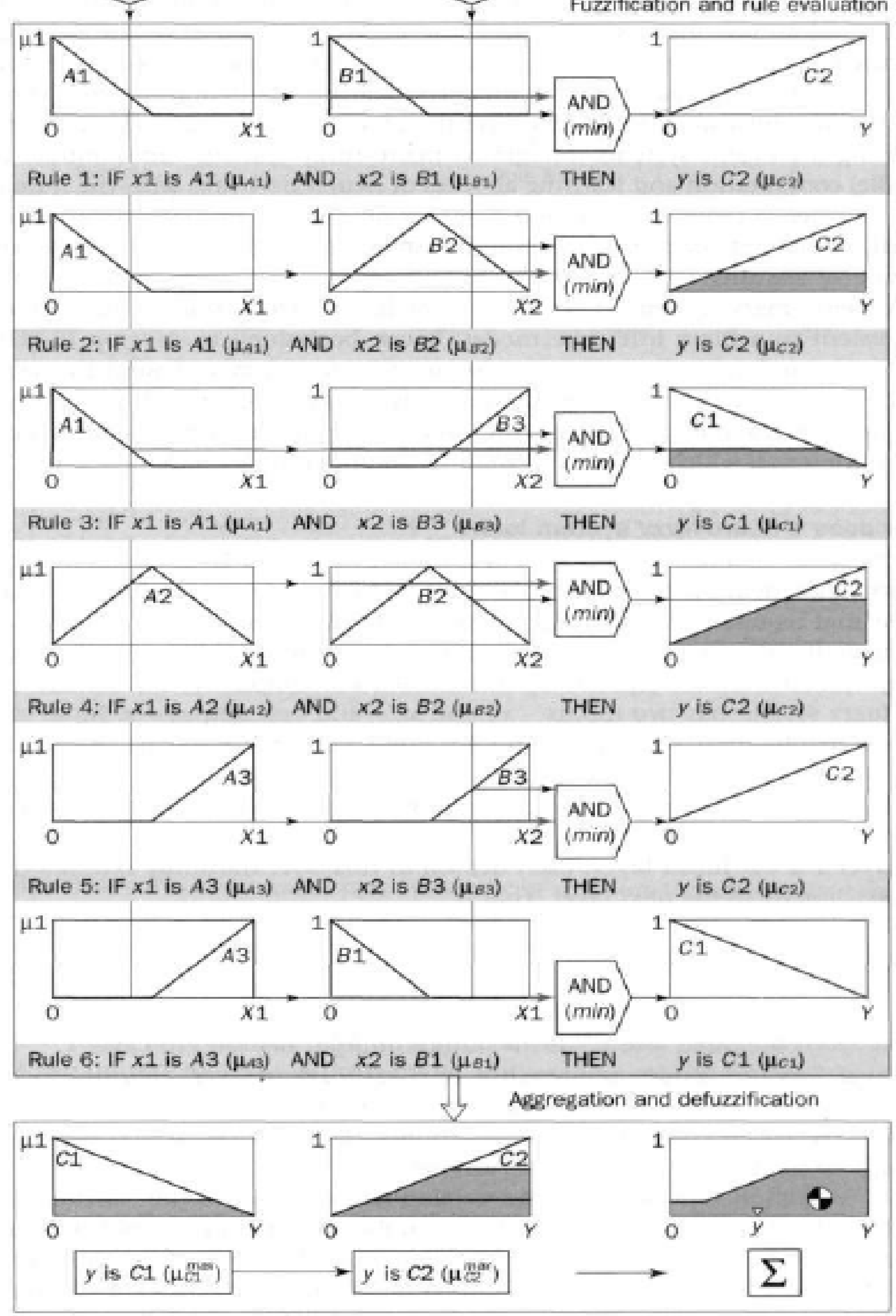


Figure 8.4 Mamdani fuzzy inference system

Mamdani fuzzy inference model, and a

neurofuzzy system that corresponds to

this model.

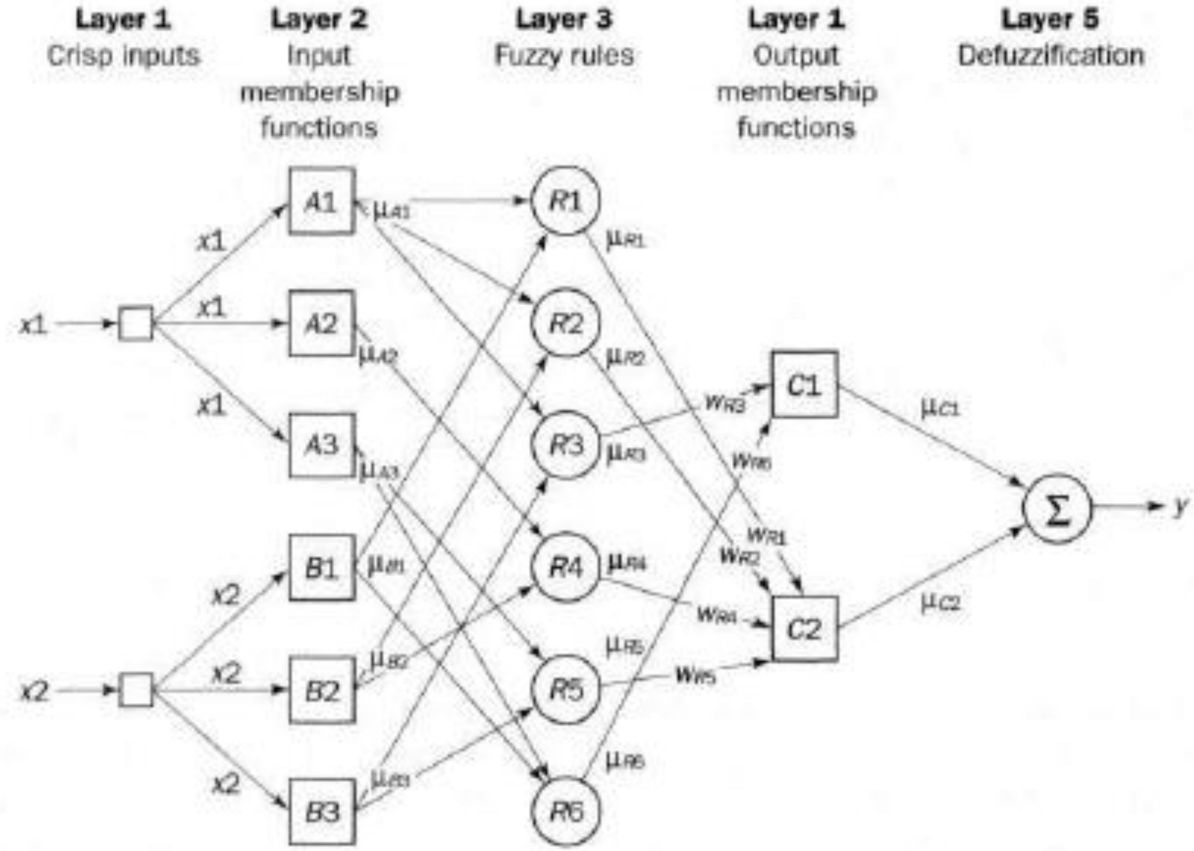


Figure 8.5 Neuro-fuzzy equivalent system

triangular membership function can be specified by two parameters $\{a, b\}$ as follows:

$$y_i^{(2)} = \begin{cases} 0, & \text{if } x_i^{(2)} \leqslant a - \dfrac{b}{2} \\ 1 - \dfrac{2|x_i^{(2)} - a|}{b}, & \text{if } a - \dfrac{b}{2} < x_i^{(2)} < a + \dfrac{b}{2} \\ 0, & \text{if } x_i^{(2)} \geqslant a + \dfrac{b}{2} \end{cases} \tag{8.4}$$

Layer 1 is the input layer

Layer 2 is the input membership or fuzzification layer.

- Neurons in this layer represent fuzzy sets used in the antecedents of fuzzy rules.
- The activation function of a membership neuron is set to the function that specifies the neuron's fuzzy set.
- A fuzzification neuron receives a crisp input and determines the degree to which this input belongs to the neuron's fuzzy set.

In this example triangular membership function. As we can see, **the output of a fuzzification neuron depends not only on its input, but also on the centre, *a, and the width, b,* of the triangular activation function. Parameters *a and b of the fuzzification neurons can play the same role in a* neuro-fuzzy system as synaptic weights in a neural network.**

Layer 3 **is the fuzzy rule layer.**

Each neuron in this layer corresponds to a single fuzzy rule. A fuzzy rule neuron receives inputs from the fuzzification neurons that represent fuzzy sets in the rule antecedents.

In a neuro-fuzzy system, intersection can be implemented by the product operator. Thus, the output of neuron *i in Layer 3* is obtained as:

$$y_i^{(3)} = x_{1i}^{(3)} \times x_{2i}^{(3)} \times \ldots \times x_{ki}^{(3)}, \tag{8.5}$$

where $x_{1i}^{(3)}, x_{2i}^{(3)}, \ldots, x_{ki}^{(3)}$ are the inputs and $y_i^{(3)}$ is the output of fuzzy rule neuron i in Layer 3. For example,

$$y_{R1}^{(3)} = \mu_{A1} \times \mu_{B1} = \mu_{R1}$$

The value of *μr1 represents the firing strength of fuzzy rule neuron R1.* The weights between Layer 3 and Layer 4 represent the **normalised degrees of confidence (known as certainty factors) of the corresponding fuzzy rules.** These weights are adjusted during training of a neuro-fuzzy system.

What is the normalised degree of confidence of a fuzzy rule?
Different rules represented in a neuro-fuzzy system may be associated with different degrees of confidence. An expert may attach the degree of confidence to each fuzzy IF-THEN rule by setting the corresponding weights within the range of [0,1]. During training, however, these weights can change. To keep them within the specified range, the weights are normalised by dividing their respective values by the highest weight magnitude obtained at each iteration.

Layer 4 is the output membership layer. Neurons in this layer represent fuzzy sets used in the consequent of fuzzy rules. An output membership neuron receives inputs from the corresponding fuzzy rule neurons and combines them by using the fuzzy operation **union. This operation can be implemented by the** probabilistic OR (also known as the algebraic sum). That is,

$$y_i^{(4)} = x_{1i}^{(4)} \oplus x_{2i}^{(4)} \oplus \ldots \oplus x_{li}^{(4)},$$

For example,

$$y_{C1}^{(4)} = \mu_{R3} \oplus \mu_{R6} = \mu_{C1}$$

The value of *μc1 represents the integrated firing strength of fuzzy rule neurons* A3 and *R6.*

Layer 5 is the defuzzification layer. Each neuron in this layer represents a single output of the neuro-fuzzy system.

- It takes the output fuzzy sets clipped by the respective integrated firing strengths and combines them into a single fuzzy set.
- The output of the neuro-fuzzy system is crisp, and thus a combined output fuzzy set must be defuzzified.

- The output of the neuro-fuzzy system is crisp, and thus a combined output fuzzy set must be defuzzified.

- Neuro-fuzzy systems can apply standard defuzzification methods, including the centroid technique.

In this example; The sum-product composition calculates the crisp output as the weighted average of the centroids of all output membership functions.The weighted average of the centroids of the clipped fuzzy sets Cl and C2 is calculated as,

$$y = \frac{\mu_{C1} \times a_{C1} \times b_{C1} + \mu_{C2} \times a_{C2} \times b_{C2}}{\mu_{C1} \times b_{C1} + \mu_{C2} \times b_{C2}},$$

How does a neuro-fuzzy system learn?

A neuro-fuzzy system is essentially a multi-layer neural network, and thus it can apply standard learning algorithms developed for neural networks, including the back-propagation algorithm (Kasabov, 1996; Lin and Lee, 1996; Nauck *et al,* 1997; Von Altrock, 1997).

- When a training input-output example is presented to the system
- The back-propagation algorithm computes the system output and compares it with the desired output of the training example.
- The difference (also called the error) is propagated backwards through the network from the output layer to the input layer.
- The neuron activation functions are modified as the error is propagated.
- To determine the necessary modifications, the backpropagation algorithm differentiates the activation functions of the neurons.

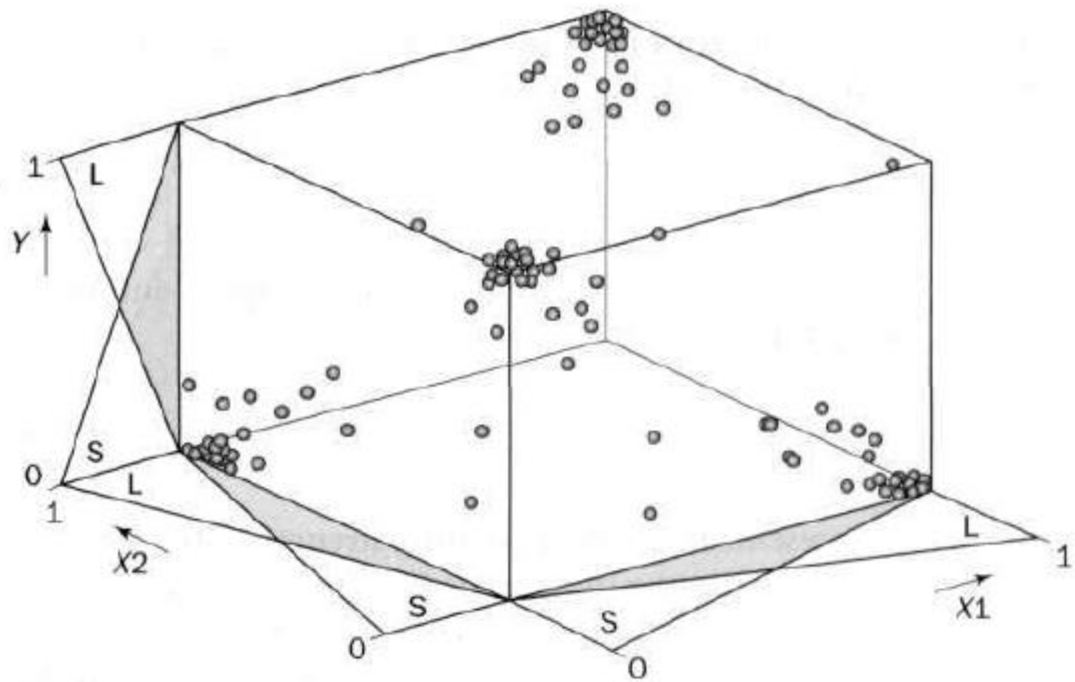
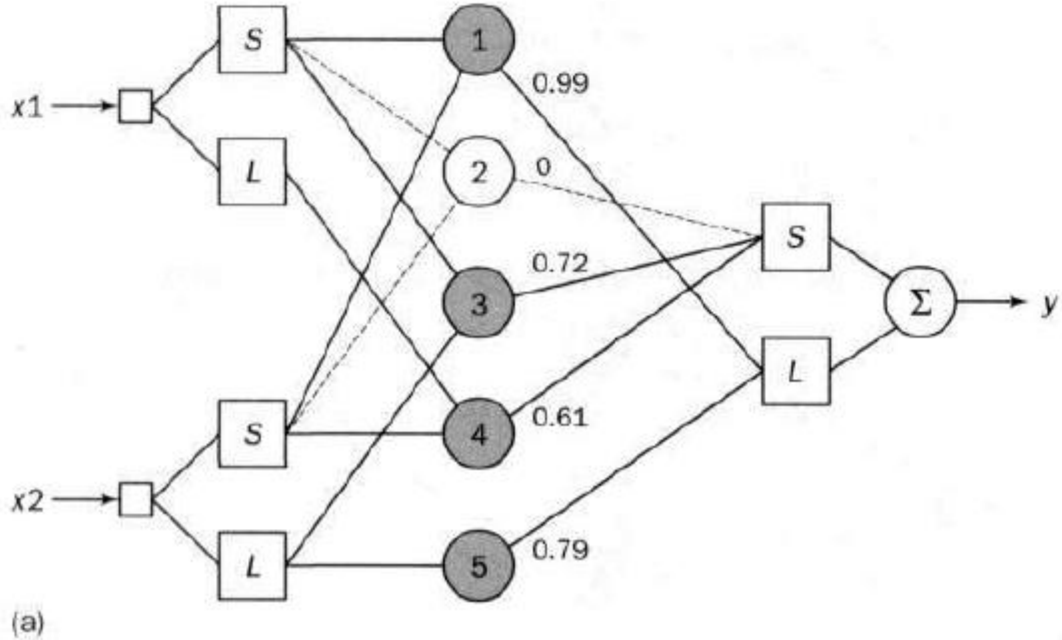


Figure 8.7 Training patterns in the three-dimensional input-output space

Distribution of 100 training patterns in the three dimensional input-output space X1 × X2 × *Y. Each training pattern here is* determined by three variables: two inputs *xl and x2, and one output y. Input* and output variables are represented by two linguistic values: *small (S) and large (L).* The data set of Figure 8.7 is used for training the five-rule neuro-fuzzy system shown in Figure 8.8(a). Suppose that fuzzy IF-THEN rules incorporated into the system structure are supplied by a domain expert.



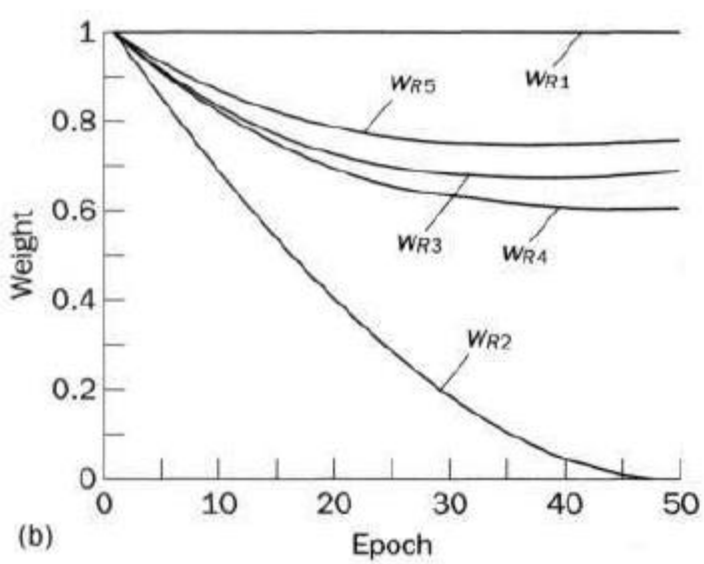


Figure 8.8 Five-rule neuro-fuzzy system for the Exclusive-OR operation: (a) five-rule system; (b) training for 50 epochs

initial weights between Layer 3 and Layer 4 are set to unity.
During training the neuro-fuzzy system uses the back-propagation algorithm to adjust the weights and to modify input and output membership functions.

On top of that, we cannot be sure that the 'expert' has not left out a few rules.

What can we do to reduce our dependence on the expert knowledge?

Can a neuro-fuzzy system extract rules directly from numerical data?

Given input and output linguistic values, a neuro-fuzzy system can automatically generate a complete set of fuzzy IF-THEN rules.

An example demonstrates that a neuro-fuzzy system extract fuzzy rules directly from numerical data:

Because expert knowledge is not embodied in the system this time, we set all initial weights between Layer 3 and Layer 4 to 0.5. After training we can eliminate all rules whose certainty factors are less than some sufficiently small number, say 0.1. As a result, we obtain the same set of four fuzzy IF-THEN rules

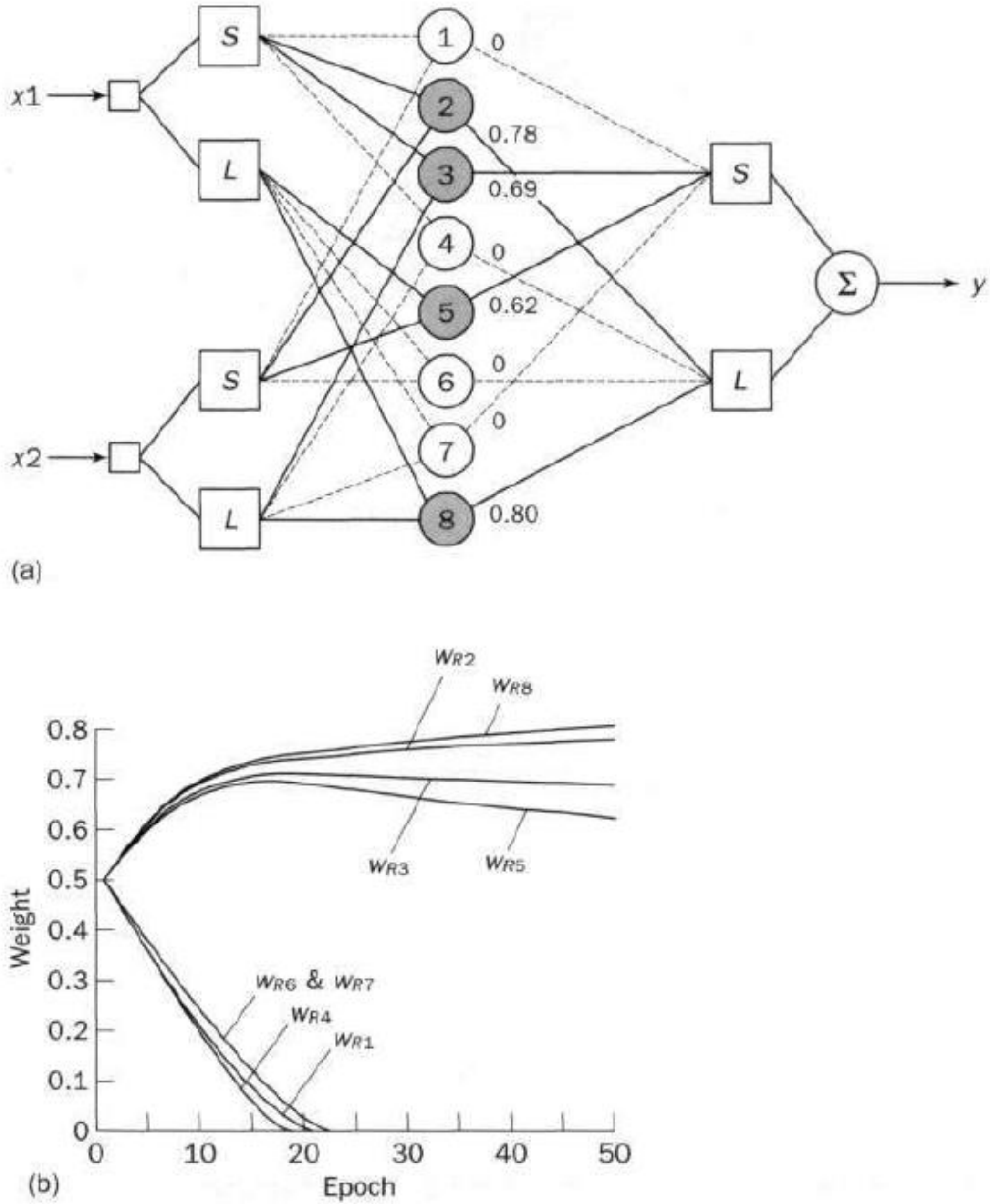


Figure 8.9 Eight-rule neuro-fuzzy system for the Exclusive-OR operation: (a) eight-rule system; (b) training for 50 epochs

The NEFCLASS Model

The main goal of NEFCLASS is to create a readable classifier that also provides an acceptable accuracy.

An interpretable fuzzy system should display the following features:

- few meaningful rules with few variables in their antecedents,
- few meaningful sets for each variable,
- there are no rule weights,
- identical linguistic terms are represented by identical fuzzy sets,
- only normal fuzzy sets are used, or even better fuzzy numbers or fuzzy intervals.