Fuzzy membership function

Generating membership function

There are possibly more ways to assign membership values or functions to fuzzy variables than there are to assign probability density functions to random variables

1. Intuition
2. Inference
3. Rank ordering
4. Neural networks
5. Genetic algorithms
6. Inductive reasoning

Intuition

- This method needs little or no introduction.
- It is simply derived from the capacity of humans to develop membership functions through their own innate intelligence and understanding.
- Consider the membership functions for the fuzzy variable temperature.
- Each curve is a membership function corresponding to various fuzzy variables, such as very cold, cold, normal, hot, and very hot. Of course, these curves are a function of context and the analyst developing them.
- For example, if the temperatures are referred to the range of human comfort we get one set of curves, and if they are referred to the range of safe operating temperatures for a steam turbine we get another set.

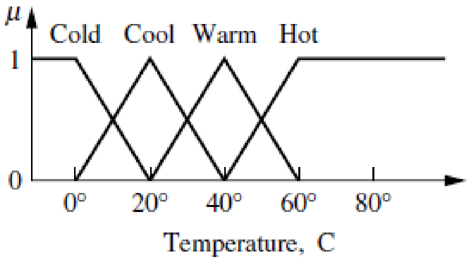


FIGURE 6.1
Membership functions for the fuzzy variable ''temperature.''

Inference

• In this method some initial knowledge to perform deductive reasoning .

•Ie. Derive or infer the conclusion given a body of facts or knowledge

•Eg: Angles of triangle (Right angled, a>b>c, a+b+c=180 etc)

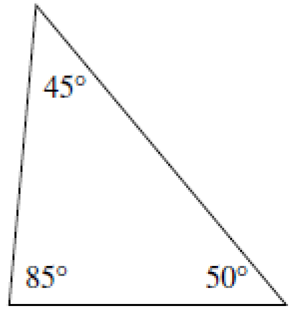


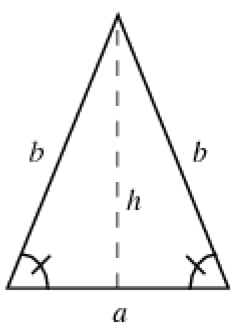
FIGURE 6.2
A specific triangle.

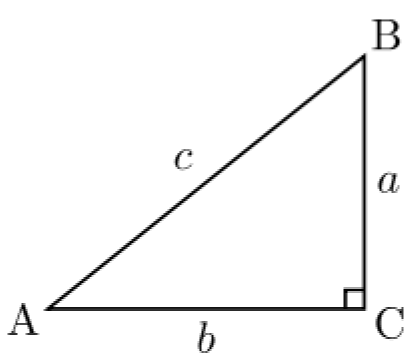
Inference

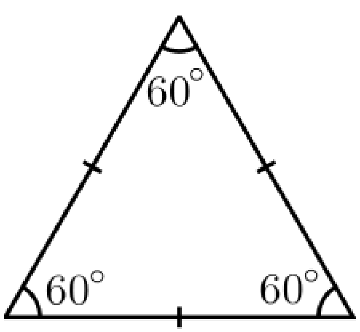
In the identification of a triangle, let A, B, and C be the inner angles of a triangle, in the order $A \geq B \geq C \geq 0$, and let U be the universe of triangles, i.e.,

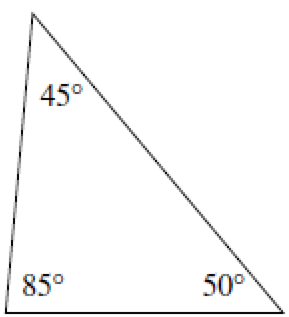
$$\mathrm{U} = \{(A, B, C) \mid A \geq B \geq C \geq 0;\ A + B + C = 180^{\circ}\} \tag{6.1}$$

$\underset{\sim}{\mathrm{I}}$	Approximate isosceles triangle
$\underset{\sim}{\mathrm{R}}$	Approximate right triangle
$\underset{\sim}{\mathrm{IR}}$	Approximate isosceles *and* right triangle
$\underset{\sim}{\mathrm{E}}$	Approximate equilateral triangle
$\underset{\sim}{\mathrm{T}}$	Other triangles







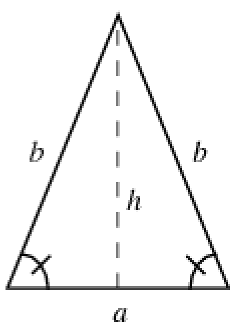


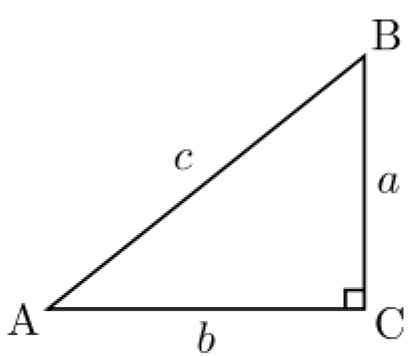
Inference

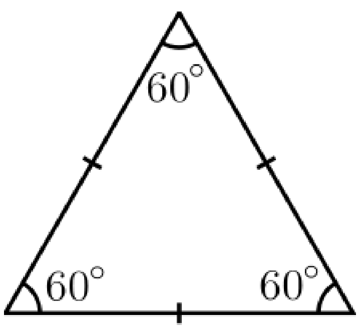
In the identification of a triangle, let A, B, and C be the inner angles of a triangle, in the order $A \geq B \geq C \geq 0$, and let U be the universe of triangles, i.e.,

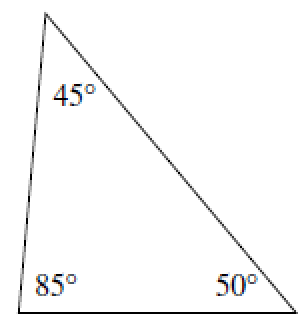
$$\mathrm{U} = \{(A, B, C) \mid A \geq B \geq C \geq 0;\ A + B + C = 180^\circ\} \qquad (6.1)$$

We can infer membership values for all of these triangle types through the method of inference, because we possess knowledge about geometry that helps us to make the membership assignments. So we shall list this knowledge here to develop an algorithm to assist us in making these membership assignments for any collection of angles meeting the constraints of Eq. (6.1).







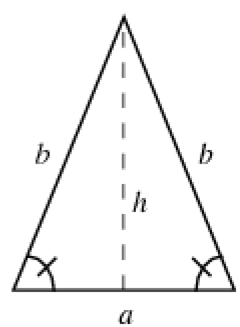


Inference

For the approximate isosceles triangle we have the following algorithm for the membership, again for the situation of $A \geq B \geq C \geq 0$ and $A + B + C = 180^\circ$:

$$\mu_{\underset{\sim}{I}}(A, B, C) = 1 - \frac{1}{60^\circ} \min(A - B, B - C) \qquad (6.2)$$

So, for example, if $A = B$ or $B = C$, the membership value in the approximate isosceles triangle is $\mu_{\underset{\sim}{I}} = 1$; if $A = 120^\circ$, $B = 60^\circ$, and $C = 0^\circ$, then $\mu_{\underset{\sim}{I}} = 0$.

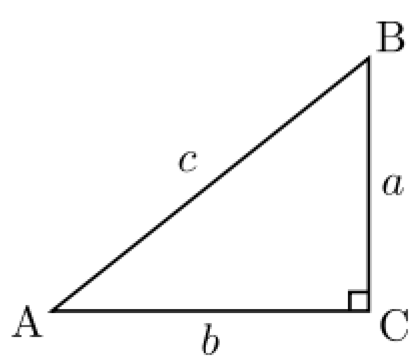


Inference

For a fuzzy right triangle,

$$\mu_{\underset{\sim}{R}}(A, B, C) = 1 - \frac{1}{90^\circ} |A - 90^\circ| \tag{6.3}$$

For instance, when $A = 90^\circ$, the membership value in the fuzzy right triangle, $\mu_{\underset{\sim}{R}} = 1$, or when $A = 180^\circ$, this membership vanishes, i.e., $\mu_{\underset{\sim}{R}} = 0$.



Inference

For the case of an approximate isosceles *and* right triangle (there is only one of these in the crisp domain), we can find this membership function by taking the logical intersection (*and* operator) of the isosceles and right triangle membership functions, or

$$\underset{\sim}{\mathrm{IR}} = \underset{\sim}{\mathrm{I}} \cap \underset{\sim}{\mathrm{R}}$$

which results in

$$\begin{aligned} \mu_{\underset{\sim}{\mathrm{IR}}}(A, B, C) &= \min[\mu_{\underset{\sim}{\mathrm{I}}}(A, B, C), \mu_{\underset{\sim}{\mathrm{R}}}(A, B, C)] \\ &= 1 - \max\left[\frac{1}{60^\circ}\min(A - B, B - C), \frac{1}{90^\circ}\left|A - 90^\circ\right|\right] \end{aligned} \tag{6.4}$$

Inference

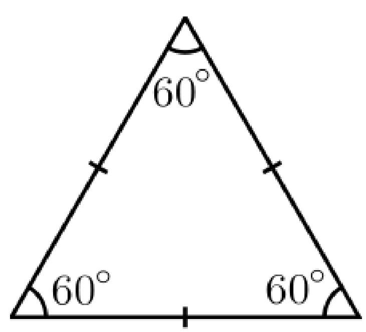
For the case of a fuzzy equilateral triangle, the membership function is given by

$$\mu_{\underset{\sim}{E}}(A, B, C) = 1 - \frac{1}{180^\circ}(A - C) \tag{6.5}$$

For example, when $A = B = C$, the membership value is $\mu_{\underset{\sim}{E}}(A, B, C) = 1$; when $A = 180^\circ$, the membership value vanishes, or $\mu_{\underset{\sim}{E}} = 0$. Finally, for the set of ''all other triangles'' (all triangular shapes other than $\underset{\sim}{I}$, $\underset{\sim}{R}$, and $\underset{\sim}{E}$) we simply invoke the complement of the logical union of the three previous cases (or, from De Morgan's principles (Eq. (2.13)), the intersection of the complements of the triangular shapes),

$$\underset{\sim}{T} = \left(\overline{\underset{\sim}{I} \cup \underset{\sim}{R} \cup \underset{\sim}{E}}\right) = \overline{\underset{\sim}{I}} \cap \overline{\underset{\sim}{R}} \cap \overline{\underset{\sim}{E}}$$

$$\begin{aligned}\mu_{\underset{\sim}{T}}(A, B, C) &= \min\{1 - \mu_{\underset{\sim}{I}}(A, B, C), 1 - \mu_{\underset{\sim}{E}}(A, B, C), 1 - \mu_{\underset{\sim}{R}}(A, B, C)\} \\ &= \frac{1}{180^\circ}\min\{3(A - B), 3(B - C), 2|A - 90^\circ|, A - C\} \end{aligned} \tag{6.6}$$



Inference

Example 6.1 [Ross, 1995]. Define a specific triangle, as shown in Fig. 6.2, with these three ordered angles:

$$\{X : A = 85^\circ \geq B = 50^\circ \geq C = 45^\circ, \text{where} A + B + C = 180^\circ\}$$

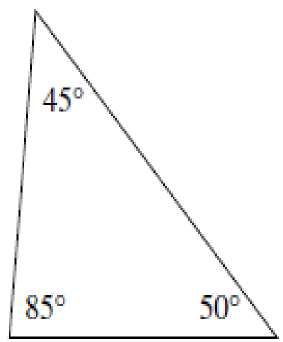


FIGURE 6.2
A specific triangle.

Inference

Example 6.1 [Ross, 1995]. Define a specific triangle, as shown in Fig. 6.2, with these three ordered angles:

$$\{X : A = 85^\circ \geq B = 50^\circ \geq C = 45^\circ, \text{ where } A + B + C = 180^\circ\}$$

The membership values for the fuzzy triangle shown in Fig. 6.2 for each of the fuzzy triangles types are determined from Eqs. (6.2)–(6.6), as listed here:

$$\mu_{\underset{\sim}{R}}(x) = 0.94$$

$$\mu_{\underset{\sim}{I}}(x) = 0.916$$

$$\mu_{\underset{\sim}{IR}}(x) = 0.916$$

$$\mu_{\underset{\sim}{E}}(x) = 0.7$$

$$\mu_{\underset{\sim}{T}}(x) = 0.05$$

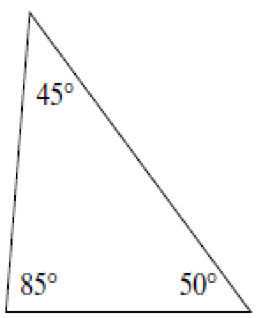


FIGURE 6.2
A specific triangle.

Hence, it appears that the triangle given in Fig. 6.2 has the highest membership in the set of fuzzy right triangles, i.e., in $\underset{\sim}{R}$. Notice, however, that the triangle in Fig. 6.2 also has high membership in the isosceles triangle fuzzy set, and reasonably high membership in the equilateral fuzzy triangle set.

Rank ordering

Assessing preferences by a single individual, a committee, a poll, and other opinion methods can be used to assign membership values to a fuzzy variable.

TABLE 6.1
Example in Rank Ordering

	Number who preferred							
	Red	**Orange**	**Yellow**	**Green**	**Blue**	**Total**	**Percentage**	**Rank order**
Red	–	517	525	545	661	2248	22.5	2
Orange	483	–	841	477	576	2377	23.8	1
Yellow	475	159	–	534	614	1782	17.8	4
Green	455	523	466	–	643	2087	20.9	3
Blue	339	424	386	357	–	1506	15	5
Total						10,000		

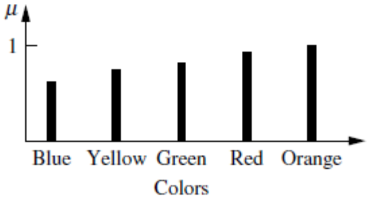


FIGURE 6.3
Membership function for best color.

Neural network

A neural network is a technique that seeks to build an intelligent program (to implement intelligence) using models that simulate the working network of the neurons in the human brain [Yamakawa, 1992; Hopfield, 1982; Hopfield and Tank, 1986].

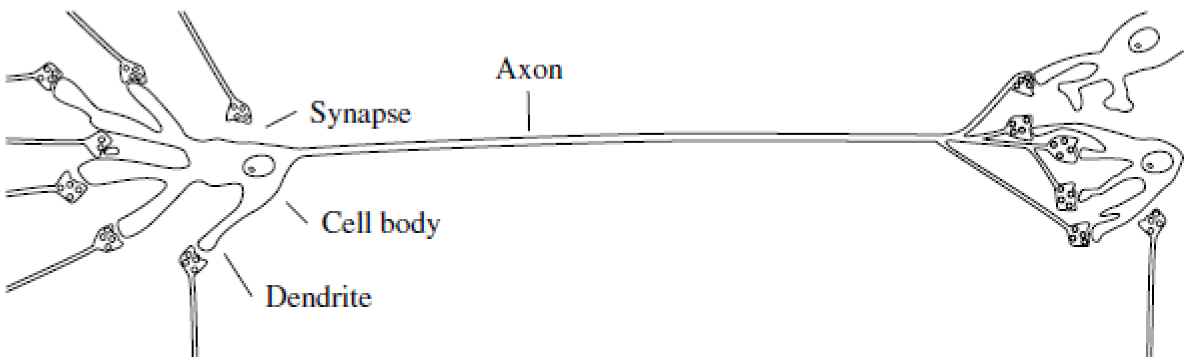


FIGURE 6.4
A simple schematic of a human neuron.

Neural network

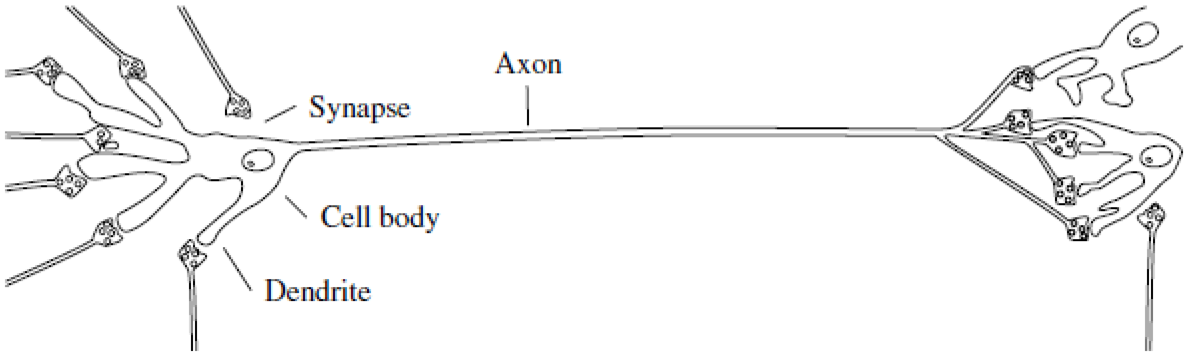


FIGURE 6.4
A simple schematic of a human neuron.

- A neuron is made up of several protrusions called dendrites and a long branch called the axon.
- A neuron is joined to other neurons through the dendrites.
- The dendrites of different neurons meet to form synapses, the areas where messages pass.
- The neurons receive the impulses via the synapses.
- If the total of the impulses received exceeds a certain threshold value, then the neuron sends an impulse down the axon where the axon is connected to other neurons through more synapses.

Neural network

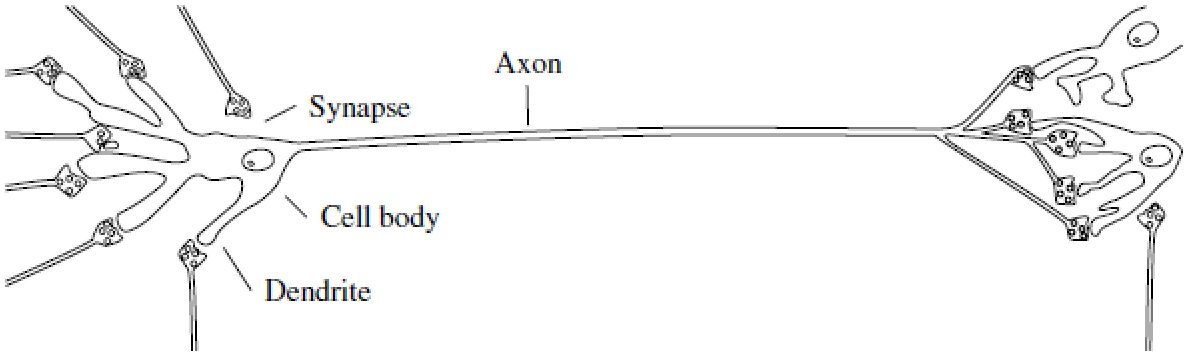


FIGURE 6.4
A simple schematic of a human neuron.

- The synapses may be excitatory or inhibitory in nature.
  - An excitatory synapse adds to the total of the impulses reaching the neuron, whereas an inhibitory neuron reduces the total of the impulses reaching the neuron.
- In a global sense, a neuron receives a set of input pulses and sends out another pulse that is a function of the input pulses.

Neural network

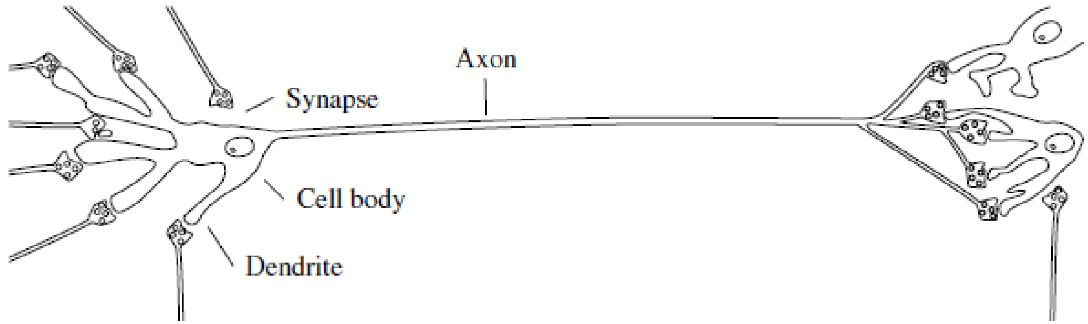


FIGURE 6.4
A simple schematic of a human neuron.

This concept of how neurons work in the human brain is utilized in performing computations on computers. Researchers have long felt that the neurons are responsible for the human capacity to learn, and it is in this sense that the physical structure is being emulated by a neural network to accomplish machine learning. Each computational unit computes some function of its inputs and passes the result to connected units in the network. The knowledge of the system comes out of the entire network of the neurons.

Neural network

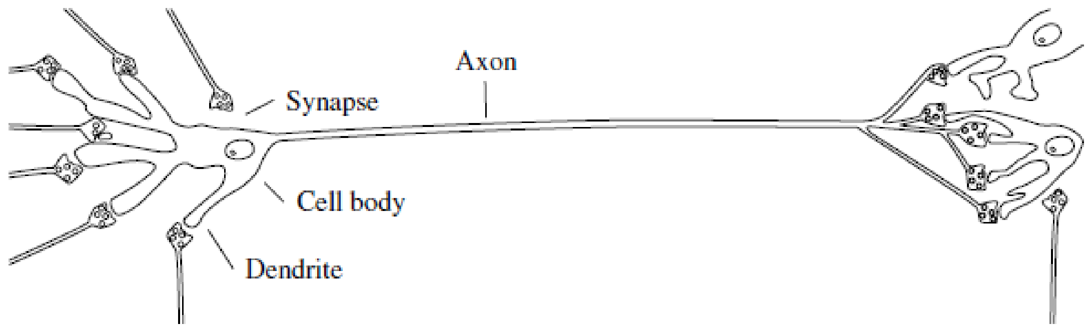


FIGURE 6.4
A simple schematic of a human neuron.

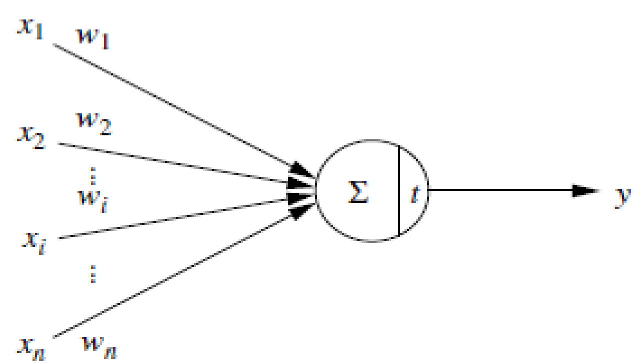


FIGURE 6.5
A threshold element as an analog to a neuron.

Neural network

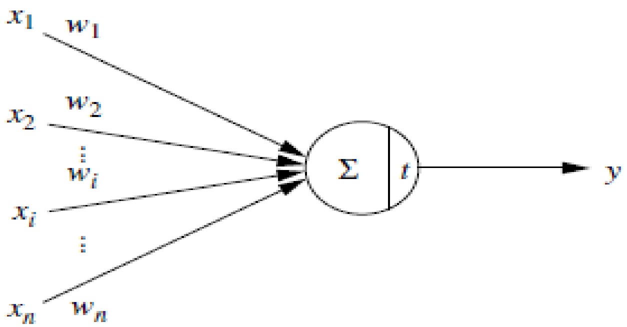


FIGURE 6.5
A threshold element as an analog to a neuron.

The variables *x1, x2,. . . , xi ,. . . , xn are the n inputs to the threshold element.*
The variables *w1, w2,. . . , wi ,. . . , wn are the weights associated with the impulses/inputs, signifying the relative importance* that is associated with the path from which the input is coming.
When *wi is positive, input xi acts as an excitatory signal for the element. When wi is negative, input xi acts as an* inhibitory signal for the element.

Neural network

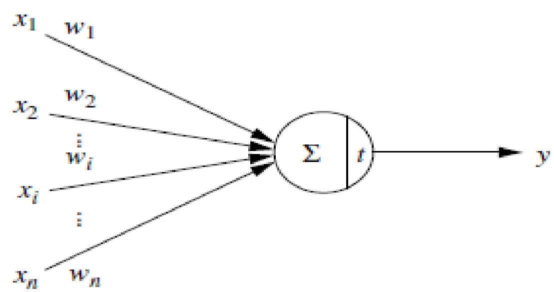


FIGURE 6.5
A threshold element as an analog to a neuron.

The threshold element sums the product of these inputs and their associated weights ($\sum w_i x_i$), compares it to a prescribed threshold value, and, if the summation is greater than the threshold value, computes an output using a nonlinear function (F). The signal output y (Fig. 6.5) is a nonlinear function (F) of the difference between the preceding computed summation and the threshold value and is expressed as

$$y = F\left(\sum w_i x_i - t\right) \tag{6.7}$$

where
- x_i signal input ($i = 1, 2, \ldots, n$)
- w_i weight associated with the signal input x_i
- t threshold level prescribed by user
- $F(s)$ is a nonlinear function, e.g., a sigmoid function $F(s) = \dfrac{1}{1 + e^{-s}}$

Neural network

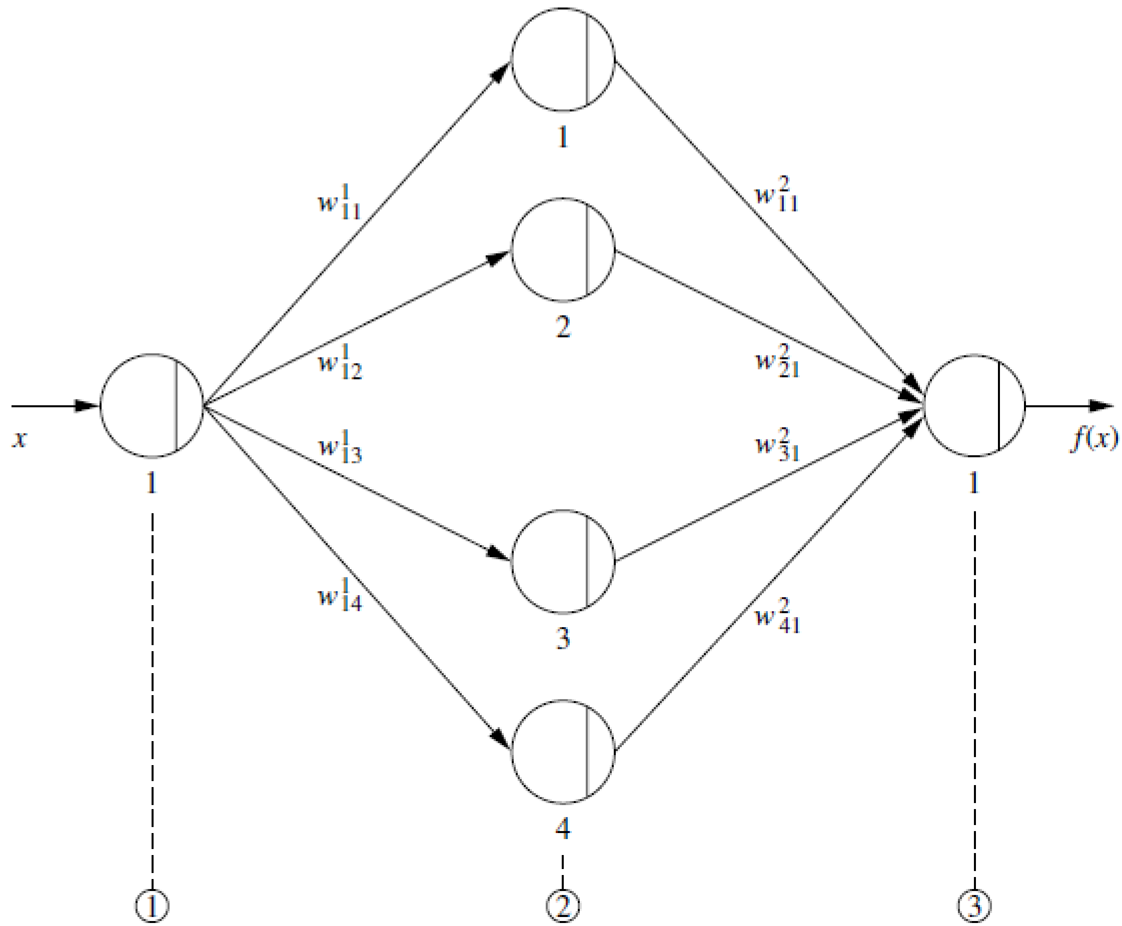


FIGURE 6.6
A simple $1 \times 4 \times 1$ neural network, where w^i_{jk} represents the weight associated with the path connecting the jth element of the ith layer to the kth element of the $(i+1)$th layer.

Neural network

$$y = F\left(\sum w_i x_i - t\right)$$

where x_i signal input $(i = 1, 2, \ldots, n)$
w_i weight associated with the signal input x_i
t threshold level prescribed by user
$F(s)$ is a nonlinear function, e.g., a sigmoid function $F(s) = \dfrac{1}{1 + e^{-s}}$

$$E = f(x)_{\text{actual}} - f(x)_{\text{output}}$$

Neural network

The error measure associated with the different elements in the hidden layers is computed as follows. Let E_j be the error associated with the jth element (Fig. 6.7). Let w_{nj} be the weight associated with the line from element n to element j and let I be the input to unit n. The error for element n is computed as

$$E_n = F'(I) w_{nj} E_j \tag{6.9}$$

where, for $F(I) = 1/(1 + e^{-I})$, the sigmoid function, we have

$$F'(I) = F(I)(1 - F(I)) \tag{6.10}$$

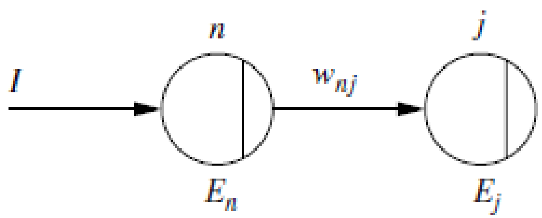


FIGURE 6.7
Distribution of error to different elements.

Neural network

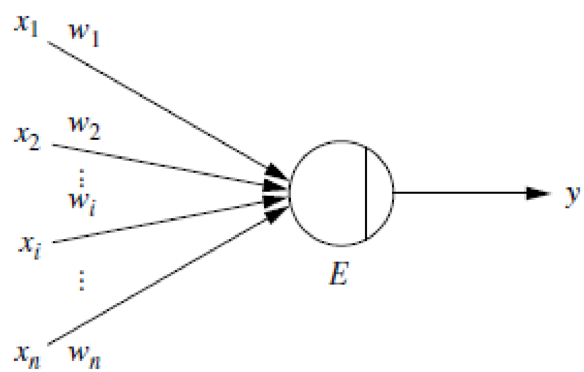


FIGURE 6.8
A threshold element with an error E associated with it.

Next the different weights w_{jk}^{i} connecting different elements in the network are corrected so that they can approximate the final output more closely. For updating the weights, the error measure on the elements is used to update the weights on the lines joining the elements.

For an element with an error E associated with it, as shown in Fig. 6.8, the associated weights may be updated as

$$w_i \text{ (new)} = w_i \text{ (old)} + \alpha E x_i \tag{6.11}$$

where α = learning constant
E = associated error measure
x_i = input to the element

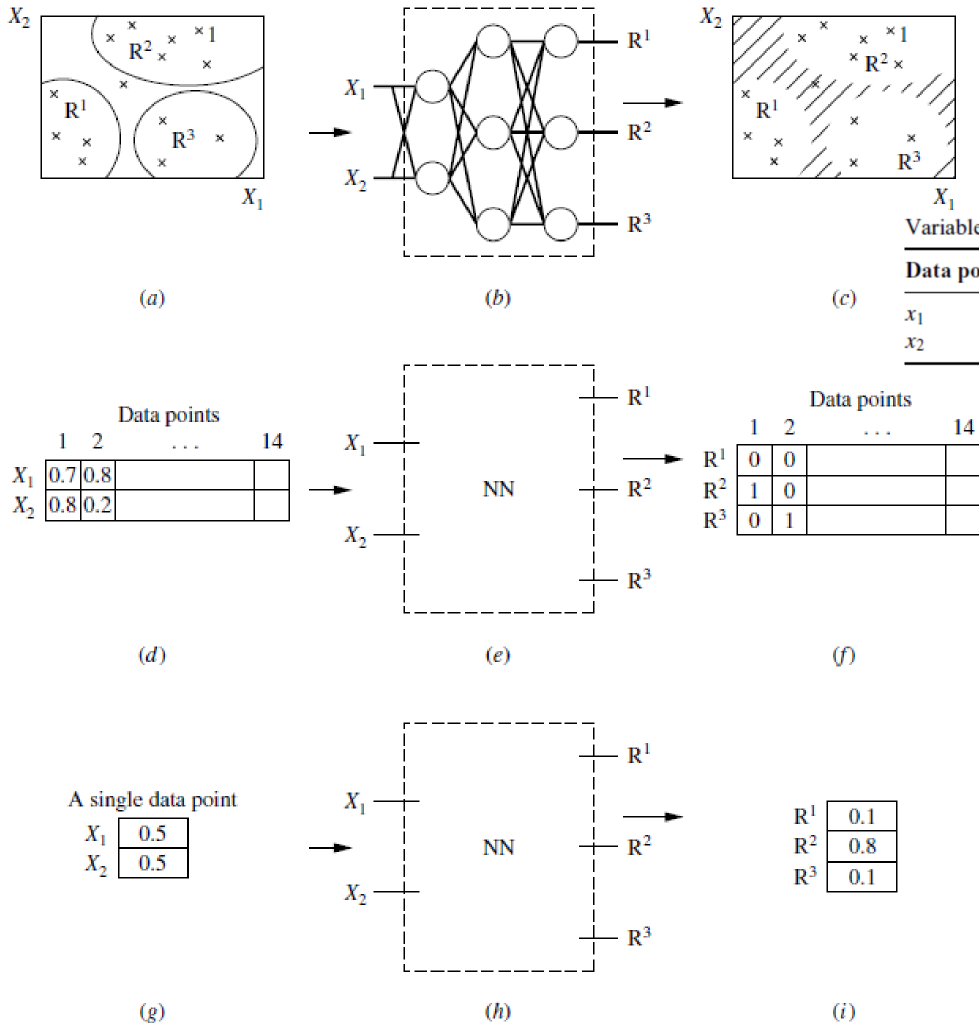
Neural network

The input value x_i is passed through the neural network (now having the updated weights) again, and the errors, if any, are computed again. This technique is iterated until the error value of the final output is within some user-prescribed limits.

The neural network then uses the next set of input–output data. This method is continued for all data in the training data set. This technique makes the neural network simulate the nonlinear relation between the input–output data sets. Finally a checking data set is used to verify how well the neural network can simulate the nonlinear relationship.

For systems where we may have data sets of inputs and corresponding outputs, and where the relationship between the input and output may be highly nonlinear or not known at all, we may want to use fuzzy logic to classify the input and the output data sets broadly into different fuzzy classes. Furthermore, for systems that are dynamic in nature (the system parameters may change in a nondeterministic fashion) the fuzzy membership functions would have to be repeatedly updated. For these types of systems it is advantageous to use a neural network since the network can modify itself (by changing the weight assignments in the neural network)

Neural network



Variables describing the data points to be used as a training data set

Data point	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
x_1	0.05	0.09	0.12	0.15	0.20	0.75	0.80	0.82	0.90	0.95
x_2	0.02	0.11	0.20	0.22	0.25	0.75	0.83	0.80	0.89	0.89

FIGURE 6.9
Using a neural network to determine membership functions [Takagi and Hayashi, 1991].

Neural network

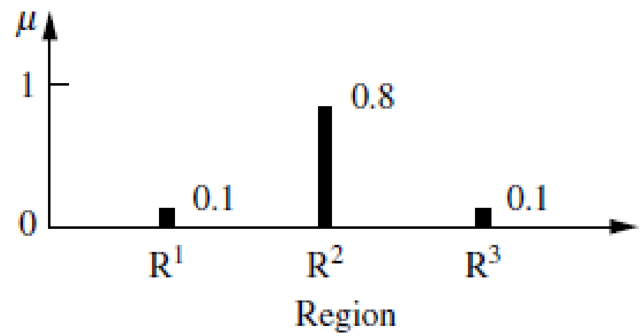


FIGURE 6.10
Membership function for data point $(X_1, X_2 = (0.5, 0.5))$.

Genetic algorithm

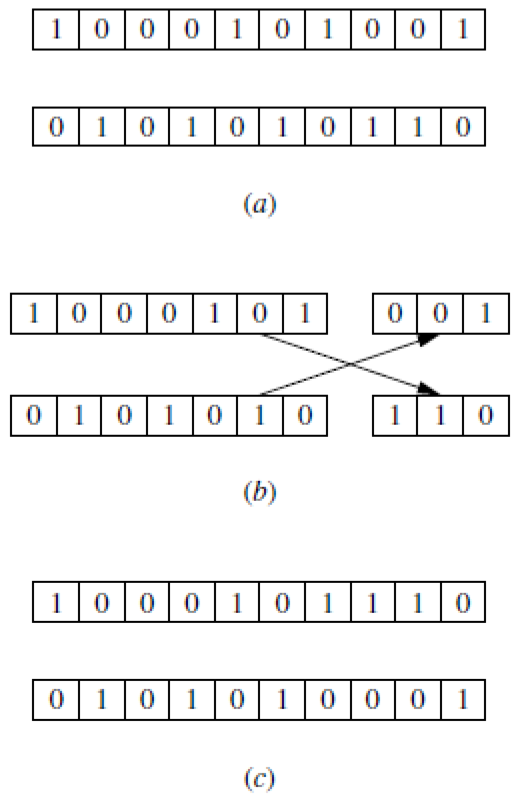


FIGURE 6.12
Crossover in strings. (*a*) Two strings are selected at random to be mated; (*b*) a random location in the strings is located (here the location is before the last three bit locations); and (*c*) the string portions following the selected location are exchanged.

Genetic algorithm

TABLE 6.8
Data for a single-input, single-output system

x	1	2	3	4	5
y	1	4	9	16	25

TABLE 6.9
Functional mapping for the system

x	S	L
y	S	VL

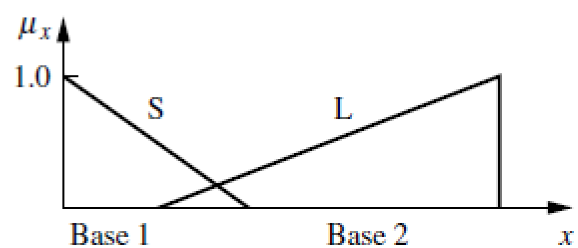


FIGURE 6.13
Membership functions for the input variables are assumed to be right triangles.

Genetic algorithm

$$C_i = C_{\min} + \frac{b}{2^L - 1}\left(C_{\max_i} - C_{\min_i}\right)$$

TABLE 6.10*a*

First iteration using a genetic algorithm for determining optimal membership functions

String number	(1) String	(2) base 1 (bin)	(3) base 2 (bin)	(4) base 3 (bin)	(5) base 4 (bin)	(6) base 1	(7) base 2	(8) base 3	(9) base 4	(10) y' $(x = 1)$	(11) y' $(x = 2)$	(12) y' $(x = 3)$	(13) y' $(x = 4)$	(14) y' $(x = 5)$	(15) 1000− $\sum(y_i - y_i')^2$	(16) Expected count = f/f_{av}	(17) Actual count
1	000111 010100 010110 110011	7	20	22	51	0.56	1.59	8.73	20.24	0	0	0	12.25	25	887.94	1.24	1
2	010010 001100 101100 100110	18	12	44	38	1.43	0.95	17.46	15.08	12.22	0	0	0	25	521.11	0.73	0
3	010101 101010 001101 101000	21	42	13	40	1.67	3.33	5.16	15.87	3.1	10.72	15.48	20.24	25	890.46	1.25	2
4	100100 001001 101100 100011	36	9	44	35	2.86	0.71	17.46	13.89	6.98	12.22	0	0	25	559.67	0.78	1
														Sum	2859.18		
														Average	714.80		
														Maximum	890.46		

TABLE 6.10*b*

Second iteration using a genetic algorithm for determining optimal membership functions

(1) Selected strings	(2) New Strings	(3) base 1 (bin)	(4) base 2 (bin)	(5) base 3 (bin)	(6) base 4 (bin)	(7) base 1	(8) base 2	(9) base 3	(10) base 4	(11) y' $(x = 1)$	(12) y' $(x = 2)$	(13) y' $(x = 3)$	(14) y' $(x = 4)$	(15) y' $(x = 5)$	(16) 1000− $\sum(y_i - y_i')^2$	(17) Expected count = f/f_{av}	(18) Actual count
000111 0101|00 010110 110011	000111 010110 001101 101000	7	22	13	40	0.56	1.75	5.16	15.87	0	0	0	15.93	25	902.00	1.10	1
010101 1010|10 001101 101000	010101 101000 010110 110011	21	40	22	51	1.67	3.17	8.73	20.24	5.24	5.85	12.23	18.62	25	961.30	1.18	2
010101 101010 001101 1|01000	010101 101010 001101 100011	21	42	13	35	1.67	3.33	5.16	13.89	3.1	12.51	16.68	20.84	25	840.78	1.03	1
100100 001001 101100 1|00011	100100 001001 101100 101000	36	9	44	40	2.86	0.71	17.46	15.87	6.11	12.22	0	0	25	569.32	0.70	0
														Sum	3273.40		
														Average	818.35		
														Maximum	961.30		

Genetic algorithm

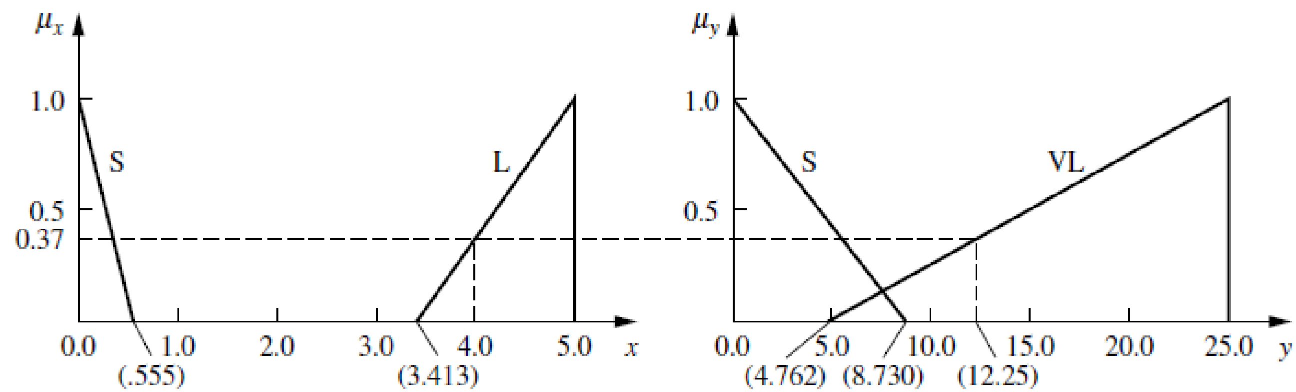


FIGURE 6.14
Physical representation of the first string in Table 4.12a and the graphical determination of y for a given x.

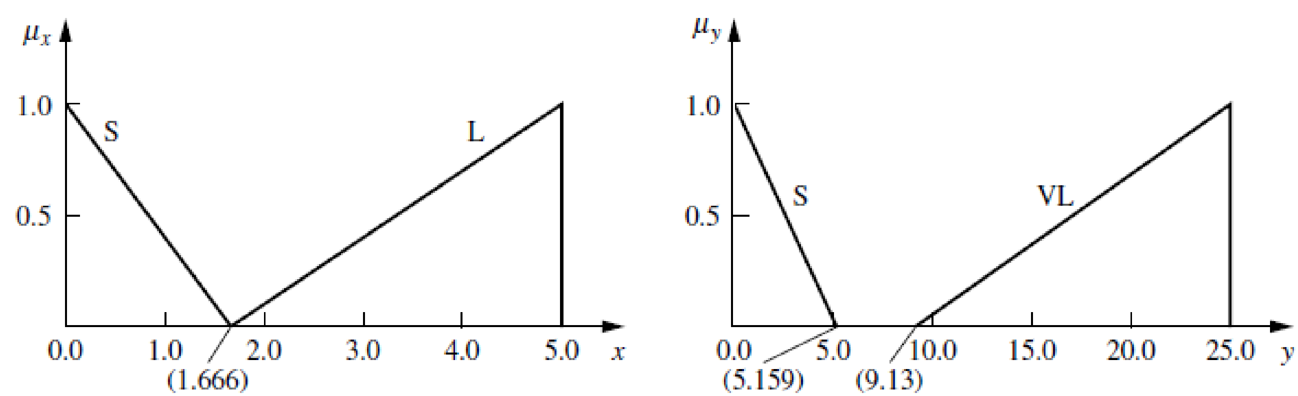


FIGURE 6.15
Physical mapping of the best string in the first generation of strings in the genetic algorithm.

Inductive reasoning

The induced rule is that rule consistent with all available information of which the entropy is minimum.

For a simple one-dimensional (one uncertain variable) case, let us assume that the probability of the ith sample w_i to be true is $\{p(w_i)\}$. If we actually observe the sample w_i in the future and discover that it is true, then we gain the following information, $I(w_i)$:

$$I(w_i) = -k \ln p(w_i) \tag{6.16}$$

where k is a normalizing parameter. If we discover that it is false, we still gain this information:

$$I(\overline{w}_i) = -k \ln[1 - p(w_i)] \tag{6.17}$$

Then the entropy of the inner product of all the samples (N) is

$$S = -k \sum_{i=1}^{N} [p_i \ln p_i + (1 - p_i) \ln(1 - p_i)] \tag{6.18}$$

where $p_i = p(w_i)$. The minus sign before parameter k in Eq. (6.18) ensures that $S \geq 0$, because $\ln x \leq 0$ for $0 \leq x \leq 1$.

Inductive reasoning

Then the entropy of the inner product of all the samples (N) is

$$S = -k \sum_{i=1}^{N} [p_i \ln p_i + (1 - p_i) \ln(1 - p_i)] \tag{6.18}$$

where $p_i = p(w_i)$. The minus sign before parameter k in Eq. (6.18) ensures that $S \geq 0$, because $\ln x \leq 0$ for $0 \leq x \leq 1$.

The third law of induction, which is typical in pattern classification, says that the entropy of a rule should be minimized. Minimum entropy (S) is associated with all the p_i being as close to ones or zeros as possible, which in turn implies that they have a very high probability of either happening or not happening, respectively. Note in Eq. (6.18) that if $p_i = 1$ then $S = 0$. This result makes sense since p_i is the probability measure of whether a value belongs to a partition or not.

Inductive reasoning

To subdivide our data set into membership functions we need some procedure to establish fuzzy thresholds between classes of data. We can determine a threshold line with an entropy minimization screening method, then start the segmentation process, first into two classes. By partitioning the first two classes one more time, we can have three different classes. Therefore, a repeated partitioning with threshold value calculations will allow us to partition the data set into a number of classes, or fuzzy sets, depending on the shape used to describe membership in each set.

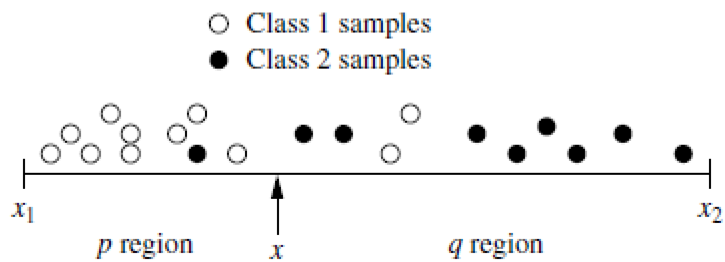


FIGURE 6.17
Illustration of threshold value idea.

Inductive reasoning

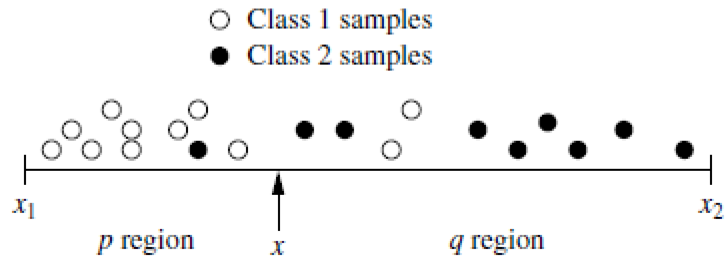


FIGURE 6.17
Illustration of threshold value idea.

An entropy with each value of x in the region x_1 and x_2 is expressed by Christensen [1980] as

$$S(x) = p(x)S_p(x) + q(x)S_q(x) \tag{6.19}$$

where

$$S_p(x) = -[p_1(x)\ln p_1(x) + p_2(x)\ln p_2(x)] \tag{6.20}$$

$$S_q(x) = -[q_1(x)\ln q_1(x) + q_2(x)\ln q_2(x)] \tag{6.21}$$

where $p_k(x)$ and $q_k(x)$ = conditional probabilities that the class k sample is in the region $[x_1, x_1 + x]$ and $[x_1 + x, x_2]$, respectively

$p(x)$ and $q(x)$ = probabilities that all samples are in the region $[x_1, x_1 + x]$ and $[x_1 + x, x_2]$, respectively

$p(x) + q(x) = 1$

Inductive reasoning

A value of x that gives the minimum entropy is the optimum threshold value. We calculate entropy estimates of $p_k(x)$, $q_k(x)$, $p(x)$, and $q(x)$, as follows [Christensen, 1980]:

$$p_k(x) = \frac{n_k(x) + 1}{n(x) + 1} \tag{6.22}$$

$$q_k(x) = \frac{N_k(x) + 1}{N(x) + 1} \tag{6.23}$$

$$p(x) = \frac{n(x)}{n} \tag{6.24}$$

$$q(x) = 1 - p(x) \tag{6.25}$$

where $n_k(x)$ = number of class k samples located in $[x_l, x_l + x]$
$n(x)$ = the total number of samples located in $[x_l, x_l + x]$
$N_k(x)$ = number of class k samples located in $[x_l + x, x_2]$
$N(x)$ = the total number of samples located in $[x_l + x, x_2]$
n = total number of samples in $[x_1, x_2]$
l = a general length along the interval $[x_1, x_2]$

Inductive reasoning

While moving x in the region $[x_1, x_2]$ we calculate the values of entropy for each position of x. The value of x that holds the minimum entropy we will call the primary threshold (PRI) value. With this PRI value, we divide the region $[x_1, x_2]$ in two.

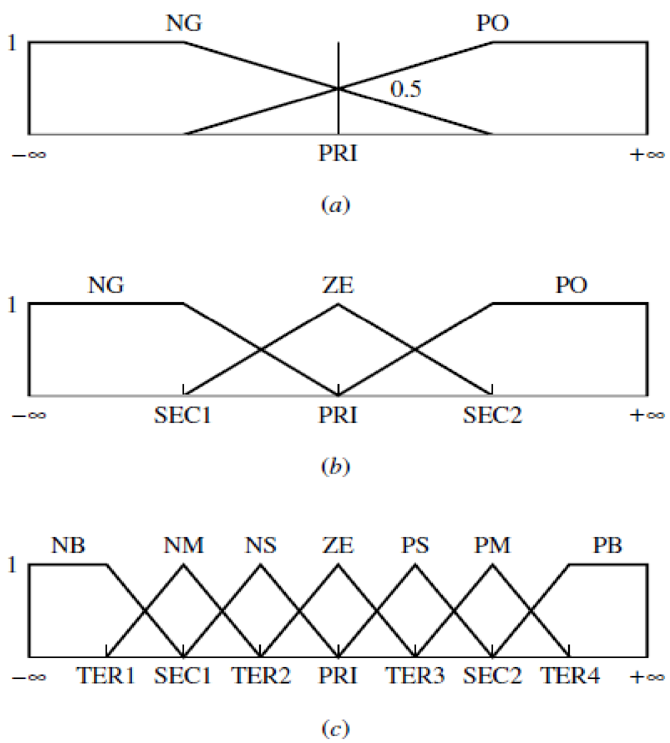


FIGURE 6.18
Repeated partitions and corresponding fuzzy set labels: (*a*) the first partition, (*b*) the second partition, and (*c*) the third partition.

Inductive reasoning

Example 6.6. The shape of an ellipse may be characterized by the ratio of the length of two chords a and b, as shown in Fig. 6.19 (a similar problem was originally posed in Chapter 1; see Fig. 1.3).

Let $x = a/b$; then as the ratio $a/b \to \infty$, the shape of the ellipse tends to a horizontal line, whereas as $a/b \to 0$, the shape tends to a vertical line. For $a/b = 1$ the shape is a circle.

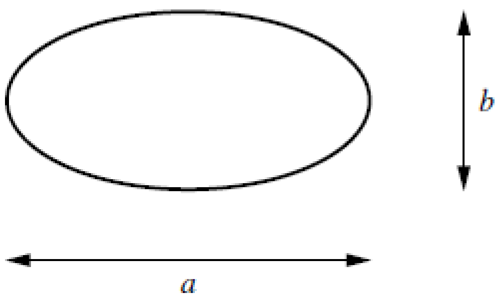


FIGURE 6.19
Geometry of an ellipse.

Inductive reasoning

Boltzmann's entropy formula

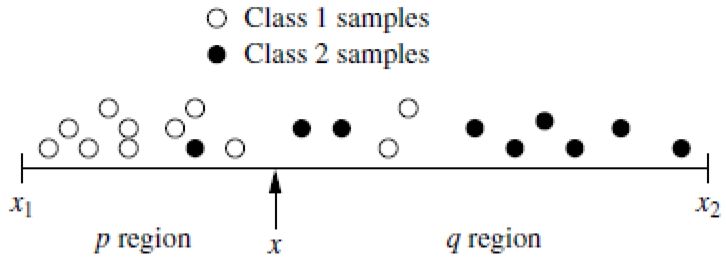


FIGURE 6.17
Illustration of threshold value idea.

An entropy with each value of x in the region x_1 and x_2 is expressed by Christensen [1980] as

$$S(x) = p(x)S_p(x) + q(x)S_q(x) \tag{6.19}$$

where

$$S_p(x) = -[p_1(x)\ln p_1(x) + p_2(x)\ln p_2(x)] \tag{6.20}$$

$$S_q(x) = -[q_1(x)\ln q_1(x) + q_2(x)\ln q_2(x)] \tag{6.21}$$

where $p_k(x)$ and $q_k(x)$ = conditional probabilities that the class k sample is in the region $[x_1, x_1 + x]$ and $[x_1 + x, x_2]$, respectively

$p(x)$ and $q(x)$ = probabilities that all samples are in the region $[x_1, x_1 + x]$ and $[x_1 + x, x_2]$, respectively

$p(x) + q(x) = 1$

Inductive reasoning

A value of x that gives the minimum entropy is the optimum threshold value. We calculate entropy estimates of $p_k(x)$, $q_k(x)$, $p(x)$, and $q(x)$, as follows [Christensen, 1980]:

$$p_k(x) = \frac{n_k(x) + 1}{n(x) + 1} \tag{6.22}$$

$$q_k(x) = \frac{N_k(x) + 1}{N(x) + 1} \tag{6.23}$$

$$p(x) = \frac{n(x)}{n} \tag{6.24}$$

$$q(x) = 1 - p(x) \tag{6.25}$$

where $n_k(x)$ = number of class k samples located in $[x_l, x_l + x]$
$n(x)$ = the total number of samples located in $[x_l, x_l + x]$
$N_k(x)$ = number of class k samples located in $[x_l + x, x_2]$
$N(x)$ = the total number of samples located in $[x_l + x, x_2]$
n = total number of samples in $[x_1, x_2]$
l = a general length along the interval $[x_1, x_2]$

Inductive reasoning

Segmentation of x into two arbitrary classes (from raw data)

$x = a/b$	0	0.1	0.15	0.2	0.2	0.5	0.9	1.1	1.9	5	50	100
Class	1	1	1	1	1	2	1	1	2	2	2	2

TABLE 6.12
Calculations for selection of partition point PRI

x	**0.7**	**1.0**	**1.5**	**3.45**
p_1	$\frac{5+1}{6+1} = \frac{6}{7}$	$\frac{6+1}{7+1} = \frac{7}{8}$	$\frac{7+1}{8+1} = \frac{8}{9}$	$\frac{7+1}{9+1} = \frac{8}{10}$
p_2	$\frac{1+1}{6+1} = \frac{2}{7}$	$\frac{1+1}{7+1} = \frac{2}{8}$	$\frac{1+1}{8+1} = \frac{2}{9}$	$\frac{2+1}{9+1} = \frac{3}{10}$
q_1	$\frac{2+1}{6+1} = \frac{3}{7}$	$\frac{1+1}{5+1} = \frac{2}{6}$	$\frac{0+1}{4+1} = \frac{1}{5}$	$\frac{0+1}{3+1} = \frac{1}{4}$
q_2	$\frac{4+1}{6+1} = \frac{5}{7}$	$\frac{4+1}{5+1} = \frac{5}{6}$	$\frac{4+1}{4+1} = 1.0$	$\frac{3+1}{3+1} = 1.0$
$p(x)$	$\frac{6}{12}$	$\frac{7}{12}$	$\frac{8}{12}$	$\frac{9}{12}$
$q(x)$	$\frac{6}{12}$	$\frac{5}{12}$	$\frac{4}{12}$	$\frac{3}{12}$
$S_p(x)$	0.49	0.463	0.439	0.54
$S_q(x)$	0.603	0.518	0.32	0.347
S	0.547	0.486	0.4 $\surd$	0.49

Inductive reasoning

Segmentation of x into two arbitrary classes (from raw data)

$x = a/b$	0	0.1	0.15	0.2	0.2	0.5	0.9	1.1	1.9	5	50	100
Class	1	1	1	1	1	2	1	1	2	2	2	2

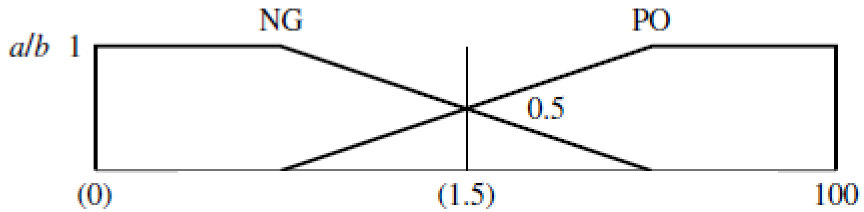


FIGURE 6.20
Partitioning of the variable $x = a/b$ into positive (PO) and negative (NG) partitions.

Inductive reasoning

Segmentation of x into two arbitrary classes (from raw data)

$x = a/b$	0	0.1	0.15	0.2	0.2	0.5	0.9	1.1	1.9	5	50	100
Class	1	1	1	1	1	2	1	1	2	2	2	2

TABLE 6.13
Calculations to determine secondary threshold value: NG side

x	**0.175**	**0.35**	**0.7**
p_1	$\frac{3+1}{3+1} = 1.0$	$\frac{5+1}{5+1} = 1.0$	$\frac{5+1}{6+1} = \frac{6}{7}$
p_2	$\frac{0+1}{3+1} = \frac{1}{4}$	$\frac{0+1}{5+1} = \frac{1}{6}$	$\frac{1+1}{6+1} = \frac{2}{7}$
q_1	$\frac{4+1}{5+1} = \frac{5}{6}$	$\frac{2+1}{3+1} = \frac{3}{4}$	$\frac{2+1}{2+1} = 1.0$
q_2	$\frac{1+1}{5+1} = \frac{2}{6}$	$\frac{1+1}{3+1} = \frac{2}{4}$	$\frac{0+1}{2+1} = \frac{1}{3}$
$p(x)$	$\frac{3}{8}$	$\frac{5}{8}$	$\frac{6}{8}$
$q(x)$	$\frac{5}{8}$	$\frac{3}{8}$	$\frac{2}{8}$
$S_p(x)$	0.347	0.299	0.49
$S_q(x)$	0.518	0.562	0.366
S	0.454	0.398$\checkmark$	0.459

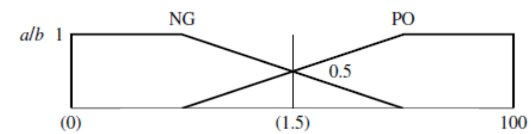
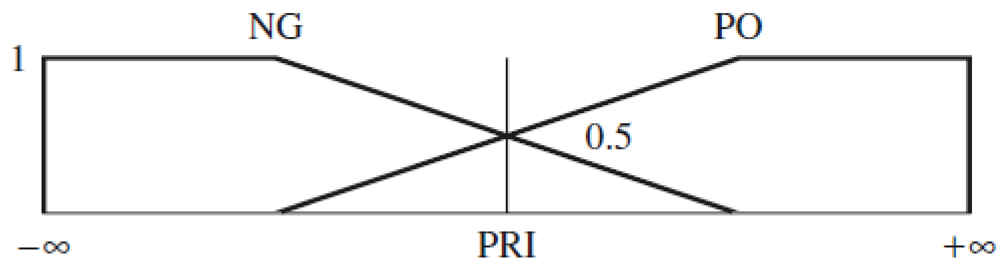
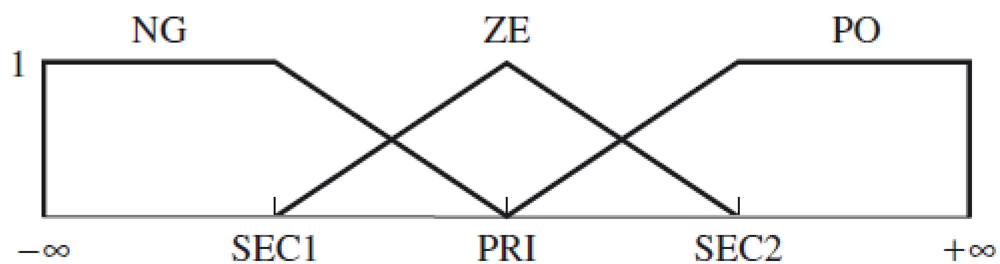


FIGURE 6.20
Partitioning of the variable $x = a/b$ into positive (PO) and negative (NG) partitions.

Inductive reasoning



(a)



(b)