Implementation of a Class Library to Model Fuzzy Sets and Operators Using SAS/AF®

Bogdan Lelutiу - Romanian National Commission for Statistics

1. Base concepts

Definition 1.1.
Be $X$ a classical set, called also support set.
A fuzzy set $\tilde{A}$ over $X$ is characterized by a membership function $\mu_i(x)$ which attaches to each element from $X$ a real number in the range $[0,1]$. The value of the function $\mu_i(x)$ in the point $x$ designates the membership degree of the element $x$ to the set $\tilde{A}$.

One notation for a fuzzy set is the enumeration of the binary tuples of the form $(x, \mu_i(x))$:
$$\tilde{A} = \{(x_1, \mu_i(x_1)), (x_2, \mu_i(x_2)), \ldots, (x_n, \mu_i(x_n))\}$$

Definition 1.2.
Two fuzzy sets $\tilde{A}$ and $\tilde{B}$ are equal if and only if their membership functions are equal.
$$\tilde{A} = \tilde{B} : \Leftrightarrow \mu_i(x) = \mu_j(x) \quad \forall x \in X$$

Definition 1.3.
Be $\tilde{A}$ and $\tilde{B}$ two fuzzy sets over $X$.
$\tilde{A}$ is a subset of $\tilde{B}$ ($\tilde{A} \leq \tilde{B}$) if the membership degree $\mu_i(x)$ does not exceed the membership degree $\mu_j(x)$.
$$\tilde{A} \leq \tilde{B} : \Leftrightarrow \mu_i(x) \leq \mu_j(x) \quad \forall x \in X$$

2. Membership functions

Membership functions representation is very important in automate modeling of fuzzy sets. The representation types of membership functions depend on the element number of the support set.

2.1. Parametrized representations

These representations are used when the support set contains a very large number of elements or models a continuous magnitude. The representations use functions that change their characteristics by setting one or more parameters values. These functions approximate the membership function.
Triangular membership functions

Definition 2.1.1.
A triangular function $f_{\text{triangle}}$ is completely defined by five parameters $(m, \alpha, b, y_{\min}, y_{\max})$, $m \in \mathbb{R}, \alpha, \beta \in \mathbb{R}^+$ and $y_{\min}, y_{\max} \in [0,1]$, as follows:

$$f_{\text{triangle}}(x, m, \alpha, \beta, y_{\min}, y_{\max}) = \begin{cases} y_{\min} & x \leq m_a, x \geq m_b \\ y_{\min} + \left( d_y(x - m_a) \right) / \alpha & m_a < x \leq m_b \\ y_{\min} + \left( d_y(m_b - x) \right) / \beta & m < x < m_b \end{cases}$$

\[ \forall x \in \mathbb{R} \]

$m_a = m - \alpha$

where: $m_b = m + \beta$

$d_y = y_{\max} - y_{\min}$

\[ \text{fig. 2.1} \]
**Trapezoidal membership functions**

**Definition 2.1.2**
A trapezoidal function $f_{\text{trapez}}$ is completely defined by six parameters $(m_1, m_2, \alpha, \beta, y_{\text{min}}, y_{\text{max}})$ where $m_1, m_2 \in \mathbb{R}, m_1 < m_2, \alpha, \beta \in \mathbb{R}$ and $y_{\text{min}}, y_{\text{max}} \in [0,1]$ as follows:

$$f_{\text{trapez}}(x, m_1, m_2, \alpha, \beta, y_{\text{min}}, y_{\text{max}}):= \begin{cases} y_{\text{min}} & x \leq m_1, x \geq m_2 \\ y_{\text{max}} & m_1 \leq x \leq m_2 \\ y_{\text{min}} + (d_y(x - m_1))/\alpha & m_1 < x < m_2 \\ y_{\text{min}} + (d_y(m_2 - x))/\beta & m_2 < x < m_2 \end{cases}$$

$\forall x \in \mathbb{R}$

where:
\[ m_a = m_1 - \alpha \]
\[ m_b = m_2 + \beta \]
\[ d_y = y_{\text{max}} - y_{\text{min}} \]

**S,Z,PI membership functions**

**Definition 2.1.3.**
A S function is defined by two parameters $a, \gamma \in \mathbb{R}$ as follows:

$$S(x, a, \gamma):= \begin{cases} 0 & x \leq (a-\gamma) \\ 2((x-a+\gamma)/2\gamma)^2 & (a-\gamma) < x \leq a \\ 1-2((a-x+\gamma)/2\gamma)^2 & a < x < (a+\gamma) \\ 1 & x \geq (a+\gamma) \end{cases}$$

$\forall x \in \mathbb{R}$
Definition 2.1.4.
A $Z$ function is defined by two parameters $a, \gamma \in R$ as follows:
\[ Z(x,a,\gamma) = 1 - S(x,a,\gamma) \quad \forall x \in R \]

Definition 2.1.5.
A $\Pi$ function is defined by two parameters $a, \gamma \in R$ as follows:
\[
\Pi(x,a,\gamma) = \begin{cases} 
S(x,a-\gamma/2,\gamma/2) & x < a \\
Z(x,a+\gamma/2,\gamma/2) & x \geq a 
\end{cases} \quad \forall x \in R
\]

$S$ functions are used for modeling when strengthening a property implies a higher membership degree of an object to a fuzzy set.
For example if we intend to formulate the membership to the "old population" then the membership function could be chosen $S(x,50,15)$, where $x$ is the age.
measured in years. So, \( a = 50, \gamma = 15, a - \gamma = 35, a + \gamma = 65 \). Under 35 years a person can not be considered "old" \( (\mu(x) = 0) \) and over 65 years a person is certly "old" \( (\mu(x) = 1) \).

### 2.2. Discrete representations

If the support set is finite (it is the case for natural objects) then the membership degree for each element can be explicitly given.

From an implementation point of view, the pairs set \( (x, \mu(x)) \) can be represented through list type or array type structures. The two types are exemplified presenting two representation methods namely the general fuzzy sets method and the fuzzy arrays method.

#### 2.2.1. General fuzzy sets

**Definition 2.2.1.1.**

A general fuzzy set \( \tilde{G} \) is represented by a set of binary tuples

\[
\tilde{G} = \{(x_i, y_i), ..., (x_n, y_n)\}, \quad \text{where} \quad (x_i, y_i), i = 1, ..., n \quad \text{with} \quad x_i < x_j \quad \forall \ i < j \quad \text{and} \quad 0 \leq y_i \leq 1 \quad \forall i \in \{1...n\} \quad \text{are points in which the growth of the membership function is modified (between two neighboured points the function is approximated by a straight line)}.
\]

We can calculate for \( x_i \leq x \leq x_{i+1} \) the corresponding \( y_i \) value by the formula:

\[
y = y_{i+1} - \frac{(y_{i+1} - y_i)(x_{i+1} - x)}{(x_{i+1} - x_i)}
\]

Representation precision grows with the number of points.

#### 2.2.2. Fuzzy arrays

The base domain \([x_{\min}, x_{\max}]\) is divided in ranges equally distanced and the membership degrees in the points defining the limits of the range are memorised.

**Definition 2.2.2.1**

Be \( x_{\min} < x_{\max} \in \mathbb{R} \).

A function \( f: [x_{\min}, x_{\max}] \rightarrow \mathbb{R} \) is called a fuzzy array if exists a division \( x_{\min} = t_0 < t_1 < ... < t_n = x_{\max} \) of the range \([x_{\min}, x_{\max}]\) and the constants \( d_k, c_0, c_1, ..., c_n \in \mathbb{R} \) so that:

- \( d_k = (x_{\max} - x_{\min})/n \)
- \( t_k = t_{k-1} + d_k \quad \forall k, 1 \leq k \leq n \)
- \( f(x) = c_0 \quad \forall x \in [t_0, t_0 + dx/2) \)
- \( f(x) = c_k \quad \forall x \in [t_k - dx/2, t_k + dx/2], \forall k \quad 1 \leq k \leq n-1 \)
- \( f(x) = c_n \quad \forall x \in [t_n - dx/2, t_n] \)
The calculation of the function value in a point $x_0$, where $x_{\min} \leq x_0 \leq x_{\max}$ provides the membership degree of the limit of the closest range to $x_0$.

Membership degrees are faster calculated through this method although approximation precision improv is less efficient.

Fuzzy vectors are used for representing the transformations of fuzzy sets in simpler forms (for example partial results created by fuzzy sets composition).

3. Fuzzy sets operations

Definition 3.1.

Be $A$, $B$ two fuzzy sets defined by the membership functions $\mu_A(x)$ and $\mu_B(x)$.

The intersection, union of the fuzzy sets $A$, $B$ and the complement of the fuzzy set $\tilde{A}$ are fuzzy sets defined by the following membership functions:

\[
\mu_A \cap \mu_B(x) = \min(\mu_A(x), \mu_B(x)) \quad \forall x \in X
\]
\[
\mu_A \cup \mu_B(x) = \max(\mu_A(x), \mu_B(x)) \quad \forall x \in X
\]
\[
\mu_{\tilde{A}}(x) = 1 - \mu_A(x) \quad \forall x \in X
\]

Definition 3.2.

Be $\tilde{A}_1, \ldots, \tilde{A}_n$ $n$ fuzzy sets over the corresponding support sets $X_1, \ldots, X_n$.

The fuzzy cartesian product $\tilde{A}_1 \otimes \ldots \otimes \tilde{A}_n$ is the fuzzy set having the membership function defined by:

\[
\mu(\tilde{A}_1, \ldots, \tilde{A}_n)(x) = \min(\mu_{\tilde{A}_1}(x_1))_{x_1 \in X_1} \quad \forall x \in X_1 \times \ldots \times X_n
\]

4. Operators

Besides the named max and min operators also other operators are used in fuzzy sets composition. They can be divided in t-norms (intersection), t-conorms (union) and intermediar operators between the two types.

Each class has both parametrised and not parametrised operators.

Definition 4.1.

A t-norm is an application $\cdot : [0,1] \times [0,1] \to [0,1]$ having the properties:
1. $t(0,0) = 0, t(x,1) = t(1,x) = x$
2. $t(u,v) \leq t(w,z)$ \hspace{1cm} $u \leq w, v \leq z$ \hspace{1cm} (monotony)
3. $t(x,y) = t(y,x)$ \hspace{1cm} (symetry)
4. $t(x,t(y,z)) = t(t(x,y),z)$ \hspace{1cm} (associativity)

Definition 4.2.
A t-conorm (or s-norm) is an application $s:[0,1] \times [0,1] \rightarrow [0,1]$ having the following properties:
1. $s(1,1) = 1; s(x,0) = s(0,x) = x$
2. $s(u,v) \leq s(w,z)$ \hspace{1cm} $u \leq w, v \leq z$ \hspace{1cm} (monotony)
3. $s(x,y) = s(y,x)$ \hspace{1cm} (symetry)
4. $s(x,s(y,z)) = s(s(x,y),z)$ \hspace{1cm} (associativity)

5. Linguistic variables

Linguistic variables are a special form of fuzzy sets which have values that represent words or expressions aimed to model approximate knowledge. This concept permits approximate characterisation of the either too complex or too weak structured phenomena that can not be associated through classical quantitative methods.

Definition 5.1. (Zadeh)
A linguistic variable is characterised by a five elements set $(X, T(x), U, G, \tilde{M})$. $X$ represents the name of the linguistic variable, $T(x)$ the term set(set of names) that form the values of the linguistic variable, $U$ the support set. $G$ is a syntactic rule(usually in the form of a context free grammar) having the role of generating a name $X$ for the variable $x$ from the name set $T(x)$. $\tilde{M}$ is a semantic rule that associates a significance $\tilde{M}(X)$ to a given $X$, where $\tilde{M}(X)$ is a fuzzy subset over $V$.

A name $X$ generated through $G$ is called a term. A term representing a set of words can be atomic if the words set can be seen as a whole or compound, if it contains more atomic terms.

$T$ can be expressed as:
$T = X_1 + X_2 + ... + X_n$ \hspace{1cm} where $X_1, X_2, ..., X_n$ are terms.

Example:
$T(\text{old}) = \text{old} + \text{very old} + \text{not at all} + \text{more or less old} + \text{young enough} + \text{not to old and not to young} + ...$, each term being the name of a fuzzy set over the support set $V = [0, 100]$. 

571
The linguistic variable "old" can be defined by the function:

\[ M(old) = \{(u, \mu_{old}(u)) / 0 \leq u \leq 100\}, \text{ choosing } \mu_{old}(u) = \begin{cases} 0, & 0 \leq u \leq 50 \\ \left(1 + \frac{(4 - 50)}{5}\right)^{-1}, & 50 \leq u \leq 100 \end{cases} \]

Zadeh shows that the construction rules for significant terms can make the object of a grammar.

5.1. Modificators

The modificators represent the possibility of fuzzy set mathematical transformation in order to verbalise the result of the modelisation. Modificators are unary operators over fuzzy sets.

Example:
For the "very" and "more or less" modificators the following functions can be used (Zadeh):

\[ \mu_{very}(x) = \mu(x)^2 \quad \forall x \in X \]
\[ \mu_{more\ or\ less}(x) = \sqrt{\mu(x)} \quad \forall x \in X \]

5.2. Linguistic approximation

The goal of linguistic approximation is to interpret a fuzzy set in the terms of colloquial speaking.

The problem can be defined as an application of the set \( S \) (the fuzzy-sets set over an support \( X \)) in a set of designation elements \( L \) generated by the use of a vocabulary \( V \).

The main methods for generating the set \( L \) are:
- the "test-fit" method - is based on testing the euclidian distance between the fuzzy set to be interpreted and the set of designators seen as fuzzy sets respecting:

\[ \text{dist}(\tilde{A}, \tilde{B}) = \sqrt{\sum_{x \in X} (\mu_{\tilde{A}}(x) - \mu_{\tilde{B}}(x))^2} \]

the designator for which the value of the euclidian distance to the interpreted reference is minimal is selected.
- the succesive approximation method - the search space is limited to the designators delimited by two atomic terms.
- the decomposition method - the decomposition of the fuzzy set to be interpreted in component fuzzy sets that should match approximatively with the designators from
the terms set is tried; all the designators are build through the combination of the
modifiers with the atomic terms.

The "test-fit" method is the simplest to implement but implies time consuming
search when the number of designators grows. Search is speeded up by using the
second method. The third method permits the tuning of the speed/precision rapport
during interpretation.

6. Fuzzy systems

6.1. Fuzzy systems structure

The general structure of a fuzzy regulators implies at least four modules:
- fuzzification module
- rule base
- inference engine
- defuzzification module

Elaboration of a fuzzy system will concern, consequently, four steps:
- establishing of input and output variables
- building a rule base
- establishing a inference strategy
- output fuzzy measures computation

6.2. Fuzzification

This module of the fuzzy controller transforms the input variables in linguistic
variables following fuzzy logic principles. The corresponding fuzzy sets are established
for each variable. A verbal characterisation of the variables is also given. The membership
function is established for each fuzzy set.

6.3. Rule base

The knowledge about a modeled process are represented by a rule base.
One possible way of representing knowledge is the production rules formalism.
The rule base contains in this case IF THEN rules together with the clauses and
the logical connectors present in the premises and conclusions. Each clause is of the
form "VAR is atribut" where VAR is interpreted as a linguistic variable. An atribut of a
linguistic variable is either a fuzzy set or a fuzzy set modified through one or more
modifiers.

The description of the rule base is made usually by a context free grammar.
Each rule has a confidence factor (a value between 0 and 1) attached that reflects the confidence in the validity of the rule.

6.4 Defuzzyfication

The result of the inference process in a fuzzy system is a fuzzy output set. It can be represented by the membership function, characterised verbally through linguistic approximation or by a numeric value determined in a defuzzyfication process.

If the fuzzy set is convex, the weight centre method is used. It is based on the calculation of the weight centre of a continuous function \( f \) on an interval \([a, b]\) through the formula:

\[
G = \frac{\int_a^b x f(x) dx}{\int_a^b f(x) dx}
\]

An automated method of determining this value is possible to be applied representing the output fuzzy set as a general fuzzy set. In this case the function \( f: [a, b] \rightarrow [0, 1] \) is linear on portions \( f_i(x) = mx + c_i \) \( m_i, c_i \in \mathbb{R} \) \( 1 \leq i \leq n \)

where \( f_i: [t_{i-1}, t_i] \rightarrow [0, 1] \) and \( a = t_0 < t_1 < ... < t_{n-1} < t_n = b \)

So \( \int_a^b f(x) dx = \sum_{i=1}^n \int_{t_{i-1}}^{t_i} f_i(x) dx \) and consequently taking into account that

\[
\int_a^b f_i(x) dx = F(b) - F(a)
\]

the formula obtained is

\[
G = \frac{\sum_{i=1}^n (1/3m_i(t_i^3 - t_{i-1}^3) + 1/2c_i(t_i^2 - t_{i-1}^2))/((1/2m_i(t_i^3 - t_{i-1}^3) + c_i(t_i - t_{i-1}))}
\]

If the fuzzy set is not convex, the defuzzified value can be obtained by choosing the maximal value of the membership function.

7. System design

7.1 System goals

The general structure of the system is that of a fuzzy controller. Values of the linguistic variables are scalars. Output fuzzy sets can have several interpretations:
- graphical representation
- defuzzyfication through weight centre method
- defuzzyfication through maximum method
- linguistic approximation

The usage interface disposes of different methods of membership function
representation(S, Z, PI, LR fuzzy numbers, triangle and trapez functions)

Different operators used in the inference process can be selected (t-norms, s-norms, intermediar operators).

The rule base should be extensible and reusable.

7.2 Implementation decisions

The SAS System was choosen for implementation from several reasons:
- the support for object oriented design offered by SAS/AF
- cross platform GUI portability
- the quality of the tools offered for human-machine dialog
- connection possibility to different input and output data sources (relational, ODBC, text files)
- further processing and interpretation of the datasets using other SAS analyse tools

An object oriented approach with a C++ MSDOS implementation is given in [1]. It runs only in conventional memory and is not linked to other traditional data sources. On the other side the implementation is a general purpose one.

The main goal of the SAS implementation was to provide a set of alternative or complementary modeling tools aimed to be used together with other SAS procedures in fields of statistical modelisation where the lack of exhaustive information makes other classic models unappliable. Also another inference approach is intended, trying to take advantage of Windows and Unix virtual memory management.

7.3 Object oriented design

The idea behind object oriented design is the trial of real world entities and facts modelisation through an abstractisation process. The transpose of these into programs should be as much as possible one to one.

Therefore, fuzzy concepts can find a natural representation in the data structures (the classes) that are implementing them.

B. Stroustrup (quoted also in [1]) defines in the process of object oriented design 6 steps:
- class identification
- class operations specification
- class dependencies specification
- interfaces specification
- class hierarchy reorganisation
- model usage

7.3.1 Class identification
The concept of a fuzzy set can be identified by a class. This class models a fuzzy set \( \tilde{A} \) over \( X \). Its main role is to provide a value of the membership function to \( \tilde{A} \) for an element \( x \in X \). Special fuzzy classes can be implemented as subclasses of an abstract base class (fig. 7a).

Another concept, that of a linguistic variable can be interpreted as a class. This class has the role to represent, by behalf of fuzzy sets, linguistic formulated knowledge. The existing relationship between the linguistic variables class and the fuzzy sets class is a "has a" type relationship (a linguistic variable "contains" one or more fuzzy sets).

The different types of operators can be modeled by derivation from a base class that provides the fundamental operations. The 2 main operators subclasses are the class of the parametrised operators and the class of the not-parametrised operators. Between these classes and the base operators class exists a "is a" type relationship (fig. 7b). t-norm, s-norm or intermediar operators can be seen as parametrised or not-parametrised operators instances.

Other identified classes are the modificators and rules classes.

7.3.2 Class operations specification

**Fuzzy sets class**

A fuzzy set has a name, a minimal and a maximal membership function value. A fuzzy set exists in the context of a linguistic variable so it has to contain a field to provide the name of this linguistic variable. Finally, in the class are also stored informations concerning the drawing of the membership function chart (color, thickness, etc).

The most important operation of the fuzzy set is the appartenance function value calculation. Another operation concerns the drawing of the membership function chart. Other operations describe the way in which the fuzzy set elements can be modified.

**Linguistic variables class**

A linguistic variable has a name, a base domain represented by 2 numeric fields and a current value. It contains also a list of fuzzy set names corresponding to the terms. Fields containing information used for the graphical representation are also stored in the class.

Part of the operations linked with linguistic variables refer to the drawing of the coordinate systems for the membership functions of the component fuzzy sets. Operations for editing the facts implied in the inference process are also provided by the class.

**Operators**
The operators base class contains fields for the name of the operator and its type (t-norm, s-norm, etc.). The derived classes of the parametrised and not-parametrised operators contain data fields the operator number and the operator definition function.

The main role of an operator is to bound one or more arguments to a new value. The operations of this class are concerned with this task.

7.3.3 Class dependencies

**Fuzzy sets**

The existent relationships between the different types of membership functions (S, Z, , LR, , triangle) are modeled through inheritance.

The subclass FuzzyArray has no structural links to the other subclasses and therefore is derived directly from the base class.

**Linguistic variables**

The linguistic variable contains a list of fuzzy set names that model the "has a" relationship to them.

**Operators**

The common elements of the ParamOperator and NonParamOperator subclasses are implemented in the base Operator class where these classes are derived from.
FuzzySet

- FuzzySType
- FuzzyZType
- FuzzyPiType

- FuzzyLR
- FuzzyTrapez
- FuzzyTriangle

fig 7.a

Operator

- NonParamOperator
- ParamOperator

fig 7.b
# Bibliography

1. A. Mayer  
   B. Mehler  
   A. Schlindwein  
   R. Wolke  
   **Fuzzy Logic**  
   Springer Verlag, Heidelberg, 1993

2. G. Böhme  
   **Fuzzy Logik**  
   Springer Verlag, Heidelberg, 1993

3. A. Kaufmann  
   Theorie des sous-ensembles flouées  

4. SAS Institute Inc  
   SAS/AF Software Frame Entry  
   SAS Campus Drive, Cary, 1993

5. SAS Institute Inc  
   SAS Screen Control Language  
   SAS Campus Drive, Cary, 1993

6. SAS Institute Inc  
   SAS Language  
   SAS Campus Drive, Cary, 1993

7. SAS Institute Inc  
   SAS Procedure Guide  
   SAS Campus Drive, Cary, 1993

Acknowledgements:

1. SAS/AF is registered trademarks of SAS Institute Inc., Cary, North Carolina, USA
2. Windows is registered trademarks of Microsoft Corp.
3. UNIX is a trademark of AT&T

The author can be contacted at:  
Bogdan Leluti  
Romanian National Commission for Statistics  
16, Libertatii Avenue  
70542 sector 5  
Bucharest  
ROMANIA  
Tel: +40-1-311.23.89