Soft Computing: Fuzzy Logic

Fuzzy Logic

- Is there a need for fuzzy logic?

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making
- this underlies the remarkable human ability to understand

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making
- this underlies the remarkable human ability to understand
- distorted speech,

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making
- this underlies the remarkable human ability to understand
- distorted speech, decipher sloppy handwriting,

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making
- this underlies the remarkable human ability to understand
- distorted speech, decipher sloppy handwriting, comprehend nuances of natural language,

Fuzzy Logic

- Is there a need for fuzzy logic?
- hard computing - decisions or actions are based on precision and certainty
- precision and certainty carry a cost
- soft computing - tolerance for imprecision and uncertainty are explored in decision making
- this underlies the remarkable human ability to understand
- distorted speech, decipher sloppy handwriting, comprehend nuances of natural language, summarize text, and recognize and classify images

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse,

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse, reason and make rational decisions in an environment of

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse, reason and make rational decisions in an environment of imprecision, uncertainty, incompleteness of information, conflicting information, imperfect information

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse, reason and make rational decisions in an environment of imprecision, uncertainty, incompleteness of information, conflicting information, imperfect information
- it is much more than a logical system

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse, reason and make rational decisions in an environment of imprecision, uncertainty, incompleteness of information, conflicting information, imperfect information
- it is much more than a logical system
- poses the ability to mimic the human mind to effectively employ modes of reasoning that are approximate rather than exact

Fuzzy Logic

- Attempt to formalization/mechanization of two remarkable human capabilities:
- the capability to converse, reason and make rational decisions in an environment of imprecision, uncertainty, incompleteness of information, conflicting information, imperfect information
- it is much more than a logical system
- poses the ability to mimic the human mind to effectively employ modes of reasoning that are approximate rather than exact
- it is a precise logic of imprecision and approximate reasoning

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers
- Computing with the words explores imprecision and tolerance

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers
- Computing with the words explores imprecision and tolerance
- In most applications, an FL solution is a translation of a human solution.

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers
- Computing with the words explores imprecision and tolerance
- In most applications, an FL solution is a translation of a human solution.
- FL can model nonlinear functions of arbitrary complexity to a desired degree of accuracy.

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers
- Computing with the words explores imprecision and tolerance
- In most applications, an FL solution is a translation of a human solution.
- FL can model nonlinear functions of arbitrary complexity to a desired degree of accuracy.
- FL is a convenient way to map an input space to an output space.

Fuzzy Logic (FL): Key features

- FL - specify mapping rules in terms of words rather than numbers
- Computing with the words explores imprecision and tolerance
- In most applications, an FL solution is a translation of a human solution.
- FL can model nonlinear functions of arbitrary complexity to a desired degree of accuracy.
- FL is a convenient way to map an input space to an output space.
- FL is one of the tools used to model a multiinput, multioutput system.

Fuzzy Logic Concepts

- Zadeh introduced the term fuzzy logic - the mathematics of fuzzy set theory (1965)

Fuzzy Logic Concepts

- Zadeh introduced the term fuzzy logic - the mathematics of fuzzy set theory (1965)
- Lukasiewicz - multi-valued logic, nothing to prevent the derivation of infinite-valued logic

Fuzzy Logic Concepts

- Zadeh introduced the term fuzzy logic - the mathematics of fuzzy set theory (1965)
- Lukasiewicz - multi-valued logic, nothing to prevent the derivation of infinite-valued logic
- FL concepts consist of

Fuzzy Logic Concepts

- Zadeh introduced the term fuzzy logic - the mathematics of fuzzy set theory (1965)
- Lukasiewicz - multi-valued logic, nothing to prevent the derivation of infinite-valued logic
- FL concepts consist of
  - Fuzzy sets and their properties
  - FL operators
  - Fuzzy proposition and rule-based systems
  - Fuzzy maps and inference engine
  - Defuzzification methods
  - the design of an FL decision system

Fuzzy set

- A fuzzy set is an extension of a crisp set.

Fuzzy set

- A fuzzy set is an extension of a crisp set.
- Crisp sets allow only full membership or no membership

Fuzzy set

- A fuzzy set is an extension of a crisp set.
- Crisp sets allow only full membership or no membership
- membership or nonmembership of element x in crisp set A is described by a characteristic function $\mu_A(x)$

Fuzzy set

- A fuzzy set is an extension of a crisp set.
- Crisp sets allow only full membership or no membership
- membership or nonmembership of element x in crisp set A is described by a characteristic function $\mu_A(x)$
- $\mu_A(x) = 1$ if $x \in A$ and $\mu_A(x) = 0$ if $x \notin A$

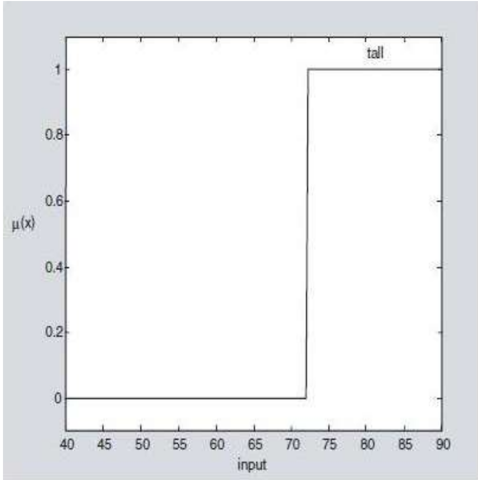
Fuzzy set

- A fuzzy set is an extension of a crisp set.
- Crisp sets allow only full membership or no membership
- membership or nonmembership of element x in crisp set A is described by a characteristic function $\mu_A(x)$
- $\mu_A(x) = 1$ if $x \in A$ and $\mu_A(x) = 0$ if $x \notin A$
- Fuzzy set theory extends this concept by defining partial membership

Fuzzy set

- A fuzzy set is an extension of a crisp set.
- Crisp sets allow only full membership or no membership
- membership or nonmembership of element x in crisp set A is described by a characteristic function $\mu_A(x)$
- $\mu_A(x) = 1$ if $x \in A$ and $\mu_A(x) = 0$ if $x \notin A$
- Fuzzy set theory extends this concept by defining partial membership
- A fuzzy set A on a universe of discourse U is characterized by a membership function $\mu_A(x)$ that takes values in the interval $[0, 1]$

Example: Crisp input



Crisp membership function (src:www)

Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.

Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.
- A given element can be a member of more than one fuzzy set at a time.

Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.
- A given element can be a member of more than one fuzzy set at a time.
- A fuzzy set A in U may be represented as a set of ordered pairs.

Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.
- A given element can be a member of more than one fuzzy set at a time.
- A fuzzy set A in U may be represented as a set of ordered pairs.
- Each pair consists of a generic element x and its grade of membership function

Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.
- A given element can be a member of more than one fuzzy set at a time.
- A fuzzy set A in U may be represented as a set of ordered pairs.
- Each pair consists of a generic element x and its grade of membership function

$$A = \{(x, \mu_A(x))\,|x \in U\}$$

x is called support value if $\mu_A(x) > 0$

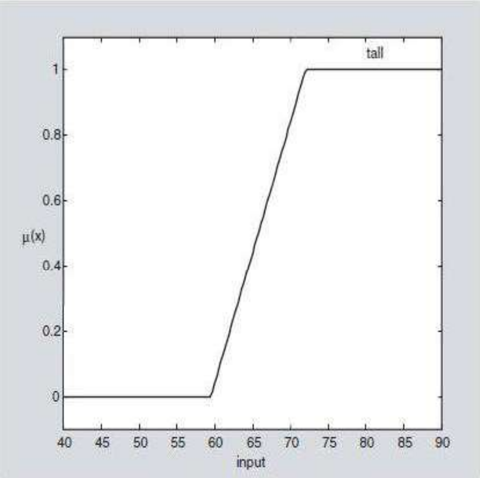
Fuzzy set

- allows to represent commonsense linguistic labels like slow, fast, small, large, heavy, low, medium, high, tall, etc.
- A given element can be a member of more than one fuzzy set at a time.
- A fuzzy set A in U may be represented as a set of ordered pairs.
- Each pair consists of a generic element x and its grade of membership function

$$A = \{(x, \mu_A(x))\,|x \in U\}$$

x is called support value if $\mu_A(x) > 0$
- $\{(\frac{\mu_A(x_i)}{x_i}), i = 1, 2, \ldots, n\}$ $\frac{\mu_A(x_i)}{x_i}$ fuzzy singleton

Example: Fuzzy input



Fuzzy membership function (src:www)

Fuzzy set

- A linguistic variable x in the universe of discourse U is characterized by

Fuzzy set

- A linguistic variable x in the universe of discourse U is characterized by

$$T(x) = \left\{T_x^1, T_x^2, \ldots, T_x^k\right\} \text{ and } \mu(x) = \left\{\mu_x^1, \mu_x^2, \ldots, \mu_x^k\right\}$$

  $T(x)$ is the term set of x - the set of names of linguistic values of x with each T_x^i being a fuzzy number with membership function μ_x^i defined on U

Fuzzy set

- A linguistic variable x in the universe of discourse U is characterized by

$$T(x) = \left\{T_x^1, T_x^2, \ldots, T_x^k\right\} \text{ and } \mu(x) = \left\{\mu_x^1, \mu_x^2, \ldots, \mu_x^k\right\}$$

  $T(x)$ is the term set of x - the set of names of linguistic values of x with each T_x^i being a fuzzy number with membership function μ_x^i defined on U
- x is height then $T(x)$ may refer to sets such as short, medium or tall

Fuzzy set

- A linguistic variable x in the universe of discourse U is characterized by

$$T(x) = \left\{T_x^1, T_x^2, \ldots, T_x^k\right\} \text{ and } \mu(x) = \left\{\mu_x^1, \mu_x^2, \ldots, \mu_x^k\right\}$$

  $T(x)$ is the term set of x - the set of names of linguistic values of x with each T_x^i being a fuzzy number with membership function μ_x^i defined on U
- x is height then $T(x)$ may refer to sets such as short, medium or tall
- A membership function is essentially a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1.

Membership functions

- Various types of membership functions

Membership functions

- Various types of membership functions
- triangular, trapezoidal, generalized bell shaped, Gaussian curves, polynomial curves, sigmoid functions

Membership functions

- Various types of membership functions
- triangular, trapezoidal, generalized bell shaped, Gaussian curves, polynomial curves, sigmoid functions
- triangular curves depend on three parameters a, b, and c and are given by

$$f(x; a, b, c) = \begin{cases} 0 & \text{for } x < a \\ \frac{x-a}{b-a} & \text{for } a \le x < b \\ \frac{c-x}{c-b} & \text{for } b \le x \le c \\ 0 & \text{for } x > c \end{cases}$$

Membership functions

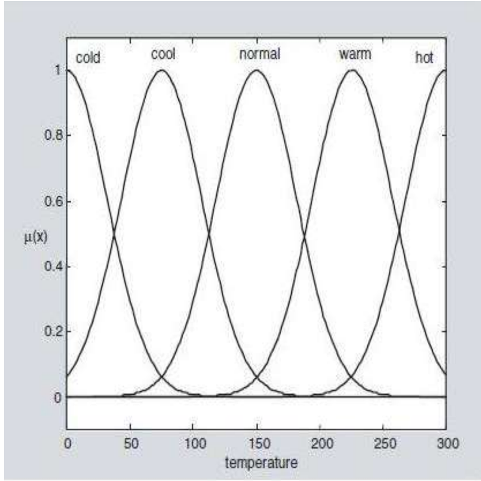
- Various types of membership functions
- triangular, trapezoidal, generalized bell shaped, Gaussian curves, polynomial curves, sigmoid functions
- triangular curves depend on three parameters a, b, and c and are given by

$$f(x;a,b,c) = \begin{cases} 0 & \text{for } x < a \\ \frac{x-a}{b-a} & \text{for } a \le x < b \\ \frac{c-x}{c-b} & \text{for } b \le x \le c \\ 0 & \text{for } x > c \end{cases}$$

- Gaussian shaped membership function

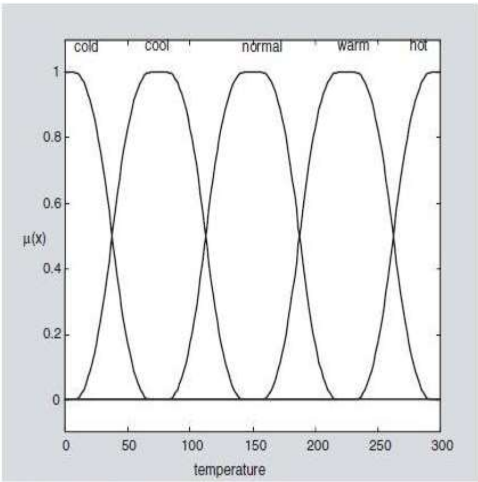
$$f(x;\sigma,c) = \exp\left[\frac{-(x-c)^2}{2\sigma^2}\right]$$

Membership functions



Gaussian membership functions (src:www)

Membership functions



π shaped membership functions (src:www)

Membership functions

- π shaped membership functions are given by

$$f(x;b,c) = \begin{cases} S(x;c-b,c-b/2,c) & \text{for } x \le c \\ 1 - S(x;c,c+b/2,c+b) & \text{for } x > c \end{cases}$$

Membership functions

- π shaped membership functions are given by

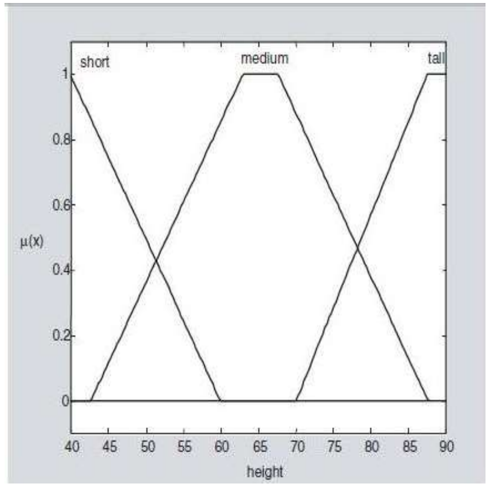
$$f(x;b,c) = \begin{cases} S(x;c-b,c-b/2,c) & \text{for } x \le c \\ 1 - S(x;c,c+b/2,c+b) & \text{for } x > c \end{cases}$$

$S(x;a,b,c)$ represents a membership function defined as

$$S(x;a,b,c) = \begin{cases} 0 & \text{for } x < a \\ \frac{2(x-a)^2}{(c-a)^2} & \text{for } a \le x < b \\ 1 - \frac{2(x-c)^2}{(c-a)^2} & \text{for } b \le x \le c \\ 1 & \text{for } x > c \end{cases}$$

The parameter b is the half width of the curve at the crossover point.

Membership functions



Trapezoidal membership functions (src:www)

Membership functions

- Trapezoidal curves depend on four parameters and are given by

$$f(x;a,b,c,d) = \begin{cases} 0 & \text{for } x < a \\ \frac{x-a}{b-a} & \text{for } a \le x < b \\ 1 & \text{for } b \le x < c \\ \frac{d-x}{d-c} & \text{for } c \le x < d \\ 0 & \text{for } d \le x \end{cases}$$

Membership functions

- Trapezoidal curves depend on four parameters and are given by

$$f(x;a,b,c,d) = \begin{cases} 0 & \text{for } x < a \\ \frac{x-a}{b-a} & \text{for } a \le x < b \\ 1 & \text{for } b \le x < c \\ \frac{d-x}{d-c} & \text{for } c \le x < d \\ 0 & \text{for } d \le x \end{cases}$$

- In designing a fuzzy inference system, membership functions are associated with term sets that appear in the antecedent or consequent of rules.

Logical Operations and If-then Rules

- Fuzzy set operations analogous to crisp set operations

Logical Operations and If-then Rules

- Fuzzy set operations analogous to crisp set operations
- crisp set operations: union, intersection, and complement

Logical Operations and If-then Rules

- Fuzzy set operations analogous to crisp set operations
- crisp set operations: union, intersection, and complement
- correspond to OR, AND, and NOT operators
- A and B be two subsets of U

$$
\begin{aligned}
\mu_{A\cup B}(x) &= 1 \text{ if } x \in A \text{ or } x \in B \\
\mu_{A\cap B}(x) &= 1 \text{ if } x \in A \text{ and } x \in B \\
\mu_{\bar{A}}(x) &= 1 \text{ if } x \notin A
\end{aligned}
$$

Fuzzy operators

- FL operators: min, max, and complement operations - preserve the results of using AND, OR, and NOT

Fuzzy operators

- FL operators: min, max, and complement operations - preserve the results of using AND, OR, and NOT

$$\begin{aligned}\mu_{A\cup B}(x) &= \max[\mu_A(x), \mu_B(x)]\\ \mu_{A\cap B}(x) &= \min[\mu_A(x), \mu_B(x)]\\ \mu_{\bar{A}}(x) &= 1-\mu_A(x)\end{aligned}$$

- Zadeh (1965) defined fuzzy union and fuzzy intersection as

Fuzzy operators

- FL operators: min, max, and complement operations - preserve the results of using AND, OR, and NOT

$$\begin{aligned}\mu_{A\cup B}(x) &= \max[\mu_A(x), \mu_B(x)]\\ \mu_{A\cap B}(x) &= \min[\mu_A(x), \mu_B(x)]\\ \mu_{\bar{A}}(x) &= 1-\mu_A(x)\end{aligned}$$

- Zadeh (1965) defined fuzzy union and fuzzy intersection as

$$\begin{aligned}\mu_{A\cup B}(x) &= \mu_A(x)+\mu_B(x)-\mu_A(x)\mu_B(x)\\ \mu_{A\cap B}(x) &= \mu_A(x)\mu_B(x)\end{aligned}$$

Fuzzy operators

- A is contained in B $(A \subseteq B) \implies \mu_A(x) \leq \mu_B(x)$ for all $x \in R$

Fuzzy operators

- A is contained in B $(A \subseteq B) \implies \mu_A(x) \leq \mu_B(x)$ for all $x \in R$
- concentration (Con) and dilation (Dil) operators

Fuzzy operators

- A is contained in B $(A \subseteq B) \implies \mu_A(x) \leq \mu_B(x)$ for all $x \in R$
- concentration (Con) and dilation (Dil) operators

$$\mu_{Con(A)}(x) = (\mu_A(x))^2 \quad \mu_{Dil(A)}(x) = (\mu_A(x))^{0.5}$$

Fuzzy operators

- A is contained in B $(A \subseteq B) \implies \mu_A(x) \leq \mu_B(x)$ for all $x \in R$
- concentration (Con) and dilation (Dil) operators

$$\mu_{Con(A)}(x) = (\mu_A(x))^2 \quad \mu_{Dil(A)}(x) = (\mu_A(x))^{0.5}$$

- constast intensification operator INT

$$\mu_{INT(A)}(x) = \begin{cases} 2[\mu_A(x)]^2 & \text{for } 0 \leq \mu_A(x) \leq 0.5 \\ 1 - 2[1 - \mu_A(x)]^2 & \text{otherwise} \end{cases}$$

Fuzzy operators

- A is contained in B $(A \subseteq B) \implies \mu_A(x) \leq \mu_B(x)$ for all $x \in R$
- concentration (Con) and dilation (Dil) operators

$$\mu_{Con(A)}(x) = (\mu_A(x))^2 \quad \mu_{Dil(A)}(x) = (\mu_A(x))^{0.5}$$

- constast intensification operator INT

$$\mu_{INT(A)}(x) = \begin{cases} 2[\mu_A(x)]^2 & \text{for } 0 \leq \mu_A(x) \leq 0.5 \\ 1 - 2[1 - \mu_A(x)]^2 & \text{otherwise} \end{cases}$$

- fuzzy modifiers not, very, and more or less
- $\mu_{not}(x) = 1 - \mu(x)$ $\mu_{very}(x) = (\mu(x))^2$ $\mu_{notvery}(x) = 1 - (\mu(x))^2$ $\mu_{moreorless} = \mu(x)^{0.5}$

Measure of fuzziness

- degree of fuzziness expresses the average amount of ambiguity in taking a decision as to whether an element belongs to the set

Measure of fuzziness

- degree of fuzziness expresses the average amount of ambiguity in taking a decision as to whether an element belongs to the set
- $I(A) = minimum$ iff $\mu_A(x_i) = 0$ *or* $1 \forall i$
- $I(A) = maximum$ iff $\mu_A(x_i) = 0.5 \forall i$
- $I(A) \geq I(A^*)$ A^* sharpened version of A

$$\mu_{A^*}(x_i) \geq \mu_A(x_i) \text{ if } \mu_A(x_i) \geq 0.5$$
$$\mu_{A^*}(x_i) \leq \mu_A(x_i) \text{ if } \mu_A(x_i) \leq 0.5$$

Measure of fuzziness

- degree of fuzziness expresses the average amount of ambiguity in taking a decision as to whether an element belongs to the set
- $I(A) = minimum$ iff $\mu_A(x_i) = 0$ *or* $1 \forall i$
- $I(A) = maximum$ iff $\mu_A(x_i) = 0.5 \forall i$
- $I(A) \geq I(A^*)$ A^* sharpened version of A

$$\mu_{A^*}(x_i) \geq \mu_A(x_i) \text{ if } \mu_A(x_i) \geq 0.5$$
$$\mu_{A^*}(x_i) \leq \mu_A(x_i) \text{ if } \mu_A(x_i) \leq 0.5$$

- $I(A) = I(A^c)$
- index of fuzziness of a fuzzy set A having n supporting points is defined $\nu(A) = \frac{2}{n^k} d(A, \underline{A})$

M. A. Zaveri, SVNIT, Surat | Soft Computing Techniques | 3 April 2011 | 25 / 157

Measure of fuzziness

- degree of fuzziness expresses the average amount of ambiguity in taking a decision as to whether an element belongs to the set
- $I(A) = minimum$ iff $\mu_A(x_i) = 0$ *or* $1 \forall i$
- $I(A) = maximum$ iff $\mu_A(x_i) = 0.5 \forall i$
- $I(A) \geq I(A^*)$ A^* sharpened version of A

$$\mu_{A^*}(x_i) \geq \mu_A(x_i) \text{ if } \mu_A(x_i) \geq 0.5$$
$$\mu_{A^*}(x_i) \leq \mu_A(x_i) \text{ if } \mu_A(x_i) \leq 0.5$$

- $I(A) = I(A^c)$
- index of fuzziness of a fuzzy set A having n supporting points is defined $\nu(A) = \frac{2}{n^k} d(A, \underline{A})$

- $d(A, \underline{A})$ - distance between fuzzy set A and its nearest ordinary set $\underline{A}$ which is defined as

M. A. Zaveri, SVNIT, Surat | Soft Computing Techniques | 3 April 2011 | 25 / 157

Measure of fuzziness

- degree of fuzziness expresses the average amount of ambiguity in taking a decision as to whether an element belongs to the set
- $I(A) = minimum$ iff $\mu_A(x_i) = 0$ *or* $1 \forall i$
- $I(A) = maximum$ iff $\mu_A(x_i) = 0.5 \forall i$
- $I(A) \geq I(A^*)$ A^* sharpened version of A

$$\mu_{A^*}(x_i) \geq \mu_A(x_i) \text{ if } \mu_A(x_i) \geq 0.5$$
$$\mu_{A^*}(x_i) \leq \mu_A(x_i) \text{ if } \mu_A(x_i) \leq 0.5$$

- $I(A) = I(A^c)$
- index of fuzziness of a fuzzy set A having n supporting points is defined $\nu(A) = \frac{2}{n^k} d(A, \underline{A})$
- $d(A, \underline{A})$ - distance between fuzzy set A and its nearest ordinary set $\underline{A}$ which is defined as

$$\mu_{\underline{A}}(x) = \begin{cases} 0 & \text{if } \mu_A(x) \leq 0.5 \\ 1 & \text{if } \mu_A(x) \geq 0.5 \end{cases}$$

M. A. Zaveri, SVNIT, Surat | Soft Computing Techniques | 3 April 2011 | 25 / 157

Measure of fuzziness

- k depends on the type of distance used $k = 1$ for Hamming distance and $k = 0.5$ for Euclidean distance

Measure of fuzziness

- k depends on the type of distance used $k = 1$ for Hamming distance and $k = 0.5$ for Euclidean distance
- corresponding indices of fuzziness are called linear index of fuzziness $\nu_l(A)$ and quadratic index of fuzziness $\nu_q(A)$ respecitvely defined as

Measure of fuzziness

- k depends on the type of distance used $k = 1$ for Hamming distance and $k = 0.5$ for Euclidean distance
- corresponding indices of fuzziness are called linear index of fuzziness $\nu_l(A)$ and quadratic index of fuzziness $\nu_q(A)$ respecitvely defined as

$$
\begin{aligned}
\nu_l(A) &= \frac{2}{n}\sum_{i=1}^{n} |\mu_A(x_i) - \mu_{\underline{A}}(x_i)| \\
&= \frac{2}{n}\sum_{i=1}^{n} [\min\{\mu_A(x_i), (1 - \mu_A(x_i))\}]
\end{aligned}
$$

Measure of fuzziness

- k depends on the type of distance used $k = 1$ for Hamming distance and $k = 0.5$ for Euclidean distance
- corresponding indices of fuzziness are called linear index of fuzziness $\nu_l(A)$ and quadratic index of fuzziness $\nu_q(A)$ respecitvely defined as

$$
\begin{aligned}
\nu_l(A) &= \frac{2}{n}\sum_{i=1}^{n} |\mu_A(x_i) - \mu_{\underline{A}}(x_i)| \\
&= \frac{2}{n}\sum_{i=1}^{n} [\min\{\mu_A(x_i), (1 - \mu_A(x_i))\}]
\end{aligned}
$$

$$
\nu_q(A) = \frac{2}{\sqrt{n}}\sqrt{\left[\sum_{i=1}^{n} \{\mu_A(x_i) - \mu_{\underline{A}}(x_i)\}^2\right]}
$$

Measure of fuzziness

- Entroy of fuzzy set

Measure of fuzziness

- Entroy of fuzzy set

$$
\begin{aligned}
H(A) &= \frac{1}{n\ln(2)}\sum_{i=1}^{n}\{S_n(\mu_A(x_i))\} \\
S_n(\mu_A(x_i)) &= -\mu_A(x_i)\ln(\mu_A(x_i)) - \{1-\mu_A(x_i)\}\ln\{1-\mu_A(x_i)\}
\end{aligned}
$$

Measure of fuzziness

- Entroy of fuzzy set

$$
\begin{aligned}
H(A) &= \frac{1}{n\ln(2)}\sum_{i=1}^{n}\{S_n(\mu_A(x_i))\} \\
S_n(\mu_A(x_i)) &= -\mu_A(x_i)\ln(\mu_A(x_i)) - \{1-\mu_A(x_i)\}\ln\{1-\mu_A(x_i)\} \\
H(A) &= \frac{1}{n(\sqrt{e}-1)}\sum_{i=1}^{n}\{S_n(\mu_A(x_i))-1\} \\
S_n(\mu_A(x_i)) &= \mu_A(x_i)e^{1-\mu_A(x_i)} + (1-\mu_A(x_i))e^{\mu_A(x_i)}
\end{aligned}
$$

T-norm and T-conorm

- triangular norm and conorms (T-operators)

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)
- a and b over $[0, 1]$ then AND (T-operation) as a T mapping function $T : [0, 1] \times [0, 1] \rightarrow [0, 1]$ $y = [a \text{ AND } b] \triangleq [aTb] = T[a, b]$

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)
- a and b over $[0, 1]$ then AND (T-operation) as a T mapping function $T : [0, 1] \times [0, 1] \rightarrow [0, 1]$ $y = [a \text{ AND } b] \triangleq [aTb] = T[a, b]$
- generalized OR (t-conorm) is defined as an S mapping function $S : [0, 1] \times [0, 1] \rightarrow [0, 1]$ $y = [a \text{ OR } b] \triangleq [aSb] = S[a, b]$

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)
- a and b over $[0, 1]$ then AND (T-operation) as a T mapping function $T : [0, 1] \times [0, 1] \rightarrow [0, 1]$ $y = [a \text{ AND } b] \triangleq [aTb] = T[a, b]$
- generalized OR (t-conorm) is defined as an S mapping function $S : [0, 1] \times [0, 1] \rightarrow [0, 1]$ $y = [a \text{ OR } b] \triangleq [aSb] = S[a, b]$
- important properties of T operator

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)
- a and b over $[0, 1]$ then AND (T-operation) as a T mapping function $T : [0, 1] \times [0, 1] \to [0, 1]$ $y = [a \text{ AND } b] \triangleq [aTb] = T[a, b]$
- generalized OR (t-conorm) is defined as an S mapping function $S : [0, 1] \times [0, 1] \to [0, 1]$ $y = [a \text{ OR } b] \triangleq [aSb] = S[a, b]$
- important properties of T operator
- $T(a, b) = T(b, a)$ commutativity
- $T(T(a, b), c) = T(a, T(b, c))$ associativity
- $T(a, b) \geq T(c, d)$ if $a \geq c$ and $b \geq d$ monotonicity
- $T(a, 1) = a$ boundary condition

T-norm and T-conorm

- triangular norm and conorms (T-operators)
- generalized form of logical operations (connectives) - OR (union) and AND (intersection)
- a and b over $[0, 1]$ then AND (T-operation) as a T mapping function $T : [0, 1] \times [0, 1] \to [0, 1]$ $y = [a \text{ AND } b] \triangleq [aTb] = T[a, b]$
- generalized OR (t-conorm) is defined as an S mapping function $S : [0, 1] \times [0, 1] \to [0, 1]$ $y = [a \text{ OR } b] \triangleq [aSb] = S[a, b]$
- important properties of T operator
- $T(a, b) = T(b, a)$ commutativity
- $T(T(a, b), c) = T(a, T(b, c))$ associativity
- $T(a, b) \geq T(c, d)$ if $a \geq c$ and $b \geq d$ monotonicity
- $T(a, 1) = a$ boundary condition
- t-conorm differens from a t-norm in boundary condition t-conorm requires $S(a, 0) = a$

T-norm and T-conorm

- family of operators

T-norm and T-conorm

- family of operators

$$
\begin{aligned}
T^m(a, b) &= a \cap b \text{ min} \\
T^p(a, b) &= a * b \text{ product} \\
T^q(a, b) &= \max(0, x + y - 1) \text{ quasilinear} \\
T(a, b) &= a \text{ if } b = 1 \\
&= b \text{ if } a = 1 \\
&= 0 \text{ otherwise}
\end{aligned}
$$

T-norm and T-conorm

- family of operators

$$
\begin{aligned}
T^m(a,b) &= a \cap b \text{ min} \\
T^p(a,b) &= a * b \text{ product} \\
T^q(a,b) &= \max(0, x+y-1) \text{ quasilinear} \\
T(a,b) &= a \text{ if } b = 1 \\
&= b \text{ if } a = 1 \\
&= 0 \text{ otherwise} \\
S^m(a,b) &= a \cap b \text{ max} \\
S^p(a,b) &= a + b - ab \text{ probabilistic sum} \\
S^q(a,b) &= \min(1, a+b) \text{ quasilinear}
\end{aligned}
$$

Fuzzy Inference System (FIS)

- FIS contains four components:

Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier

Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier
- It essentially defines a nonlinear mapping of the input data vector into a scalar output, using fuzzy rules.

Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier
- It essentially defines a nonlinear mapping of the input data vector into a scalar output, using fuzzy rules.
- An FIS with multiple outputs can be considered as a collection of independent multiinput, single-output systems.

Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier
- It essentially defines a nonlinear mapping of the input data vector into a scalar output, using fuzzy rules.
- An FIS with multiple outputs can be considered as a collection of independent multiinput, single-output systems.
- The mapping process involves input/output membership functions, FL operators, fuzzy if-then rules, aggregation of output sets, and defuzzification.

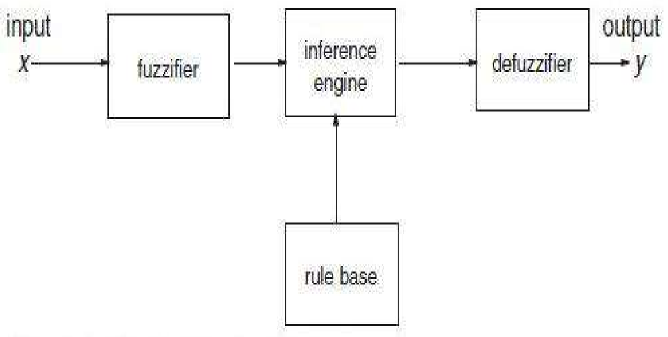
Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier
- It essentially defines a nonlinear mapping of the input data vector into a scalar output, using fuzzy rules.
- An FIS with multiple outputs can be considered as a collection of independent multiinput, single-output systems.
- The mapping process involves input/output membership functions, FL operators, fuzzy if-then rules, aggregation of output sets, and defuzzification.
- The inference engine defines mapping from input fuzzy sets into output fuzzy sets.

Fuzzy Inference System (FIS)

- FIS contains four components: the fuzzifier, inference engine, rule base, and defuzzifier
- It essentially defines a nonlinear mapping of the input data vector into a scalar output, using fuzzy rules.
- An FIS with multiple outputs can be considered as a collection of independent multiinput, single-output systems.
- The mapping process involves input/output membership functions, FL operators, fuzzy if-then rules, aggregation of output sets, and defuzzification.
- The inference engine defines mapping from input fuzzy sets into output fuzzy sets.
- The rule base contains linguistic rules that are provided by experts.

Fuzzy Inference System



Block diagram of a fuzzy inference system (src:www)

Fuzzy Inference System

- if-then rules specify a relationship between the input and output fuzzy sets

Fuzzy Inference System

- if-then rules specify a relationship between the input and output fuzzy sets
- Fuzzy relations play an important role in fuzzy inference systems

Fuzzy Inference System

- if-then rules specify a relationship between the input and output fuzzy sets
- Fuzzy relations play an important role in fuzzy inference systems
- Fuzzy relations present a degree of presence or absence of association or interaction between the elements of two or more sets.

Fuzzy Inference System

- if-then rules specify a relationship between the input and output fuzzy sets
- Fuzzy relations play an important role in fuzzy inference systems
- Fuzzy relations present a degree of presence or absence of association or interaction between the elements of two or more sets.
- Let U and V be two universes of discourse
- A fuzzy relation $R(U, V)$ is a set in the product space $U \times V$ and is characterized by the membership function $\mu_R(x, y)$ where, $x \in U$ and $y \in V$ $\mu_R(x, y) \in [0, 1]$

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B
- if part of the rule, x is A, is called the antecedent or premise

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B
- if part of the rule, x is A, is called the antecedent or premise
- then part of the rule, y is B, is called the consequent or conclusion

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B
- if part of the rule, x is A, is called the antecedent or premise
- then part of the rule, y is B, is called the consequent or conclusion
- Interpreting an if-then rule involves two distinct steps:

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B
- if part of the rule, x is A, is called the antecedent or premise
- then part of the rule, y is B, is called the consequent or conclusion
- Interpreting an if-then rule involves two distinct steps:
- The first step is to evaluate the antecedent, which involves fuzzifying the input

Fuzzy Inference System

- A singleton fuzzy rule assumes the form - if x is A, then y is B
- if part of the rule, x is A, is called the antecedent or premise
- then part of the rule, y is B, is called the consequent or conclusion
- Interpreting an if-then rule involves two distinct steps:
- The first step is to evaluate the antecedent, which involves fuzzifying the input
- The second step is implication, or applying the result of the antecedent to the consequent, which essentially evaluates the membership function $\mu_{A \to B}(x, y)$

Fuzzy Inference System

- For most applications, the fuzzy membership function $\mu_{A \to B}(x, y)$ for a given relation is obtained with the minimum or product implication

Fuzzy Inference System

- For most applications, the fuzzy membership function $\mu_{A \to B}(x, y)$ for a given relation is obtained with the minimum or product implication

$$\begin{aligned} \mu_{A \cap B}(x) &= \mu_A(x)\mu_B(x) \\ \mu_{A \cap B}(x) &= \min[\mu_A(x), \mu_B(x)] \end{aligned}$$

- It determines the degree to which the antecedent is satisfied for each rule.

Fuzzy Inference System

- For most applications, the fuzzy membership function $\mu_{A \to B}(x, y)$ for a given relation is obtained with the minimum or product implication

$$\begin{aligned} \mu_{A \cap B}(x) &= \mu_A(x)\mu_B(x) \\ \mu_{A \cap B}(x) &= \min[\mu_A(x), \mu_B(x)] \end{aligned}$$

- It determines the degree to which the antecedent is satisfied for each rule.
- If the antecedent of a given rule has more than one clause, fuzzy operators are applied to obtain one number that represents the result of the antecedent for that rule.

Fuzzy Inference System

- It is possible that one or more rules may fire at the same time.

Fuzzy Inference System

- It is possible that one or more rules may fire at the same time.
- Outputs for all rules are then aggregated.

Fuzzy Inference System

- It is possible that one or more rules may fire at the same time.
- Outputs for all rules are then aggregated.
- During aggregation, fuzzy sets that represent the output of each rule are combined into a single fuzzy set.
- Fuzzy rules are fired in parallel.

Fuzzy Inference System

- It is possible that one or more rules may fire at the same time.
- Outputs for all rules are then aggregated.
- During aggregation, fuzzy sets that represent the output of each rule are combined into a single fuzzy set.
- Fuzzy rules are fired in parallel.
- In an FIS, the order in which rules are fired does not affect the output.

Fuzzy Inference System

- The defuzzifier maps output fuzzy sets into a crisp number.

Fuzzy Inference System

- The defuzzifier maps output fuzzy sets into a crisp number.
- Given a fuzzy set that encompasses a range of output values, the defuzzifier returns one number, thereby moving from a fuzzy set to a crisp number.

Fuzzy Inference System

- The defuzzifier maps output fuzzy sets into a crisp number.
- Given a fuzzy set that encompasses a range of output values, the defuzzifier returns one number, thereby moving from a fuzzy set to a crisp number.
- Several methods for defuzzification are used in practice.

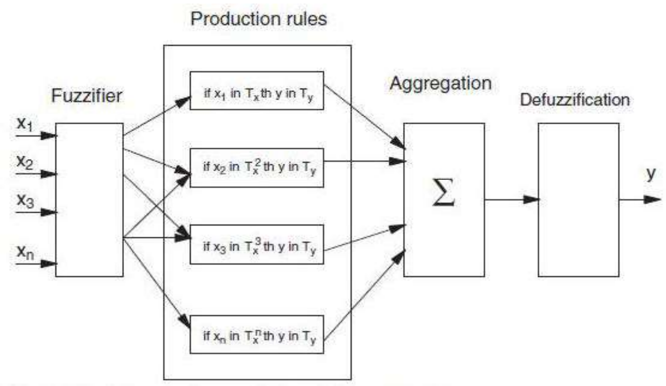
Fuzzy Inference System

- The defuzzifier maps output fuzzy sets into a crisp number.
- Given a fuzzy set that encompasses a range of output values, the defuzzifier returns one number, thereby moving from a fuzzy set to a crisp number.
- Several methods for defuzzification are used in practice.
- the centroid, maximum, mean of maxima, height, and modified height defuzzifier

Fuzzy Inference System

- The defuzzifier maps output fuzzy sets into a crisp number.
- Given a fuzzy set that encompasses a range of output values, the defuzzifier returns one number, thereby moving from a fuzzy set to a crisp number.
- Several methods for defuzzification are used in practice.
- the centroid, maximum, mean of maxima, height, and modified height defuzzifier
- The most popular defuzzification method is the centroid, which calculates and returns the center of gravity of the aggregated fuzzy set.

Fuzzy Inference System



Schematic diagram of a fuzzy inference system (src:www)

Fuzzy Inference System

- In an FIS, multiple regions are combined in the output space to produce a composite region.

Fuzzy Inference System

- In an FIS, multiple regions are combined in the output space to produce a composite region.
- For a multiinput, multioutput system

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 38 / 157

Fuzzy Inference System

- In an FIS, multiple regions are combined in the output space to produce a composite region.
- For a multiinput, multioutput system
- Let $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n})^{\mathbf{T}}$ be the input vector and $\mathbf{y} = (\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_m})^{\mathbf{T}}$ be the output vector.

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 38 / 157

Fuzzy Inference System

- In an FIS, multiple regions are combined in the output space to produce a composite region.
- For a multiinput, multioutput system
- Let $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n})^{\mathbf{T}}$ be the input vector and $\mathbf{y} = (\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_m})^{\mathbf{T}}$ be the output vector.
- The linguistic variable x_i in the universe of discourse U is characterized by

$$\mathbf{T}(\mathbf{x}) = \left\{\mathbf{T}_{\mathbf{x}}^{\mathbf{1}}, \mathbf{T}_{\mathbf{x}}^{\mathbf{2}}, \ldots, \mathbf{T}_{\mathbf{x}}^{\mathbf{k}}\right\} \text{ and } \mu(\mathbf{x}) = \{\mu_{\mathbf{x}}^{\mathbf{1}}, \mu_{\mathbf{x}}^{\mathbf{2}}, \ldots, \mu_{\mathbf{x}}^{\mathbf{k}}\}$$

- $\mathbf{T}(\mathbf{x})$ is a term set of $\mathbf{x}$; that is, it is the set of names of linguistic values of $\mathbf{x}$, with each T_x^i being a fuzzy member and the membership function μ_x^i defined on U.

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 38 / 157

Example: Fuzzy Inference System

- fuzzy inference system with two inputs ($n = 2$) and one output ($m = 1$)

Example: Fuzzy Inference System

- fuzzy inference system with two inputs ($n = 2$) and one output ($m = 1$)
- Let the two inputs represent the number of years of education and the number of years of experience, and let the output of the system be salary.

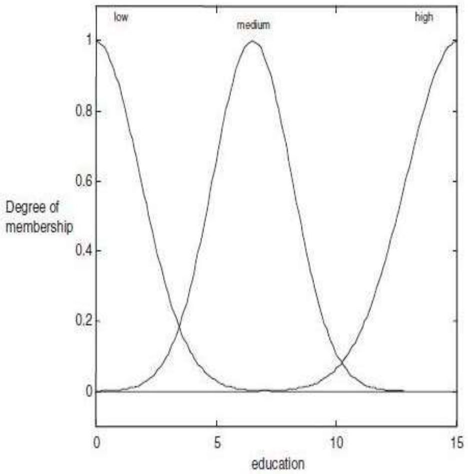
Example: Fuzzy Inference System

- fuzzy inference system with two inputs ($n = 2$) and one output ($m = 1$)
- Let the two inputs represent the number of years of education and the number of years of experience, and let the output of the system be salary.
- Let x_1 indicate the number of years of education, $T(x_1)$ represent its term set $\{low, medium, high\}$, and the universe of discourse be $[0 - 15]$.

Example: Fuzzy Inference System

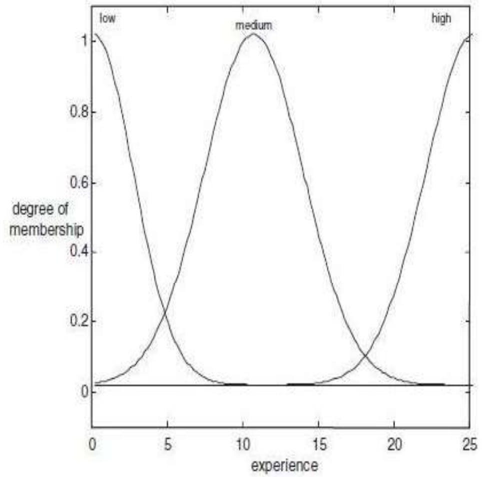
- fuzzy inference system with two inputs ($n = 2$) and one output ($m = 1$)
- Let the two inputs represent the number of years of education and the number of years of experience, and let the output of the system be salary.
- Let x_1 indicate the number of years of education, $T(x_1)$ represent its term set $\{low, medium, high\}$, and the universe of discourse be $[0 - 15]$.
- Let x_2 indicate the number of years of experience, the universe of discourse be $[0 - 30]$, and the corresponding term set be $\{low, medium, high\}$.

FIS Example: Input x_1



Fuzzy membership functions for Education input (src:www)

FIS Example: Input x_2



Fuzzy membership functions for Experience input (src:www)

FIS Example: Output (Salary)

- linguistic variable y in the universe of discourse V is characterized by $T(\mathbf{y}) = \{\mathbf{T_y^1}, \mathbf{T_y^2}, \ldots, \mathbf{T_y^l}\}$

FIS Example: Output (Salary)

- linguistic variable y in the universe of discourse V is characterized by $T(\mathbf{y}) = \{\mathbf{T_y^1}, \mathbf{T_y^2}, \ldots, \mathbf{T_y^l}\}$
- $T(\mathbf{y})$ is a term set of $\mathbf{y}$; that is, T is the set of names of linguistic values of $\mathbf{y}$, with each T_y^i being a fuzzy membership function μ_y^i defined on V.

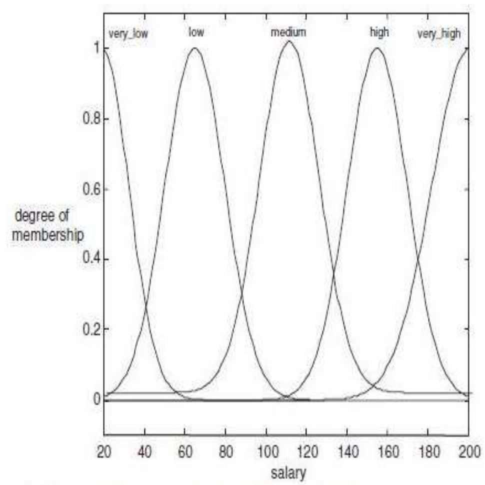
FIS Example: Output (Salary)

- linguistic variable y in the universe of discourse V is characterized by $T(\mathbf{y}) = \{\mathbf{T_y^1}, \mathbf{T_y^2}, \ldots, \mathbf{T_y^l}\}$
- $T(\mathbf{y})$ is a term set of $\mathbf{y}$; that is, T is the set of names of linguistic values of $\mathbf{y}$, with each T_y^i being a fuzzy membership function μ_y^i defined on V.
- Variable y represents salary, then $T(y)$ represents a term set $\{verylow, low, medium, high, veryhigh\}$, and the universe of discourse is $[20 - 200]$, which represents the minimum and maximum in thousands of dollars, that is, 20,000, and 200,000, respectively.

FIS Example: Output (Salary)

- linguistic variable y in the universe of discourse V is characterized by $T(\mathbf{y}) = \{\mathbf{T_y^1}, \mathbf{T_y^2}, \ldots, \mathbf{T_y^l}\}$
- $T(\mathbf{y})$ is a term set of $\mathbf{y}$; that is, T is the set of names of linguistic values of $\mathbf{y}$, with each T_y^i being a fuzzy membership function μ_y^i defined on V.
- Variable y represents salary, then $T(y)$ represents a term set $\{verylow, low, medium, high, veryhigh\}$, and the universe of discourse is $[20 - 200]$, which represents the minimum and maximum in thousands of dollars, that is, 20,000, and 200,000, respectively.
- In order to map input variables x_1 and x_2 to output y, it is necessary that we first define the corresponding fuzzy sets.

FIS Example: Output y



Fuzzy membership function for salary output (src:www)

FIS Example

- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.

FIS Example

- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.
- The fuzzifier block performs the mapping from the input feature space to fuzzy sets in a certain universe of discourse.

FIS Example

- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.
- The fuzzifier block performs the mapping from the input feature space to fuzzy sets in a certain universe of discourse.
- A specific value x_1 is then mapped to the fuzzy set $T^1_{x_i}$ with degree $\mu^1_{x_i}$, x_2 to fuzzy set $T^2_{x_i}$ with degree $\mu^2_{x_i}$, and so on.

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 44 / 157

FIS Example

- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.
- The fuzzifier block performs the mapping from the input feature space to fuzzy sets in a certain universe of discourse.
- A specific value x_1 is then mapped to the fuzzy set $T^1_{x_i}$ with degree $\mu^1_{x_i}$, x_2 to fuzzy set $T^2_{x_i}$ with degree $\mu^2_{x_i}$, and so on.
- In order to perform this mapping, we can use fuzzy sets of any shape, such as triangular, Gaussian, π-shaped, etc.

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 44 / 157

FIS Example

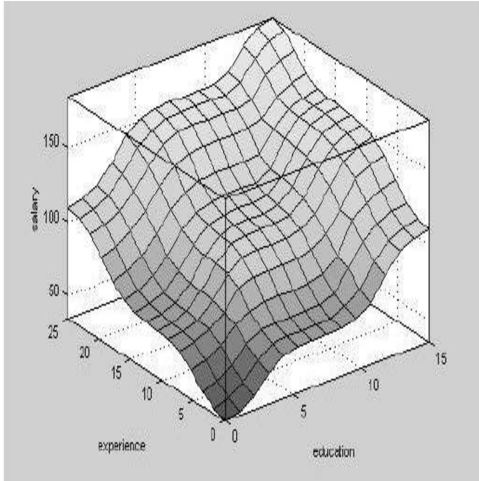
- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.
- The fuzzifier block performs the mapping from the input feature space to fuzzy sets in a certain universe of discourse.
- A specific value x_1 is then mapped to the fuzzy set $T^1_{x_i}$ with degree $\mu^1_{x_i}$, x_2 to fuzzy set $T^2_{x_i}$ with degree $\mu^2_{x_i}$, and so on.
- In order to perform this mapping, we can use fuzzy sets of any shape, such as triangular, Gaussian, π-shaped, etc.
- A fuzzy rule base contains a set of fuzzy rules R. A single if-then rule assumes the form -if x is T_x then y is T_y.

M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 44 / 157

FIS Example

- The first step in evaluating the output of a FIS is to apply the inputs and determine the degree to which they belong to each of the fuzzy sets.
- The fuzzifier block performs the mapping from the input feature space to fuzzy sets in a certain universe of discourse.
- A specific value x_1 is then mapped to the fuzzy set $T^1_{x_i}$ with degree $\mu^1_{x_i}$, x_2 to fuzzy set $T^2_{x_i}$ with degree $\mu^2_{x_i}$, and so on.
- In order to perform this mapping, we can use fuzzy sets of any shape, such as triangular, Gaussian, π-shaped, etc.
- A fuzzy rule base contains a set of fuzzy rules R. A single if-then rule assumes the form -if x is T_x then y is T_y.
- An example of a rule might be if *education* is *high* and *experience* is *high*, then *salary* is very *high*.

Mapping surface



Example of Mapping surface (salary) (src:www)

FIS Example

- For a multiinput, multioutput system,

$$\begin{aligned} R &= (R_1, R_2, \ldots, R_n) \\ R_i &= \text{ if } \left(x_1 \text{is } T_{x_1}, \text{and } \ldots, x_p \text{ is } T_{x_p}\right) \text{ then} \\ & \quad \left(y_1 \text{ in } T_{y_1}, \text{ and} \ldots, y_q \text{ in } T_{y_q}\right) \end{aligned}$$

FIS Example

- For a multiinput, multioutput system,

$$\begin{aligned} R &= (R_1, R_2, \ldots, R_n) \\ R_i &= \text{ if } \left(x_1 \text{is } T_{x_1}, \text{and } \ldots, x_p \text{ is } T_{x_p}\right) \text{ then} \\ & \quad \left(y_1 \text{ in } T_{y_1}, \text{ and} \ldots, y_q \text{ in } T_{y_q}\right) \end{aligned}$$

- The p preconditions of R_i form a fuzzy set $T_{x_1} \times T_{x_2} \times \cdots \times T_{x_p}$, and the consequent is the union of q independent outputs.

FIS Example

- For a multiinput, multioutput system,

$$
\begin{aligned}
R &= (R_1, R_2, \ldots, R_n) \\
R_i &= \text{ if } \left(x_1 \text{is } T_{x_1}, \text{and } \ldots, x_p \text{ is } T_{x_p}\right) \text{ then} \\
&\quad \left(y_1 \text{ in } T_{y_1}, \text{ and} \ldots, y_q \text{ in } T_{y_q}\right)
\end{aligned}
$$

- The p preconditions of R_i form a fuzzy set $T_{x_1} \times T_{x_2} \times \cdots \times T_{x_p}$, and the consequent is the union of q independent outputs.
- If we consider a multiinput, single-output system, then the consequent reduces to $(y_1 \text{ is } T_{y_1})$.

FIS Example

- For the given example, the rules are stated as

FIS Example

- For the given example, the rules are stated as
- R_1: if edu is low and exp is low, then sal is very low

FIS Example

- For the given example, the rules are stated as
- R_1: if edu is low and exp is low, then sal is very low
- R_2: if edu is low and exp is medium, then sal is low

FIS Example

- For the given example, the rules are stated as
- R_1: if edu is low and exp is low, then sal is very low
- R_2: if edu is low and exp is medium, then sal is low
- R_3: if edu is low and exp is high, then sal is medium

FIS Example

- For the given example, the rules are stated as
- R_1: if edu is low and exp is low, then sal is very low
- R_2: if edu is low and exp is medium, then sal is low
- R_3: if edu is low and exp is high, then sal is medium
- R_4: if edu is medium and exp is low, then sal is low
- R_5: if edu is medium and exp is medium, then sal is medium
- R_6: if edu is medium and exp is high, then sal is high
- R_7: if edu is high and exp is low, then sal is medium
- R_8: if edu is high and exp is medium, then sal is high
- R_9: if edu is high and exp is high, then sal is very high

Fuzzy rule

- Interpreting if-then rule three part process:

Fuzzy rule

- Interpreting if-then rule three part process:
  - Resolve all fuzzy statements in the antecedent to a degree of membership between 0 and 1

Fuzzy rule

- Interpreting if-then rule three part process:
  - Resolve all fuzzy statements in the antecedent to a degree of membership between 0 and 1
  - if there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1, is the result being the degree of support for the rule

Fuzzy rule

- Interpreting if-then rule three part process:
  - Resolve all fuzzy statements in the antecedent to a degree of membership between 0 and 1
  - if there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1, is the result being the degree of support for the rule
  - apply the implication method, using the degree of support for the entire rule to shape the output fuzzy set

Fuzzy rule

- Interpreting if-then rule three part process:
  - Resolve all fuzzy statements in the antecedent to a degree of membership between 0 and 1
  - if there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1, is the result being the degree of support for the rule
  - apply the implication method, using the degree of support for the entire rule to shape the output fuzzy set
  - If the rule has more than one antecedent, the fuzzy operator is applied to obtain one number that represents the result of applying that rule

Fuzzy rule

- Interpreting if-then rule three part process:
  - Resolve all fuzzy statements in the antecedent to a degree of membership between 0 and 1
  - if there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1, is the result being the degree of support for the rule
  - apply the implication method, using the degree of support for the entire rule to shape the output fuzzy set
  - If the rule has more than one antecedent, the fuzzy operator is applied to obtain one number that represents the result of applying that rule
- For example, consider an ith rule
- R_i: if x_1 is T_{x_i} and x_2 is T_2^i then y is T_y^i

Fuzzy rule

- the firing strength or membership of the rule can be defined as

Fuzzy rule

- the firing strength or membership of the rule can be defined as
- $\alpha_i = \min\left(\mu^i_{x_1}(x_1), \mu^i_{x_2}(x_2)\right)$
- $\alpha_i = \mu^i_{x_1}(x_1)\mu^i_{x_2}(x_2)$

Fuzzy rule

- the firing strength or membership of the rule can be defined as
- $\alpha_i = \min\left(\mu^i_{x_1}(x_1), \mu^i_{x_2}(x_2)\right)$
- $\alpha_i = \mu^i_{x_1}(x_1)\mu^i_{x_2}(x_2)$
- represents fuzzy intersection with the minimum or product operators
- Each fuzzy rule yields a single number that represents the firing strength of that rule.

Fuzzy rule

- the firing strength or membership of the rule can be defined as
- $\alpha_i = \min\left(\mu^i_{x_1}(x_1), \mu^i_{x_2}(x_2)\right)$
- $\alpha_i = \mu^i_{x_1}(x_1)\mu^i_{x_2}(x_2)$
- represents fuzzy intersection with the minimum or product operators
- Each fuzzy rule yields a single number that represents the firing strength of that rule.
- The firing strength is then used to shape the output fuzzy set that represents the consequent part of the rule.

Fuzzy rule

- the firing strength or membership of the rule can be defined as
- $\alpha_i = \min\left(\mu^i_{x_1}(x_1), \mu^i_{x_2}(x_2)\right)$
- $\alpha_i = \mu^i_{x_1}(x_1)\mu^i_{x_2}(x_2)$
- represents fuzzy intersection with the minimum or product operators
- Each fuzzy rule yields a single number that represents the firing strength of that rule.
- The firing strength is then used to shape the output fuzzy set that represents the consequent part of the rule.
- The implication method is defined as the shaping of the consequent (the output fuzzy set), based on the antecedent.

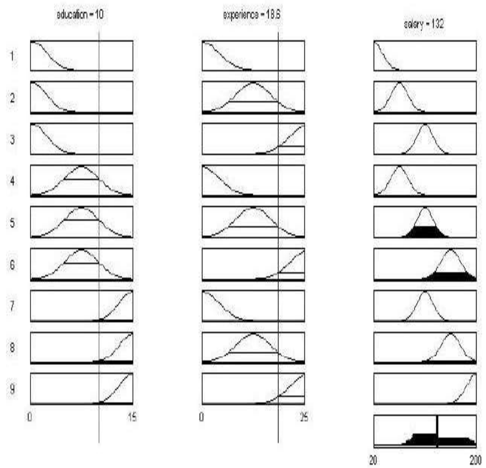
Fuzzy rule

- The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set.
- Two methods are commonly used: the minimum and the product methods, represented, respectively, by

$$
\begin{aligned}
\mu^i_y(w)' &= \min\left(\alpha_i, \mu^i_y(w)\right) \\
\mu^i_y(w)' &= \alpha_i \mu^i_y(w)
\end{aligned}
$$

  w is the variable that represents the support value of the membership function.
- For the given example, if we assume that education equals 10 years and experience equals 18.6 years, then the rules R_5 and R_6 fire.

FIS Example: Fuzzy rule



Fuzzy rule (input: education and experience and output: salary) (src:www)

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set
- The process of combining output fuzzy sets into a single set is called aggregation, a process that unifies the outputs of all the rules.

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set
- The process of combining output fuzzy sets into a single set is called aggregation, a process that unifies the outputs of all the rules.
- aggregation takes all fuzzy sets that represent the output for each rule and combines them into a single fuzzy set that is used as the input to the defuzzification process

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set
- The process of combining output fuzzy sets into a single set is called aggregation, a process that unifies the outputs of all the rules.
- aggregation takes all fuzzy sets that represent the output for each rule and combines them into a single fuzzy set that is used as the input to the defuzzification process
- Aggregation occurs only once for each output variable.

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set
- The process of combining output fuzzy sets into a single set is called aggregation, a process that unifies the outputs of all the rules.
- aggregation takes all fuzzy sets that represent the output for each rule and combines them into a single fuzzy set that is used as the input to the defuzzification process
- Aggregation occurs only once for each output variable.
- The inputs to the aggregation process are truncated or modified output fuzzy sets obtained as the output of the implication process.

Fuzzy Inference System: Aggregation

- After obtain firing strengths of the rules, need to combine the corresponding output fuzzy sets into one composite fuzzy set
- The process of combining output fuzzy sets into a single set is called aggregation, a process that unifies the outputs of all the rules.
- aggregation takes all fuzzy sets that represent the output for each rule and combines them into a single fuzzy set that is used as the input to the defuzzification process
- Aggregation occurs only once for each output variable.
- The inputs to the aggregation process are truncated or modified output fuzzy sets obtained as the output of the implication process.
- The output of the aggregation process is a single fuzzy set that represents the output variable.

Fuzzy Inference System: Aggregation

- the aggregation method is commutative, the order in which the rules are executed is not important.
- The commonly used aggregation method is the *max* method.

Fuzzy Inference System: Aggregation

- the aggregation method is commutative, the order in which the rules are executed is not important.
- The commonly used aggregation method is the *max* method.
- If we have two rules with output fuzzy sets represented by two fuzzy sets $\mu_y^1(w)$ and $\mu_y^2(w)$, then, combining the two sets, we obtain the output decision
- $\mu_y(w) = \max\left(\mu_y^1(w)\mu_y^2(w)\right)$
- it is a membership curve

Defuzzification

- The five methods may be described as follows:

Defuzzification

- The five methods may be described as follows:
- Centroid defuzzification method:

Defuzzification

- The five methods may be described as follows:
- Centroid defuzzification method: the defuzzifier determines the center of gravity (centroid) y_i' of B and uses that value as the output of the FLS.
- For a continuous aggregated fuzzy set, the centroid is given by

$$y' = \frac{\int_s y_i \mu_B(y) dy}{\int_s \mu_B(y) dy}$$

S denotes the support of $\mu_B(y)$

$$y' = \frac{\sum_{i=1}^{n} y_i \mu_B(y_i)}{\sum_{i=1}^{n} \mu_B(y_i)}$$

Defuzzification

- The five methods may be described as follows:
- Centroid defuzzification method: the defuzzifier determines the center of gravity (centroid) y_i' of B and uses that value as the output of the FLS.
- For a continuous aggregated fuzzy set, the centroid is given by

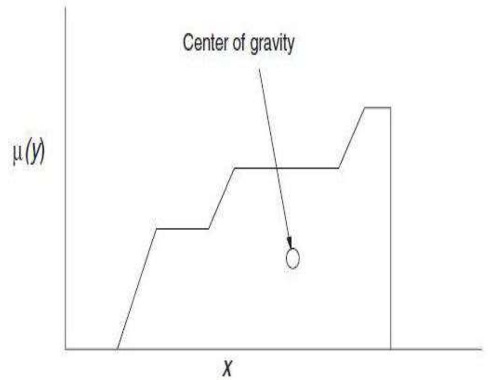
$$y' = \frac{\int_s y_i \mu_B(y) dy}{\int_s \mu_B(y) dy}$$

S denotes the support of $\mu_B(y)$

$$y' = \frac{\sum_{i=1}^{n} y_i \mu_B(y_i)}{\sum_{i=1}^{n} \mu_B(y_i)}$$

- The centroid defuzzification method finds the 'balance' point of the solution fuzzy region by calculating the weighted mean of the output fuzzy region.

Example: Defuzzification



Centroid defuzzification method (src:www)

Defuzzification

- It is the most widely used technique because, when it is used, the defuzzified values tend to move smoothly around the output fuzzy region.

Defuzzification

- It is the most widely used technique because, when it is used, the defuzzified values tend to move smoothly around the output fuzzy region.
- The technique is unique, however, and not easy to implement computationally.

Defuzzification

- It is the most widely used technique because, when it is used, the defuzzified values tend to move smoothly around the output fuzzy region.
- The technique is unique, however, and not easy to implement computationally.
- Maximum-decomposition method: the defuzzifier examines the aggregated fuzzy set and chooses that output y for which $\mu_B(y)$ is the maximum.

Defuzzification

- It is the most widely used technique because, when it is used, the defuzzified values tend to move smoothly around the output fuzzy region.
- The technique is unique, however, and not easy to implement computationally.
- Maximum-decomposition method: the defuzzifier examines the aggregated fuzzy set and chooses that output y for which $\mu_B(y)$ is the maximum.
- Unlike the centroid method, the maximum-decomposition method has some properties that are applicable to a narrower class of problems.

Defuzzification

- It is the most widely used technique because, when it is used, the defuzzified values tend to move smoothly around the output fuzzy region.
- The technique is unique, however, and not easy to implement computationally.
- Maximum-decomposition method: the defuzzifier examines the aggregated fuzzy set and chooses that output y for which $\mu_B(y)$ is the maximum.
- Unlike the centroid method, the maximum-decomposition method has some properties that are applicable to a narrower class of problems.
- The output value for this method is sensitive to a single rule that dominates the fuzzy rule set.
- Also, the output value tends to jump from one frame to the next as the shape of the fuzzy region changes.

Defuzzification

- Center of maxima: In a multimode fuzzy region, the center-of-maxima technique finds the highest plateau and then the next highest plateau.

Defuzzification

- Center of maxima: In a multimode fuzzy region, the center-of-maxima technique finds the highest plateau and then the next highest plateau.
- The midpoint between the centers of these plateaus is selected.

Defuzzification

- Center of maxima: In a multimode fuzzy region, the center-of-maxima technique finds the highest plateau and then the next highest plateau.
- The midpoint between the centers of these plateaus is selected.
- Height defuzzification: the defuzzifier first evaluates $\mu_{B_i}(y)$ at y_1' and then computes the output of the FLS, where denotes the center of gravity of fuzzy sets B_i

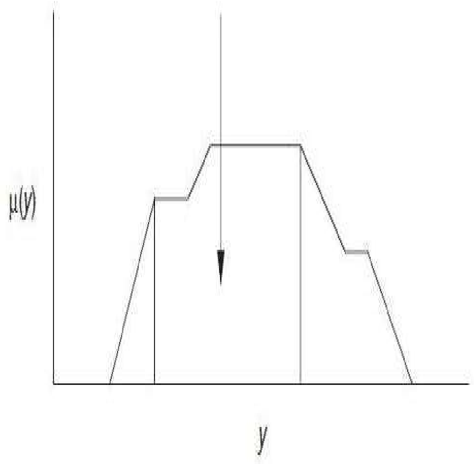
Defuzzification

- Center of maxima: In a multimode fuzzy region, the center-of-maxima technique finds the highest plateau and then the next highest plateau.
- The midpoint between the centers of these plateaus is selected.
- Height defuzzification: the defuzzifier first evaluates $\mu_{B_i}(y)$ at y_1' and then computes the output of the FLS, where denotes the center of gravity of fuzzy sets B_i
- Output y_h in this case is given by

$$y_h = \frac{\sum_{i=1}^{m} y_i' \mu_B(y_i)}{\sum_{i=1}^{m} \mu_B(y_i)}$$

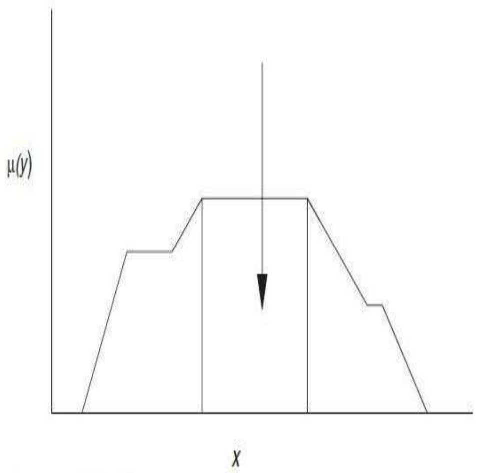
- m represents the number of output fuzzy sets obtained after implication and y_i represents the centroid of fuzzy region i.
- This technique is easy to use because the centers of gravity of commonly used membership functions are known ahead of time.

Example: Defuzzification



Defuzzification (average of maximums) (src:www)

Example: Defuzzification



Defuzzification (maximum-decomposition method) (src:www)

Summary: FIS

- The fuzzy inference process defines the mapping surface $y = f(x_1, x_2)$.

Summary: FIS

- The fuzzy inference process defines the mapping surface $y = f(x_1, x_2)$.
- The fuzzy inference process can be described completely in the five steps.

Summary: FIS

- The fuzzy inference process defines the mapping surface $y = f(x_1, x_2)$.
- The fuzzy inference process can be described completely in the five steps.
    - Fuzzify inputs
    - Apply fuzzy operators
    - Apply implication method
    - Aggregate all output fuzzy sets
    - Defuzzify

Summary: FIS

- The fuzzy inference process defines the mapping surface $y = f(x_1, x_2)$.
- The fuzzy inference process can be described completely in the five steps.
    - Fuzzify inputs
    - Apply fuzzy operators
    - Apply implication method
    - Aggregate all output fuzzy sets
    - Defuzzify
- Step 1: Fuzzy Inputs The first step is to take inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions.

Summary: FIS

- Step 2: Apply Fuzzy Operators Once the inputs have been fuzzified, knowing the degree to which each part of the antecedent has been satisfied for each rule.
- If a given rule has more than one part, the fuzzy logical operators are applied to evaluate the composite firing strength of the rule.

Summary: FIS

- Step 2: Apply Fuzzy Operators Once the inputs have been fuzzified, knowing the degree to which each part of the antecedent has been satisfied for each rule.
- If a given rule has more than one part, the fuzzy logical operators are applied to evaluate the composite firing strength of the rule.
- Step 3: Apply the Implication Method The implication method is defined as the shaping of the output membership functions on the basis of the firing strength of the rule.
- The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set. Two commonly used methods of implication are the minimum and the product.

Summary: FIS

- Step 4: Aggregate all Outputs Aggregation is a process whereby the outputs of each rule are unified.
- Aggregation occurs only once for each output variable. The input to the aggregation process is the truncated output fuzzy sets returned by the implication process for each rule.
- The output of the aggregation process is the combined output fuzzy set.

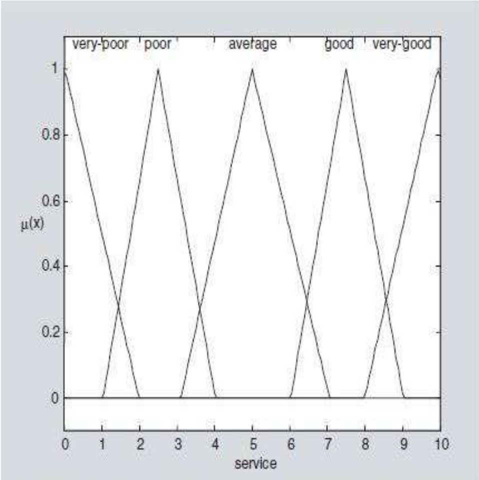
Summary: FIS

- Step 4: Aggregate all Outputs Aggregation is a process whereby the outputs of each rule are unified.
- Aggregation occurs only once for each output variable. The input to the aggregation process is the truncated output fuzzy sets returned by the implication process for each rule.
- The output of the aggregation process is the combined output fuzzy set.
- Step 5: Defuzzify The input for the defuzzification process is a fuzzy set (the aggregated output fuzzy set),
- the output of the defuzzification process is a crisp value obtained by using some defuzzification method such as the centroid, height, or maximum.

Another Example of FIS

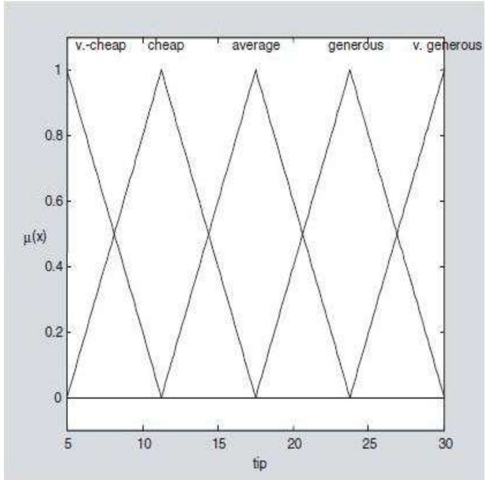
- consider a system that determines dinner in a restaurant on the basis of the service received.
- the input x denotes the quality of the service, which is represented by a number between 0 and 20, where 20 designates *verygood* and 0 *verypoor*.
- The input x is represented by the term set $\{verypoor, poor, average, good, verygood\}$.
- The output y represents the *tip*, which varies between 5 and 30 percent, and is given by the term set $\{verycheap, cheap, average, generous, verygenerous\}$.

Example: Service input



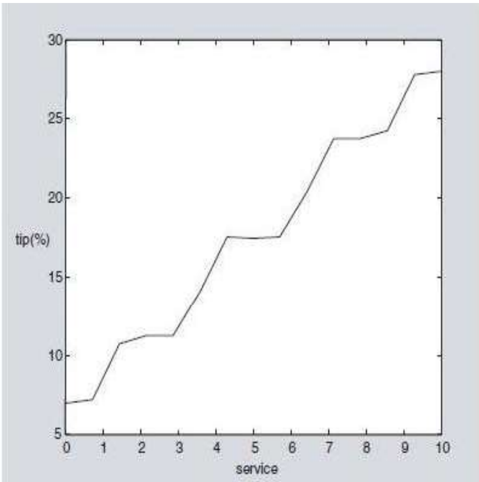
Input fuzzy sets for service (src:www)

Example: Tip output



Output fuzzy sets (tip) (src:www)

Example: Mapping surface



Mapping surface (service) (src:www)

Example: FIS

- Five rules that define mapping $y = f(x)$:

Example: FIS

- Five rules that define mapping $y = f(x)$:
- R_1: if service is very poor, then tip is very cheap

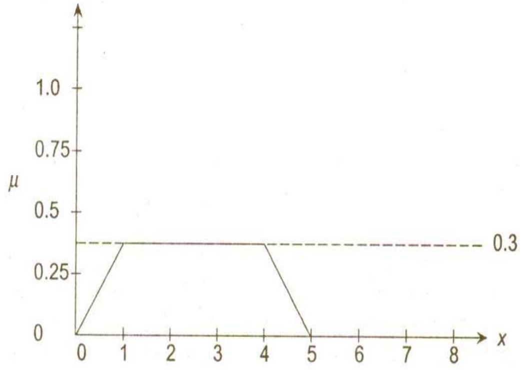
Example: FIS

- Five rules that define mapping $y = f(x)$:
- R_1: if service is very poor, then tip is very cheap
- R_2: if service is poor, then tip is cheap
- R_3: if service is average, then tip is average
- R_4: if service is good, then tip is generous
- R_5: if service is very good, then tip is very generous
- The mapping function $y = f(x)$ depends on the number of input fuzzy sets and their shapes.

Example: FIS

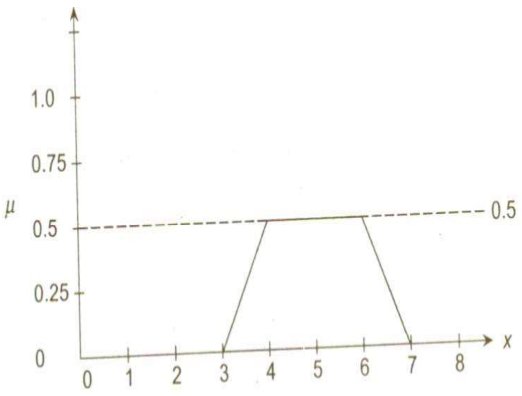
- Five rules that define mapping $y = f(x)$:
- R_1: if service is very poor, then tip is very cheap
- R_2: if service is poor, then tip is cheap
- R_3: if service is average, then tip is average
- R_4: if service is good, then tip is generous
- R_5: if service is very good, then tip is very generous
- The mapping function $y = f(x)$ depends on the number of input fuzzy sets and their shapes.
- increase the overlap between the input fuzzy sets, the mapping function becomes smoother.

Example of Aggregation



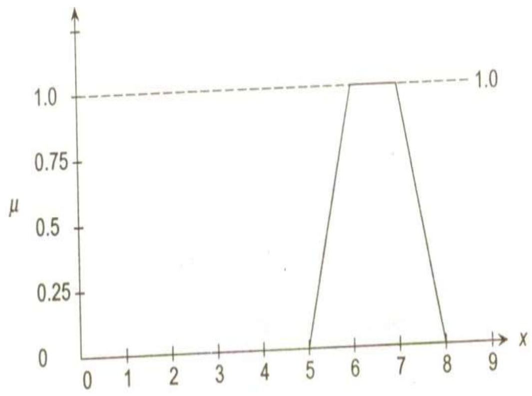
A_1 fuzzy set (src:NFG book)

Example of Aggregation



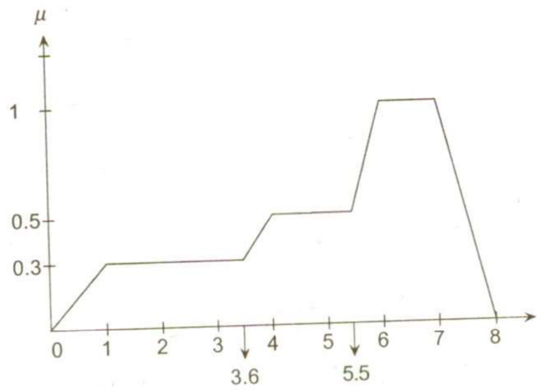
A_2 fuzzy set (src:NFG book)

Example of Aggregation



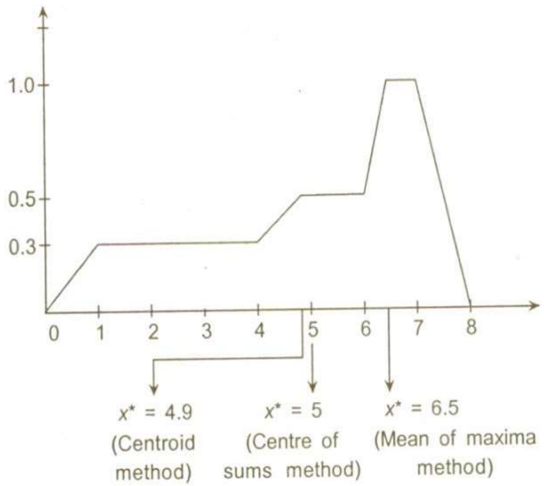
A_3 fuzzy set (src:NFG book)

Example of Aggregation



Aggregation of fuzzy sets A_1, A_2 and A_3 (src:NFG book)

Example of Defuzzification



Defuzzified outputs of the aggregate of A_1, A_2 and A_3 (src:NFG book)

Example of Airconditioner Controller

- dial to control the flow of warm/hot or cool/cold air

Example of Airconditioner Controller

- dial to control the flow of warm/hot or cool/cold air
- thermometer to measure room temperature

Example of Airconditioner Controller

- dial to control the flow of warm/hot or cool/cold air
- thermometer to measure room temperature
- difference in temperature between room temperature and desired temperature

Example of Airconditioner Controller

- dial to control the flow of warm/hot or cool/cold air
- thermometer to measure room temperature
- difference in temperature between room temperature and desired temperature
- dial turned positive to supply warm/hot air
- dial turned negative to supply cool/cold air
- dial set to zero no air is supplied from airconditioner

Example of Airconditioner Controller

- dial to control the flow of warm/hot or cool/cold air
- thermometer to measure room temperature
- difference in temperature between room temperature and desired temperature
- dial turned positive to supply warm/hot air
- dial turned negative to supply cool/cold air
- dial set to zero no air is supplied from airconditioner
- determine to what extent the dial should be turned

Example of Airconditioner Controller: Fuzzy rule

- Few terms

approximately zero (ZE)	positively large (PL)
negative large (NL)	negative medium (NM)
positively small (PS)	negatively small (NS)
positive medium (PM)	

Example of Airconditioner Controller: Fuzzy rule

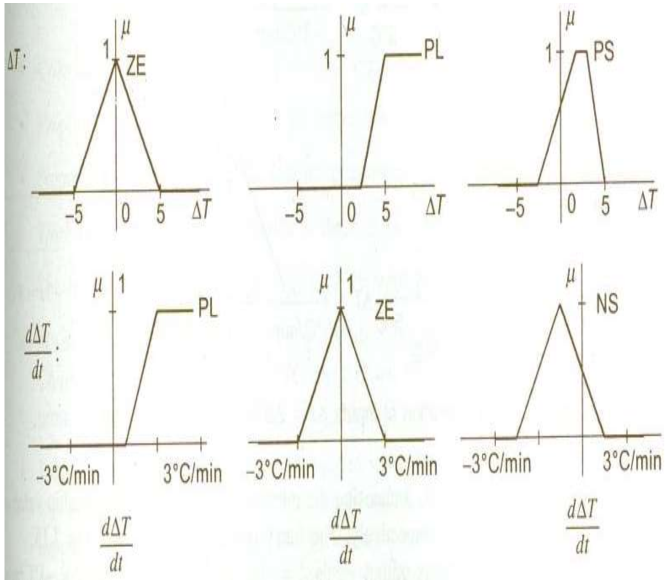
- Few terms

approximately zero (ZE)	positively large (PL)
negative large (NL)	negative medium (NM)
positively small (PS)	negatively small (NS)
positive medium (PM)	

- if $\triangle T$ is ZE and $\frac{d\triangle T}{dt}$ is PL then dial should be NL
- if $\triangle T$ is PL and $\frac{d\triangle T}{dt}$ is ZE then dial should be NM
- if $\triangle T$ is PS and $\frac{d\triangle T}{dt}$ is NS then dial should be ZE

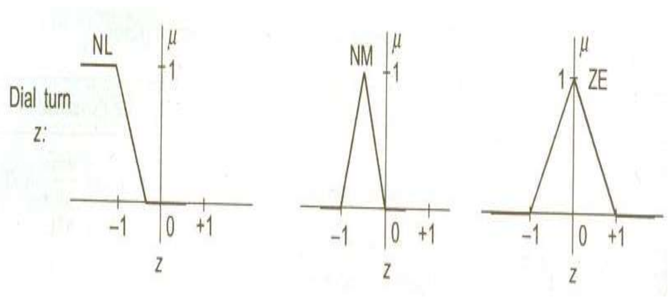
M. A. Zaveri, SVNIT, Surat Soft Computing Techniques 3 April 2011 74 / 157

Example of Airconditioner control system



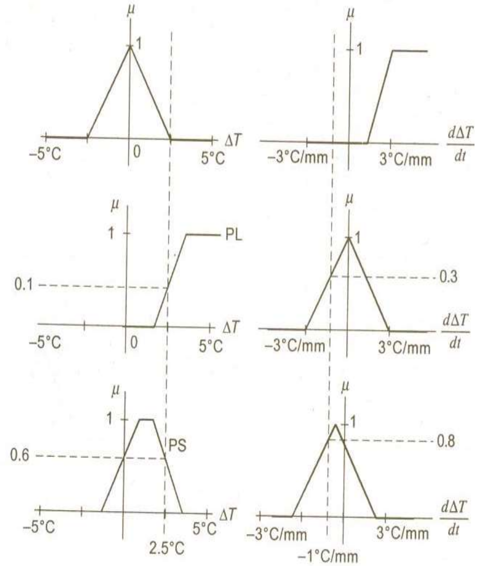
Fuzzy sets for system inputs (src:NFG book)

Example of Airconditioner control system



Fuzzy sets for system output (src:NFG book)

Example of Airconditioner control system

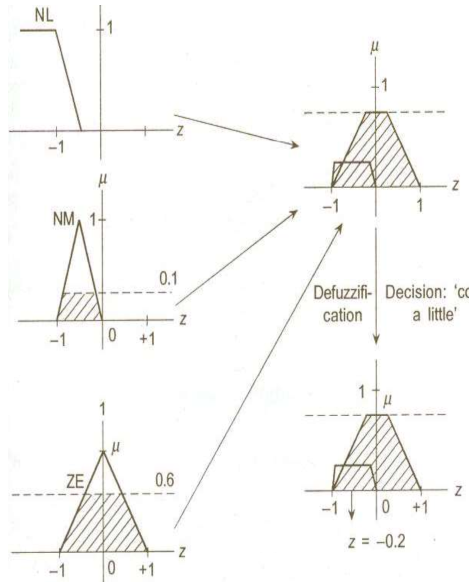


Fuzzification of inputs (src:NFG book)

Example of Airconditioner Controller

- Rules 1, 2 and 3 choosing the minimum of fuzzy membership value of antecedents are 0, 0.1, and 0.6
- Defuzzification of fuzzy output $z = -0.2$ for $\triangle T = 2.5°C$ and $\frac{d\triangle T}{dt} = -1°C/min$
- dial to be turned in negative direction

Example of Airconditioner control system



Defuzzification of fuzzy outputs for z (turn of the dial) (src:NFG book)

Example of Fuzzy cruise controller

- maintain a vehicle at a desired speed

Example of Fuzzy cruise controller

- maintain a vehicle at a desired speed
- two fuzzy inputs: speed difference and acceleration

Example of Fuzzy cruise controller

- maintain a vehicle at a desired speed
- two fuzzy inputs: speed difference and acceleration
- fuzzy output: throttle control

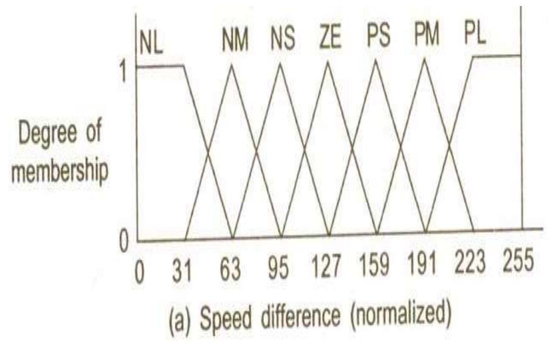
Example of Fuzzy cruise controller

- maintain a vehicle at a desired speed
- two fuzzy inputs: speed difference and acceleration
- fuzzy output: throttle control
- Fuzzy rule base

Example of Fuzzy cruise controller

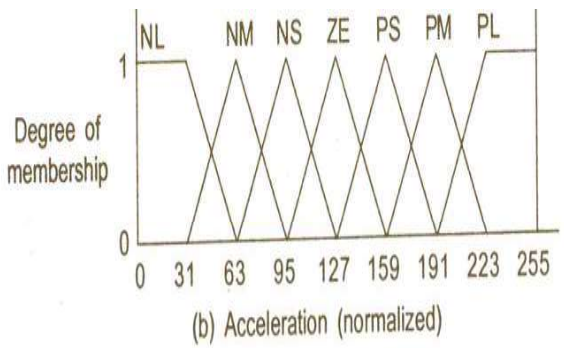
- maintain a vehicle at a desired speed
- two fuzzy inputs: speed difference and acceleration
- fuzzy output: throttle control
- Fuzzy rule base
- R_1: if speed difference is NL and acceleration is ZE then throttle control is PL
- R_2: if speed difference is ZE and acceleration is NL then throttle control is PL ....

Example of Fuzzy cruise control



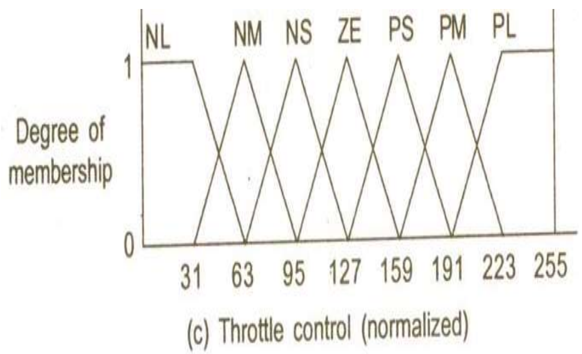
Speed difference: fuzzy input (normalized) (src:NFG book)

Example of Fuzzy cruise control



Acceleration: fuzzy input (normalized) (src:NFG book)

Example of Fuzzy cruise control



Throttle control: fuzzy output (normalized) (src:NFG book)