TECHNICAL UNIVERSITY OF CRETE

SCHOOL OF PRODUCTION ENGINEERING AND MANAGEMENT

Applications of Intelligent Control and Optimization Techniques in the field of Credit Insurance

Engineering Diploma Thesis

by Konstantina K. Ainatzoglou

Chania, July 2020

Advisory Committee

Georgios E. Stavroulakis (Supervisor)

Professor, School of Production Engineering and Management,

Technical University of Crete, Chania, Greece

Georgios K. Tairidis (Co-supervisor)

Post-doc. Researcher, School of Production Engineering and Management,

Technical University of Crete, Chania, Greece

Lecturer, Hellenic Mediterranean University, Heraklion, Greece

Examination Committee Members

Constantin Zopounidis

Professor, School of Production Engineering and Management,

Technical University of Crete, Chania, Greece

Maria Bakatsaki

Laboratory Teaching Staff, School of Production Engineering and Management,

Technical University of Crete, Chania, Greece

Acknowledgements

There are some people who contributed to the completion of this thesis, whom I would like to acknowledge.

Above any other, I would like to express my absolute and sincere gratitude to my advisor, Dr. Georgios K. Tairidis for his selfless support and continuous devotion to helping his students. From the conception of the topic till the correction of every detail he has been supporting me with great patience and enthusiasm. His persistence in helping students at all times and his ability of inspiring us, reflect his love about his profession. His selfless actions helped me conceive the topic of this thesis and urged me to have higher expectations of myself and my work. He had a major impact on my personality and the way in which I will be facing my career from now on. I wholeheartedly thank him for everything he has done for me.

I would like to thank Professor Georgios Stavroulakis and Professor Constantin Zopounidis for offering valuable knowledge regarding their respective fields. I would also like to thank Mrs. Maria Bakatsaki for her presence at the presentation of my thesis and her help.

I would also like to thank Mrs. Ioanna Kouidourmazi, who provided me with all necessary details, along with Mrs. Alexandra Ainatzoglou for their unconditional love and support at all times.

Table of Contents

Abstract

In the present thesis, problems in the field of insurance will be addressed through intelligent control and optimization techniques. More specifically, a way of calculating the price of insurance policies that has to be paid by a prospective client of an insurance company will be suggested. This model will be created and implemented with the use of fuzzy systems. Moreover, a neurofuzzy control system will be created in order to establish the correct operation of the system. The training data that will be used will derive from real anonymous insurance policies. Additional algorithms and nature inspired optimization techniques such as the genetic algorithm will be used wherever required.

1. Introduction and theoretical background

The current paper is an attempt to research the application of tools offered by the fuzzy and the adaptive neurofuzzy inference systems on the domain of credit insurance. The purpose of this paper is to explore the effectiveness of this alternative approach in order to automate the processes of calculating prices of insurance credit policies and of approving the credit coverage of prospective clients.

There are many papers in literature which have previously addressed some applications of fuzzy logic in the field of insurance (Calibo et al. 2017), (Shapiro, 2005), (Sokolovska, 2017), (Yazdani and Kwasnicka, 2012). The first article that has made use of fuzzy logic in insurance was the one of DeWit (1982). One of the scopes of the formerly mentioned paper was to quantify fuzziness in the field of underwriting. Since then, there have been numerous other attempts that examine how fuzzy logic can be involved in the field of insurance. There are papers which examine the theoretical dimension of how fuzzy inference systems could be used in order to improve the processes of risk assessment and risk decision making (Shapiro, 2007). There have been efforts of evaluating credit risk using neurofuzzy logic (Sreekantha and Kulkarni, 2010) and attempts to develop fuzzy logic distribution for soft data and variables used for the corporate client credit risk assessment (Brkic et al., 2017).

The purpose of this paper is to address topics of credit insurance from two different scopes. From the perspective of the credit insurance brokerage, a fuzzy inference system will be developed in order to calculate the price of a credit policy that has to be submitted by the company who has requested an insurance coverage. Using anonymous credit insurance policies as an input in an adaptive neurofuzzy inference system, new rules and results will be produced for the calculation of prices in credit insurance policies. Furthermore, the results produced by the rules created in the fuzzy inference system (direct problem) will be compared to the results produced by the adaptive neurofuzzy inference system (inverse problem).

In order to facilitate the comprehension of the total paper, an initial analysis regarding the basic concepts and definitions that govern this particular thesis will be provided. In this paper, concepts stemming from different fields of professional activity will be combined. Definitions regarding the field of credit insurance as well as the field of intelligent control systems will be examined. Furthermore, a theoretical background of the optimization techniques that will be used for correction of potential deviations from the expected results will also be presented.

Trade credit insurance: “Trade credit insurance protects manufacturers, traders and service providers against losses from non-payment of a commercial trade debt. If a buyer does not pay (often due to bankruptcy or insolvency) or pays very late, the trade credit insurance policy will pay out a percentage of the outstanding debt. The primary function of trade credit insurance is to protect sellers against buyers that do not or cannot pay”. (Moorcraft, 2018).

Control system: A system is anything that has receives inputs and produces outputs. A system that has to be controlled called a plant. A control system is a system that can transform the inputs to the plant in order to produce a desired output. From a more technical perspective, a control system is an interconnection of components which form a system configuration that is able to produce a desired system response. (Dorf and Bishop, 2011).

Fuzzy logic: “The basic idea of fuzzy logic is to associate a number with each object indicating the degree to which it belongs to a particular class of objects” (Pfeifer, 2013).

Fuzzy inference system (FIS): “A nonlinear mapping that derives its output based on fuzzy reasoning and a set of fuzzy if-then rules. The domain and range of the mapping could be fuzzy sets or points in multidimensional spaces.” (Jang and Sun, 1997).

Adaptive neurofuzzy inference system (ANFIS): “There is a class of adaptive networks that are functionally equivalent to fuzzy inference systems. The architecture of these networks is referred to as ANFIS, which stands for adaptive network-based fuzzy inference system or semantically equivalently, adaptive neurofuzzy inference fuzzy inference system.” (Jang and Sun, 1997).

Genetic algorithms: A genetic algorithm is a search technique that is used in computing for the calculation of true values or their approximation of solutions for optimizations and search problems. They belong to the category of global search heuristics. These algorithms are a specific class of evolutionary algorithms that use operations inspired by the field of evolutionary biology such as inheritance, selection and crossover (Michalewicz, 1996).

2. Fuzzy inference systems

2.1. The basic idea of fuzzy systems

Fuzzy sets were initially introduced by Zadeh (1965) for the representation and management of data that was not in a precise format, but rather fuzzy. The basic idea of fuzzy logic is to provide a specific inference format that allows approximate human reasoning skills to be used in a knowledge-based system. Fuzzy logic can provide a mathematical base in order to capture the uncertainty involved in the human cognitive process, such as reasoning and decision making. The previous approach to knowledge modelling failed to embrace the concept of fuzziness. This was the main reason why techniques such as the first order logic and the classical probability theory cannot deal with the representation and modelling of commonsense knowledge. The necessity of addressing problems of uncertainty and verbal imprecision led to the adoption of the fuzzy logic concepts.

2.2. Important qualities of fuzzy logic

Some of the qualities that describe fuzzy logic relate to exact reasoning faced as a limiting case of the broader approximate reasoning and scaling. Another important parameter is the fact that knowledge is conceived as a set of elastic variables and that inference is viewed as a function that propagates elastic constraints. Any logical system can be modelled with fuzzy logic.

There are two characteristics that make fuzzy systems preferable in certain applications. Fuzzy systems are applicable for uncertain or approximate reasoning, particularly in case that the system has a mathematical problem that is difficult to construct. Another valuable characteristic of fuzzy logic is that it enables the process of decision making with estimated values under incomplete or uncertain data.

2.2.1 Fuzzy sets and membership functions

In crisp sets, an element belongs to a set only when it takes the value 1. In any other case, the element takes the value 0 and is not part of the set. In fuzzy sets each element can take values from a range with some participation (membership) rate of the element in the set. The higher the value, the greater the participation of the element within the set. This set is called fuzzy set, while the function is called membership function. A graphical comparison between a membership function and a crisp set is shown below.

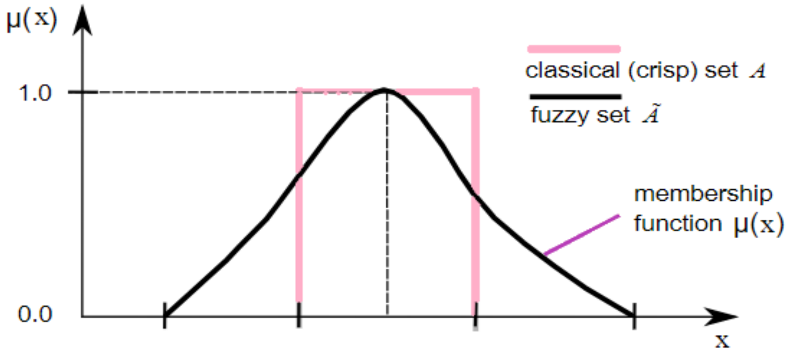


Figure 1. A fuzzy membership function in comparison to a crisp set (https://commons.wikimedia.org/wiki/File:Fuzzy_crisp.svg)

A fuzzy set A is referred to as a triangular fuzzy number with peak (or center) $a > 0$, left width $\alpha > 0$ and right width $\beta > 0$ if it has the following form (Fullér, 1995):

$$A(t) = 1 - (\alpha - t)/\alpha \, , if \; a - \alpha \le t \le \alpha$$

$$A(t) = 1 - (t - \alpha)/\beta \, , if \; a \le t \le a + \beta$$

$$A(t) = 0 \;\; otherwise$$

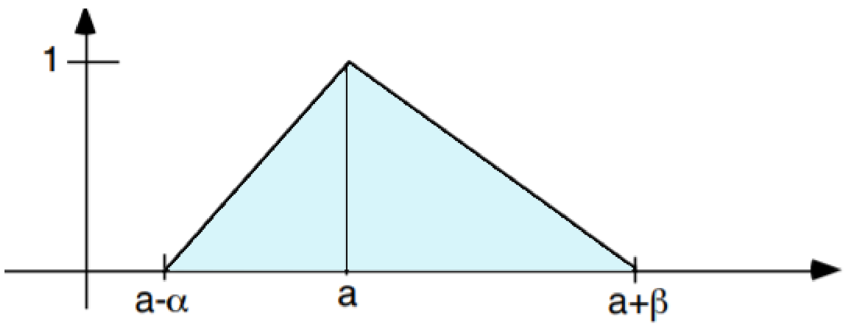


Figure 2. Triangular fuzzy number (Robert Fullér 1995)

A fuzzy set is referred to as a trapezoidal fuzzy number with tolerance interval $[a, b]$, left width α and right width β if it has the following form:

$$A(t) = 1 - (\alpha - t)/\alpha \, , if \; a - \alpha \le t \le \alpha$$
$$A(t) = 1 \;\; , if \; a \le t \le b$$
$$A(t) = 1 - (t - \alpha)/\beta \, , if \; a \le t \le b + \beta$$
$$A(t) = 0 \;\; otherwise$$

and the notation $A = (a, b, \alpha, \beta)$ is used in order to describe this pattern.

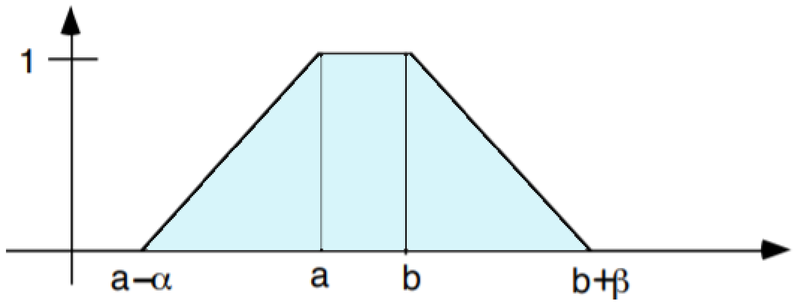


Figure 3. Trapezoidal fuzzy number (Robert Fullér 1995)

A fuzzy subset A of a set X can be considered as a set of pairs in a certain order, each with the first element coming from X and the second element coming from the interval [0,1], with exactly one ordered pair present for each element that belongs to X. This process creates a mapping μ_A between the elements of X and the values that derive from the interval [0,1]. A zero value denotes complete non-membership, a value of one denotes complete membership and the values in between denote intermediate percentages of membership. The set X can be described as the universe of discourse for the fuzzy subset A. The mapping μ_A represents as function called "the membership function of A". Consequently, the definitions "membership function" and "fuzzy subset" can be used interchangeably. The following definitions will provide clarifications regarding the previously analysed terms (Fullér, 1995).

Assuming that X is a nonempty set, a fuzzy set A in X is characterized by the membership function:

$$\mu_A : X \to [0,1]$$

and $\mu_A(x)$ is interpreted as the percentage of membership x in fuzzy set A for each given x that belongs to X.

The percentage of fuzziness that characterizes a fuzzy set is represented with the use of its membership functions. These functions can be depicted either in a numeric or in a graphical way. There are numerous forms in which a membership function can be graphically described. The most popular ones are the following (Tairidis, 2016):

a. Triangular membership functions
b. Trapezoidal membership functions

c. Bell membership functions

d. Gaussian membership functions

e. Sigmoid membership functions

f. Polynomial membership functions

2.2.2 Operations on fuzzy sets

The classical operations from the theory of ordinary sets can also be applied to fuzzy sets. The same symbols used in the ordinary set theory will be used when operations are extended to fuzzy sets. Let a nonempty crisp X set (a crisp set is part of the distinct set theory that employs bi-valued logic) with its fuzzy subsets A and B:

- The intersection of A and B is described as:

$$(A \cap B)(t) = min\{A(t), B(t)\}, \forall t \in X$$

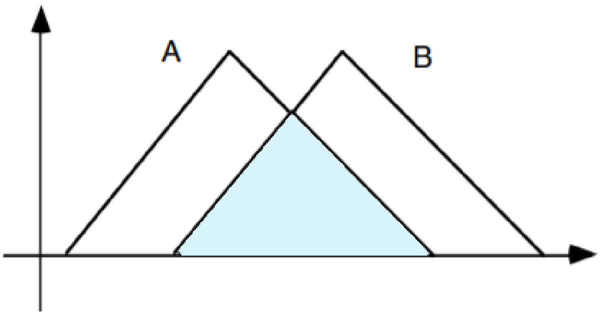


Figure 4. Intersection of two triangular fuzzy numbers (Robert Fullér, 1995)

- The union of A and B is described as:

$$(A \cup B) = max\{A(t), B(t)\}, \forall t \in X$$

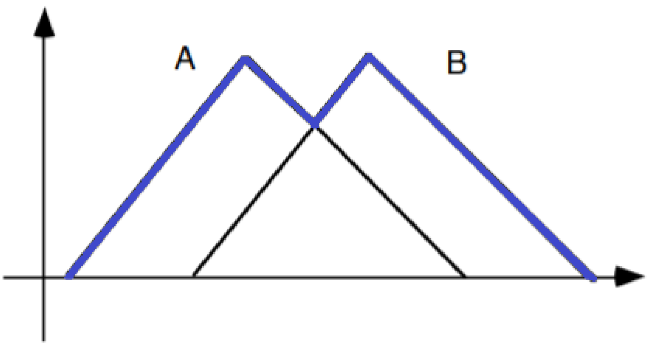


Figure 5. Union of two triangular fuzzy numbers (Robert Fullér, 1995)

- The complement of a fuzzy set A is described as:

$$(-A)(t) = 1 - A(t)$$

2.2.3 The extension principle

In order to make use of fuzzy concepts and relations in an intelligent system, arithmetic operations with these fuzzy quantities should be applied. More specifically, the operations of addition, subtraction, multiplication and division with fuzzy quantities should be performed. In this thesis, the first two operations will be mathematically defined. The extension principle is an essential concept from fuzzy set theory that needs to be analyzed before proceeding to the examination of arithmetic operations. This principle enables the extension of any point operation to operations between fuzzy sets.

The extension principle can be explained as follows:

If X and Y are nonempty crisp sets and f is a mapping from X and g is a mapping from X to Y:

$$g: X \rightarrow Y$$

such that for each $x \in X$, $g(x) = y \in Y$. Let that A is a fuzzy subset of X, with the use of the extension principle, g(A) can be defined as a fuzzy subset of Y such that:

$$g(A)(y) = sup_{\ x \in g^{-1}(y)} A(x), if\ g^{-1}(y) \neq 0$$

$$g(A)(y) = 0\ otherwise$$

where $g^{-1}(y) = \{x \in X \mid f(x) = y\}$,

and $sup\ (A(x)) = \{x \in X \mid A(x) > 0\}$

After the mathematical definition of the extension principle, the operations of extended addition and extended subtraction can be analyzed.

The operation of extended addition can be described as:

Let $g: X \times X \rightarrow X$ be defined as $g(x_1, x_2) = x_1 + x_2$. If A_1 and A_2 are fuzzy subsets belonging to X, then according to the extension principle:

$$g(A1, A2)(y) = sup_{x_1 - x_2 = y} \min\{(A_1(x_1), A_2(x_2)\} \; or \; g(A_1, A_2) = A_1 + A_2$$

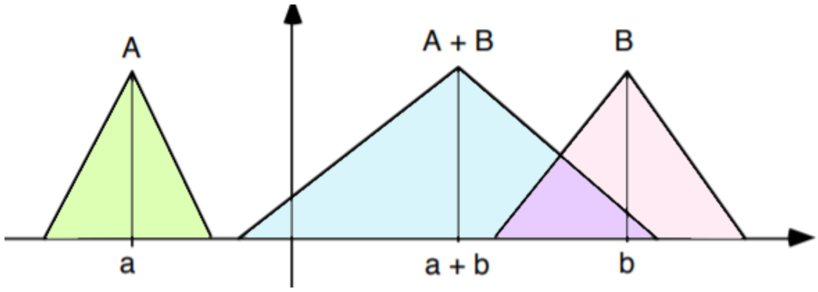


Figure 6. Addition of triangular fuzzy numbers (Robert Fullér, 1995)

The operation of extended subtraction can be described as:

Let g: X × X → X be defined as $g(x_1, x_2) = x_1 - x_2$. If A_1 and A_2 are fuzzy subsets belonging to X, then according to the extension principle:

$$g(A_1, A_2)(y) = sup_{x_1 - x_2 = y}\{min\,(A_1(x_1), A_2(x_2)\} \; or \; g(A_1, A_2) = A_1 - A_2$$

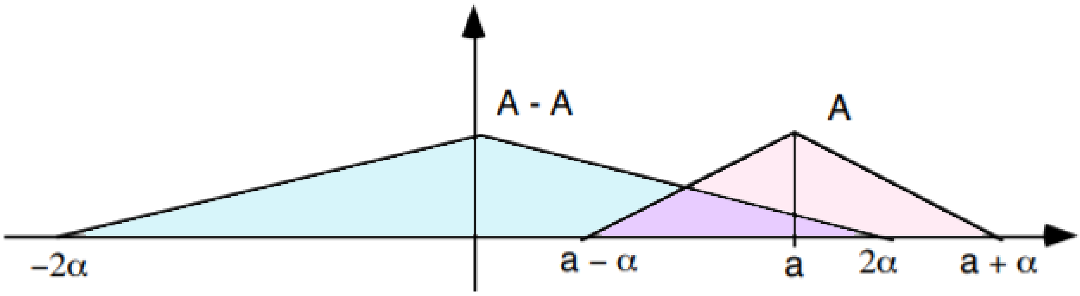


Figure 7. **Subtraction** of **triangular** fuzzy **numbers** (A-A) (Robert Fullér, 1995)

2.2.4. Linguistic Variables

The use of fuzzy sets enables a systematic and organized management of vague and imprecise concepts. More specifically, linguistic variables can be represented by fuzzy sets. A linguistic variable is a variable whose value is a fuzzy number or a variable which is described in lexical terms.

A linguistic variable can be denoted as $(z, T(z), U, G, M)$,

where:

z is the variable, T(z) is the term set of z or alternatively, the set of names of lexical values of z with each value being a fuzzy number defined on the universe U.G is a rule that generates the names of the values of z, and M is a rule that associates each value to its meaning.

For example, if price is interpreted as a linguistic variable, z=price, then its term set T (price) can be described as:

T={very low, low, medium, high, very high,....}

where each term in T(price) is described by a fuzzy set in a universe of interval U=[0,100]. Then the following lexical variables could denote the following arithmetic intervals:

- Very low = “a price below 20”
- Low = “a price between 20 and 40”
- Medium= “a price between 40 and 60”
- High = “a price between 60 and 80”
- Very high = “a price between 80 and 100”

These descriptions can be described as fuzzy sets whose membership functions are shown in the following figure.

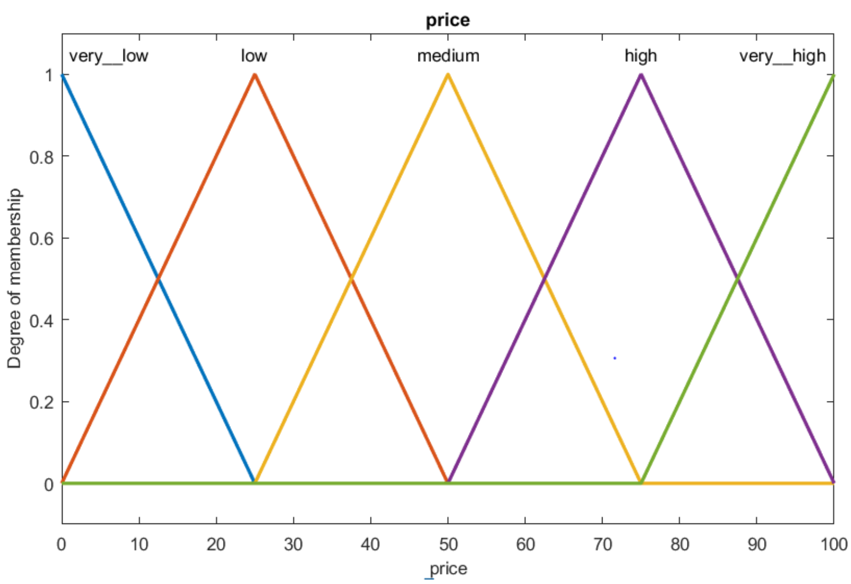


Figure 8. Values of linguistic variable “price”

2.3 The theory of approximate reasoning

The theory of approximate reasoning enables modelling a reasoning that involves imprecision and uncertainty of information. This theory describes premises as statements assigning fuzzy sets as values to variables. Let two interactive variables x∈ X and y∈ Y with their causal relationship defined,

$$y = f\ (x)$$

An obvious example of inference that can be made is:

Assumption [$y = f(x)$] +Fact [$x = x'$] → Consequence: [$y = f(x')$]

Zadeh has created a group of translation rules that enable the representation of a number of commonly used lexical statements that refer to propositions in a certain language. Some of these translation rules are analysed in this thesis.

- Entailment rule:
  - {z is A} and {A ⊂ B}→ x is B
  - {George is very smart} + {very smart ⊂ smart} → George is smart
- Conjunction rule:
  - {z is A} and {z is B} → z is A∩B
  - {temperature is not very high} + {temperature is not very low} → {temperature is not very high and not very low}
- Disjunction rule:
  - {x is A} or {x is B} → {x is A ∪ B}
  - {temperature is high} or {temperature is low} →{temperature is high or low}
- Negation:
  - not{x is A} →{ x is !A}
  - not{x is high}→{x is not high}
- The Modus Ponens inference rule:
  - Statement {if a then b} and Fact {a} → consequence {b}
- Basic property:
  - {if x is A then y is B} + {x is A} → {y is B}
  - {if speed is high then price is high} + {speed is high} →{price is high}
- Total indeterminance:

- {if x is A then y is B} + {x is not A}→y is unknown
- {if speed is high then price is high} + {speed is not high} → {price is unknown}

2.4. Fuzzy logic controllers

A feedback controller within a fuzzy system checks if the response of the output deriving from the fuzzy system is the expected one. The process of maintaining the value of the real output close to the value of the reference input (desired output) despite any deviances and noise that the system parameters may create, is referred to as regulation. The output that derives from the controller, which is used as input for the system, is called control action.

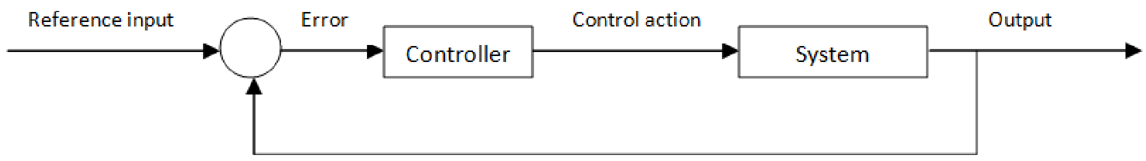


Figure 9. Basic feedback control system

In a fuzzy logic controller, the fuzzy system acts dynamically based on a set of verbal rules that are created by an expert. The knowledge provided by the expert is based on IF-THEN statements. These statements are called conditional statements. A fuzzy control rule is a conditional statement in which the "IF" part refers to a condition in the application domain of the fuzzy control system and the "THEN" part is a control action within the fuzzy control system.

In order for the fuzzy rule-based system to accept the form of these conditions in a form of fuzzy sets, crisp inputs should be fuzzified. The output of a fuzzy system will always be a fuzzy set which has to be defuzzified. The structure of the fuzzy logic control system is the following:

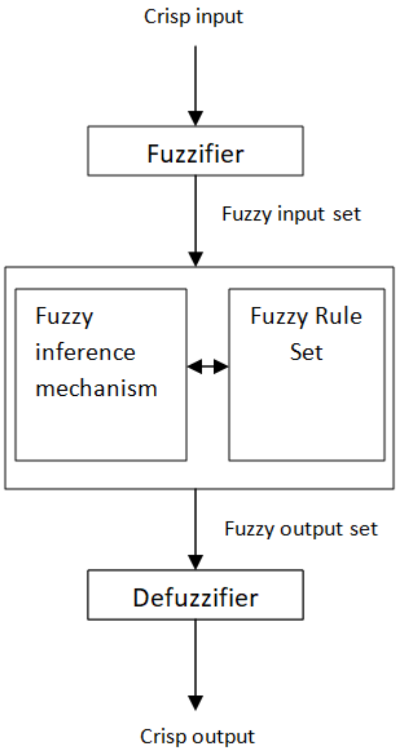


Figure 10. Fuzzy Logic Controller

2.4.1. Fuzzification

Fuzzification is one of the most essential parts of fuzzy theory. It is the process of transforming a crisp quantity into a fuzzy one, which is depicted through membership functions. From a practical point of view, application errors might occur. These errors might have an impact on the reduction of data accuracy. This reduction can be also depicted through the membership functions. For the process of fine tuning of the membership functions various techniques can be used. Some methods that could be used for fine tuning are the following:

- Intuition
- Inference
- Optimization (e.g. Genetic Algorithms)
- Deep learning (e.g. Neural Networks)

2.4.2. Defuzzification

The process of defuzzification enables the translation of the fuzzy output set produced by the fuzzy logic rule-based system. There is a great number of methods used in order to defuzzify the fuzzy output set.

The most common techniques are presented below:

1. Maximum membership principle
2. Centroid
3. Bisector
4. Middle or mean of maximum (MOM)
5. Smallest of maximum (SOM)
6. Largest of maximum (LOM)
7. Centre of sums
8. Centre of largest area
9. Weighted average (WTAVER)

The choice of the most suitable defuzzification method depends on the requirements of the researcher and the parameters of the problem. It is possible that two methods give identical or completely different results.

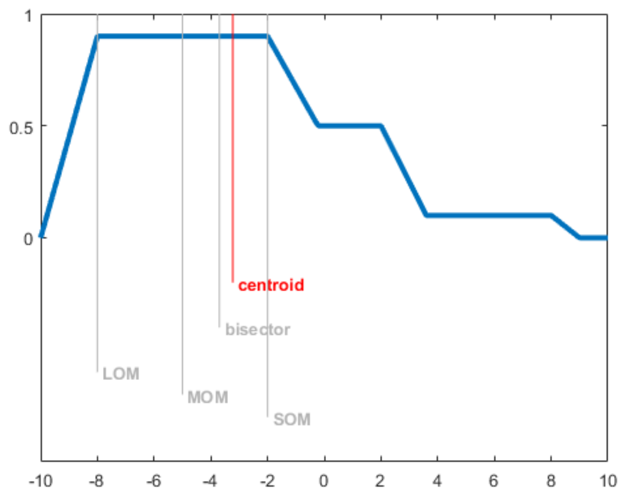


Figure 11. Illustration of defuzzification methods

(https://www.mathworks.com/help/fuzzy/defuzzification-methods.html)

2.5. Adaptive neurofuzzy inference systems

The basic idea of a fuzzy inference system is the construction of membership functions that represent the inputs and outputs of the system based on a set of verbal rules in order to support a decision making process. The choice of membership functions is either arbitrary or based on experience. The structure of rules should be predefined and based on the knowledge of an expert (Tairidis 2016).

Fuzzy inference systems produce satisfactory solutions when applied to control. However there are some limitations, such as the absence of systematic framework or the method of transforming the human cognitive experience into a set of if-then rules, which hinder the total efficacy of the system and may be responsible for deviations between the results produced and the expected results. It is a common phenomenon that when the control mechanism is built, the system designer cannot make a decision about the form and other qualities of membership functions or the structure of the rules of the system taking into consideration just the available data that derives from the expert (Tairidis 2016), (Tairidis and Stavroulakis 2019).

2.5.1.How Adaptive Neurofuzzy Systems Work

Adaptive Neurofuzzy Inference Systems (ANFIS) belong to the most commonly used adaptive fuzzy systems. The structure of ANFIS is based on a fuzzy inference system which is implemented inside the framework of adaptive neural networks.

ANFIS consists of a set of fuzzy rules which in contrast to conventional fuzzy systems, are local mappings instead of global ones (Jang & Sun, 1995). These mappings enable the minimal disturbance principle, according to which the adaptation should diminish the output error for the current training pattern but also reduce as much as possible the disturbance to response already learned. (Widrow & Lehr,1990).

During the construction of a fuzzy inference system, one of the most basic processes followed was fuzzy modelling. Neurofuzzy modelling is the process of applying learning methods, developed using the neural network theory, to fuzzy inference systems. Back-propagation neural networks are commonly used for the definition of parameters of an adaptive fuzzy inference system.

In the case of a hybrid learning procedure, the control model could create an input and output mapping depending on both human knowledge, just like in fuzzy systems and input-output data combinations. However, there is also the option to construct a control model with input-output mapping even when the human knowledge is not available. In this case, the initial parameters are given intuitively and the fuzzy rules could be constructed using a learning process to estimate the expected performance. Consequently, instead of selecting the parameters of the controller (membership functions, rules, etc.) arbitrarily, an automated process can produce membership functions for the fuzzy variables based on the available training dataset. A set of rules or other parameters can also be included and the controller can be trained in order to function under different circumstances.

2.5.2. ANFIS in MATLAB environment

An ANFIS can be created with the use of the fuzzy logic toolbox in MATLAB. It is a training routine for the creation of adaptive Sugeno-type neurofuzzy inference systems.

With ANFIS, a fuzzy inference system can be structured through the use of an input-output training dataset. One way of setting up the parameters of the system is the use of the back-propagation algorithm, either alone or combined with the least squares algorithm (hybrid method). This tuning technique enables fuzzy systems to learn from the data they are modelling. The learning method is similar to the one used in neural networks.

The modelling process begins with the introduction of a parameterized model and the collection and application of a training dataset. This data is used by the fuzzy system for automatic approximation of its parameters until an error criterion is fulfilled.

The training dataset should be carefully selected. As far as simple models are concerned, the more training data available for the learning process, the more accurate the approximation of parameters. For systems that contain noise, model validation is an essential process.

Model validation can be accomplished by using a second dataset, the so called testing data. During the process of model validation, new inputs coming from the testing data and were not included in the training phase, are introduced to the system in order to check whether the system produces accurate results in terms of output prediction. This is an essential step in order to ensure that the model does not overfit the training data. Apart from this function, testing data enables the process of checking whether the constructed fuzzy inference system is robust and produces proper results in different conditions.

2.5.3. Training of ANFIS through MATLAB

At the beginning of the process lies the collection of training data with input and output data that refers to the system to be modelled. This dataset must be in the form of arrays organized as column vectors with the output data in the last column. The training dataset could be loaded from a file or from MATLAB workspace. The initial fuzzy inference system variables should be parameterized arbitrarily or, in case human expertise cannot define their form, automatically by clustering on the data. More specifically, the structure of the model can be either loaded by a pre-existing Sugeno fuzzy inference system structure or produced through a partitioning technique, such as grid partitioning or subtractive clustering.

The first method produces a single-output Sugeno fuzzy inference system through applying grid partitioning on the data. The second method generates an initial form of model for ANFIS training after applying subtractive clustering on the training dataset.

A typical grid partition in a two dimensional input space can produce satisfactory results when a small number of membership functions describes each input. For larger numbers of inputs, the grid partition method may produce unexpected results. For instance, a fuzzy model with seven inputs and three membership functions for each input would create a set of 2187 if-then rules, which is admittedly large. This issue is called the curse of dimensionality and can be partially solved with the use of other partition techniques.

Subtractive clustering on the contrary is the appropriate partition technique in case that the number of clusters there should be at each input is unknown. This algorithm is fast in terms of estimating the number of clusters and the cluster centres in the training dataset. These approximations can enable the initialization of optimization-based clustering methods and model identification methods like ANFIS.

2.5.4. The training process through ANFIS

After loading the training dataset and creating the initial FIS structure, the system is ready for the training process. As previously stated, the back-propagation and the hybrid method are the two methods used as far optimization is concerned. Both methods are used for enabling the training of the membership function parameters. These parameters are formed in a way that approximates the training dataset.

The back-propagation method belongs to the gradient descend methods. It calculates the derivative of the function of error, taking into consideration all the neural network weights. The derivative calculated by the process is used as input to the optimization method which uses it in order to update the values of weights and minimize the produced error. The method of least squares calculates an approximate solution in overdetermined systems. According to this process, the overall solution produced should minimize the sum of errors computed for every equation.

The hybrid method uses back propagation to calculate the parameters that refer to membership functions and least squares method to approximate the parameters that refer to the output membership functions.

3. Optimization processes

3.1. The basic idea of Genetic Algorithms

The idea of genetic algorithms stemmed from the need for optimizing an already existing set of solutions that were initially generated by some other preceding procedure. This category of algorithms imitates the procedure of natural selection and evolution. Their principle is to mimic the biological process through which new and enhanced populations of offspring are developed during the evolutionary process. Unlike most heuristic algorithms, genetic algorithms make use of an already existing group of solutions that are named individuals. A genetic algorithm functions as a stochastic repetitive process, which produces a population of the same size in each iteration. Every such iteration is called a generation. The main characteristic of a genetic algorithm is the matching of two solutions for the production of a new solution. This procedure, which is called crossover, is one of the basic functions of the algorithm. Another important operation which is called mutation is also necessary for the better exploration of the solution space. In order to create a new solution these two operators are needed; the binary operator named crossover and the unit operator that is called mutation. The crossover accepts two individuals who are named parents as intakes. It produces two new individuals that are called offspring through exchange of parts of the two initial parents. Every individual in the produced population represents a potential solution (Marinakis, 2019).

3.2. How genetic algorithms work

In order to describe the way in which genetic algorithms work, terms borrowed from the field of biology can be used. The formerly mentioned definition of an individual can be described using a single chromosome. Each chromosome consists of a set of genes which follow a certain pattern of sequence. These genes, which encode and describe one or more qualities of the organism, can be found in particular positions on the sequence of the chromosome. The characteristics that can be attributed to an individual are a result of the information encoded into the respective gene which is responsible for the particular trait.

The aggregated amount of genetic material existing within an organism is called a genome and a particular group of genes within the genome is called a genotype. The genotype is responsible for the revelation of physical and mental traits of an organism.

This set of physical and mental characteristics is called the phenotype of the organism. As a result, the genotype refers to the set of genes that contain information which creates the phenotype and the phenotype is the physical and mental description of the encoded information hidden in the genotype. The process of natural selection is based on criteria that refer to the phenotype, since the latter expresses external characteristics which reveal how the organism interacts with the environment. Nevertheless, the genotype is indirectly affected by this process of natural selection, since the most adaptive to the environment organisms are the ones which eventually survive.

A chromosome can be modified if the sequence of genes within it is altered. A new group of the same initial genes create a new chromosome. There are two basic functions taking place within organisms; reproduction and mutation. The first refers to the procedure of genetic material exchange between two parts of the organism (parents) for the reproduction and the creation of an offspring. The chromosomes are fragmented into a number of parts and then fragments of the first chromosome are mixed with fragments of the second. This procedure is referred to as crossover. Each parent contributes to the reproduction with the use of one simple chromosome called a gamete, so that the offspring will have the same amount of genetic material as each one of its parents alone. Mutation is described as an "erroneous" function within the reproduction of genetic material during cell division. In case that an error occurs after the crossover, the altered chromosome might be inherited to the next generation. Mutations can result from genetic or environmental factors.

The main purposes of genetic algorithms are the preservation of the amount of genetic information within the surviving population and the evolution of this population in order to survive. This process of evolution is based on the selection of individuals which are best adapted to the environment and on the process of updating the genetic material through recombination (Tairidis, 2016).

3.3 Modelling of the genetic algorithm

In a classic genetic algorithm each gene is represented by independent variables which consist of a certain number of parameters. Every variable is a gene encoded in binary format and consists of a digit or a set of digits. The chromosome in this case is a row that is formed by the respective genes. When independent variables take certain values within the chromosome, a solution occurs. The process of coding a certain system as a chromosome produces a genotype and the external expression when the genotype is executed is the phenotype. There is a fitness function that indicates how well the chromosome is performing and it depends on the value of the objective function that is produced by the particular chromosome. The following step is a process of selecting the most well performing chromosomes and combing them with other respectively “strong” chromosomes. The scope of optimizing the fitness function is to adapt the characteristics of individuals to the external environment which operates according to the principle of natural selection.

The function of the crossover operator is based on the exchange of parts between two solutions that belong to the same generation. An example of the crossover implementation is fragmenting two chromosomes in the same randomly selected point and then exchanging their second fragments with each other. The result of the exchange will be the production of two new solutions, which are called offspring. The mutation in this case of a binary modelled chromosome occurs through altering the value of a randomly selected gene within the chromosome. For example, in the binary model of a chromosome, if the selected part of a gene had the value 0, it would be converted to 1.

The mutation and crossover operators search within a set of already existing solutions and a parameter called selection operator exploits the information within the population. The process implemented by the first two operators is called exploration, whereas the process performed by the latter operator is called exploitation. The mechanism of selection is responsible for maintaining the balance between these two processes. The term “selective pressure” refers to increasing the intensity of exploitation and the cost of exploration property. Increasing the value of this parameter helps the algorithm converge faster into a solution. However “selective pressure” might trap the algorithm into a local optimum.

In every genetic algorithm, the optimization process begins with a randomly created number of P solutions in the form of binary strings. These solutions represent the chromosomes. After the initial generation of the population, each individual of the population represents a potential solution of the problem. Each individual has a “fitness value” indicating how suitable it is for solving the given problem. Pairs of “probably suitable” solutions, or individuals, are chosen for the reproduction of offspring. The two parents combine their strings in order to reproduce their offspring. Crossover is used for combining the strings of parents whereas mutation randomly changes values in the string of the new offspring. When M new solutions (offspring) are created, a new generation has been formed. After many generations, the algorithm will eventually converge to the optimal solution of the problem. The number of generations which are necessary for this to happen is proportional to the complexity of the optimization problem.

4. Credit Insurance systems and products

4.1. What is Credit Insurance

All over the world, businesses produce and trade products. The way these businesses sell their products is either in cash or on trade credit. If the transactions of businesses depended solely on sales in cash, the turnover would be much more limited. Sales on trade credit maximize the transactions volume and the size of every company as well as the size of total economy.

All businesses worldwide sell their products on credit either solely to local buyers within the borders of their country or to buyers in other nations to promote exports of their country.

The sum of transactions made by businesses depends on the credibility between transacting members, supplier and buyer. The promise of payment is transferred to a date after the date of sale. This can be 30, 60 up to 180 days after the date of sale. In some cases, there are transactions that can be completed even up to 12 or 18 months depending on the nature of the product.

The trading behaviour and financial status of buyers is of vital importance as buyers conduct the promise of payment and they are responsible for fulfilling it after a certain period of time.

4.2. Credit insurance companies

In order to fill this gap of trading credibility, to promote commerce between transacting companies and to increase exports worldwide, in other words in order to boost the economy, financial organisations and credit insurance companies have been established. These companies are responsible for providing insurance coverage of this credibility gap. These credit insurance companies have agencies worldwide.

Today the largest insurance companies are EULER HERMES (ALLIANZ Group), ATRADIUS and COFACE. There are also numerous other smaller and more specialized insurance companies like Lloyd’s, Equinox, Mercury and others. The largest credit insurance company in China is called Sinosure.

These organisations are robust businesses and their main functions are referring to three parameters:

a) financial underwriting for each and every buyer at an international level

b) conduction of insurance policies with tailor-made special terms and

c) undertaking of legal actions against the insolvent buyer, after compensating a claim.

4.3 Financial underwriting

Insurance companies execute corporate investigation depending on financial data collected by their local agencies, specialized credit rating agencies like ICAP Group and Teiresias SA in the respective nations, banks and already existing credit insurance policies that refer to risk coverage. In corporate investigation potential negative characteristics and indicators for the credibility of buyers should be thoroughly examined.

Private individuals who buy on credit, i.e. a family which buys an air-conditioner and decides to pay in instalments, cannot be characterized as buyer. Buyers should be characterized by commercial status, registered office, organization and structure. They should be companies which operate according to the rules of the country where they are established.

This could be a general partnership family business, a limited company of medium size or a multinational company. During the corporate investigation process, the total transaction risk, the moral risk and the country risk of the buyer-company should be examined. Companies in Venezuela, for instance, cannot be covered in terms of credit insurance due to the political risk (foreign exchange prohibition risk). The buyer company receives a unique identification code in the insurance company's underwriting system. The amount of credit insurance coverage a buyer-company will receive depends on the request of the insured supplier company, the sector and the size of the buyer, the liabilities and other factors. Credit limits can vary greatly depending on all these factors mentioned.

Credit limits depend on the turnover of the buyer company, its liabilities and claims, its equity, the sector in which the company operates and the period of time the request is submitted. In some cases, external unexpected factors may influence the final credit limits approved. For instance, there is a difference between credit limits approved before and after the Covid-19 crisis, which destabilized global economy.

Credit limits may transform depending on the financial behaviour of the buyer-company, i.e. if the buyer shows punctuality in payments, respective credit limits could increase. Credit limits also depend on the financial status of the buyer-company, i.e. the image presented in balance sheets regarding the last three years. Newly established companies might receive a credit limit which will be relatively low due to their lack of payment history.

Companies whose shareholders have declared bankruptcy in the past cannot receive credit limits due to moral risk.

The most interesting part of credit insurance limits is that they are dynamic and they depend on a great number of factors that equals the number of risks within a business.

4.4. Insurance terms & regulations

The basis of credit insurance is the feeling of trust between the insurance company and the company that receives the insurance. This trust is essential prerequisite for the establishment of a long lasting relationship between the two parts in the scope of protecting the interests of the insured company.

The insured company should present all issues it faces concerning its buyers in order for these issues to be thoroughly examined and resolved through coverage of non-payment risks. The insured company may wish to share its plans regarding new exports, including the target countries, the product the company wants to promote and potential competitors it would like to hinder.

There should be thorough recording of the needs of the insured company and the desirable credit limits regarding buyer companies which could be operating in other nations like Brazil, Angola or South Africa.

Every credit insurance policy is tailor made to suit the unique demands of the insured business.

The general steps for the conduction of a credit insurance policy are the following:

1. Statement of the turnover to be insured.
2. Statement of the countries in which the buyer companies operate.
3. Definition of the credit period given to buyer companies.
4. Approximation of the price of the insurance policy - overall clearance usually takes place at the end of insurance period.
5. Statement of the maximum yearly compensation by the insurance company.
6. Definition of the coverage percentage, in other words the amount of compensation to be provided by an insurance company in case that a buyer becomes disloyal and insolvent. This percentage is usually 90%.
7. Other tailor made rules depending on the characteristics of the company to be insured.

4.5. Insurance Compensation and Claim Assertion

A case of non payment can occur due to a variety of factors. The buyer company might not have properly managed its finance, it might not have been paid by a client or it might be operating under unfavourable financial conditions, i.e. the financial instability due to Covid-19.

In this case, the insured company declares a claim for unpaid insured receivables and calls the insurance company to compensate for the non payment of the buyer.

The insurance company compensates and legally substitutes the insured supplier company. Consequently, the credit insurance company, which has its own legal department, has the right to be posed legally against the debtor and assert the amount due. The cost of all legal proceedings is included in the price of a credit insurance policy. This legal claim of the amount due, might last for years depending on the case. The resolution of this claim is no longer an issue that should be addressed by the insured supplier company, since the client has already received the compensation. The insured supplier company has the right to record the incident of non payment and the compensation received in its financial reports accounts.

Considering the case that the debtor company might be operating in Brazil, the legal assistance provided by the insurance company might prove a very helpful parameter. The credit insurance branch operating in the country of the insured supplier can grant the credit insurance subsidiary in Brazil, which has expertise on the legal system of the particular country a mandate to legally proceed with the assertion of the amount due.

Moreover, in case of total recovery of the amount due, the insured supplier company is entitled to 10% of this amount, since the insured company has previously received coverage of 90% of the amount due.

4.6. Credit Insurance Benefits

The insured company can safely sell products to every buyer company the insurance company has approved a credit limit to. The supplier does not need “to be familiar” with the buyer. This promotes the total image of an insured company in terms of marketing. The evolution of the insured company is guaranteed, as the credit insurance policy can be used as a valuable tool for corporate investigation in the scope of increasing sales.

The department of financial management within the insured company can focus on other aspects of financial development of the company, since it is guaranteed that risks are limited and manageable. The department can organize the use of budget available on a completely different basis. Financial security and avoidance of unexpected financial incidents regarding clients are remarkably reduced.

Banking institutions lend to an insured company at lower interest rates, since a financially healthy company is more likely to be punctual in terms of payments. A credit insurance policy can be submitted to a bank or factoring company for extra provision of funding to the insured company.

Accurate and early information regarding buyers facing difficulties in payments protects the insured supplier company and enables it to take steps in order to cease further sales to this particular buyer.

The total benefits of credit insurance lie in a more efficient function of the insured supplier company in terms of marketing and financial management as well as the decisions taken by the chief executive manager.

5. Modelling

During the process of evaluating the price of a certain insurance policy, the insurance broker needs to take into consideration, among others, three basic criteria that are recorded on the balance sheets of every company. The first criterion regards the claims of the insured company. As previously mentioned, claims are defined as the amount of money that other companies owe to the insured company or, alternatively, the amount of money the insured company expects to receive from other companies. The second criterion that determines the pricing of an insurance policy regards the liabilities of the insured company. Liabilities are defined as the amount of money the insured company owes to other companies or, alternatively, the cash outflow of the insured company. The third criterion that is taken into consideration during the pricing process is the total turnover of the insured client.

The first step during the process of pricing is defining the inputs and outputs of the problem. The problem has three inputs (claims, liabilities, turnover) and one output, that is, the rate to turnover, which will in turn determine the final insurance policy price.

Each of these four variables, three inputs and one output, contains a number of membership functions that represents the different categories included in each variable. For the formation of the fuzzy inference system with its inputs and outputs, an expert provided a basic framework of pricing scenarios depending on the different possible combinations of the three input variables. Table 1 simulates the decision making of an expert in the field of credit insurance during the process of pricing insurance policies. Namely, 24 possible scenarios are presented.

	T=0 to 1 mil.		**T=1 to 3 mil.**		**T=3 to 10 mil.**		**T>10 mil.**	
	L=0-50%	**L=50-100%**	**L=0-50%**	**L=50-100%**	**L=0-50%**	**L=50-100%**	**L=0-50%**	**L=50-100%**
C=5-25%	0.70%	0.90%	0.50%	0.70%	0.35%	0.45%	0.45%	0.33%
C=25-50%	0.80%	1.00%	0.60%	0.80%	0.45%	0.55%	0.55%	0.38%
C=50-70%	0.90%	1.10%	0.70%	0.90%	0.55%	0.65%	0.65%	0.43%

Table 1. Pricing by an expert

5.1. Fuzzy Inference System with 5 output categories

Given unique triplets (combinations) of input values (claims, liabilities, turnover), the credit insurance expert defines the factor (rate) based on which the final price of a certain insurance policy is calculated. According to Table 1 the minimum value of this factor is 0 whereas the maximum value the factor can take is 1.1. Based on these values, the five output categories are the following:

Membership function label	**Range**
LOW-	[0.28, 0.444]
LOW+	[0.444, 0.608]
MEDIUM	[0.608, 0.772]
HIGH-	[0.772, 0.936]
HIGH+	[0.936, 1.1]

Table 2. Fuzzy System with 5 output categories

The problem of insurance policy pricing is therefore modelled as follows:

	T=0 to 1 mil.		**T=1 to 3 mil.**		**T=3 to 10 mil.**		**T>10 mil.**	
	L=0-50%	**L=50-100%**	**L=0-50%**	**L=50-100%**	**L=0-50%**	**L=50-100%**	**L=0-50%**	**L=50-100%**
C=5-25%	MED	HIGH-	LOW+	MED	LOW-	LOW+	LOW-	LOW-
C=25-50%	HIGH-	HIGH+	LOW+	HIGH-	LOW+	LOW+	LOW-	LOW-
C=50-70%	HIGH-	HIGH+	MED	HIGH-	LOW+	MED	LOW-	LOW-

Table 3. Model 1

Based on the model presented in Table 3, a fuzzy inference system can now be created within the MATLAB environment. The fuzzy inference system will have 3 input variables (claims, liabilities and turnover) and one output variable (pricing factor). The main fuzzy workspace depicts all input and output variables and provides information about the type of the particular fuzzy system (Mamdani) and the defuzzification method used (centroid).

5.1.1. Fuzzy Inference System with 5 output categories using triangular and trapezoid membership functions

The first attempt towards creating a system with five output categories will be executed with the use of triangular and trapezoid membership functions within the input and output variables.

As far as the first input of the fuzzy inference system is concerned, namely the variable that refers to claims, there are three categories and therefore three membership functions that are equally distributed within the interval [0, 72.5].

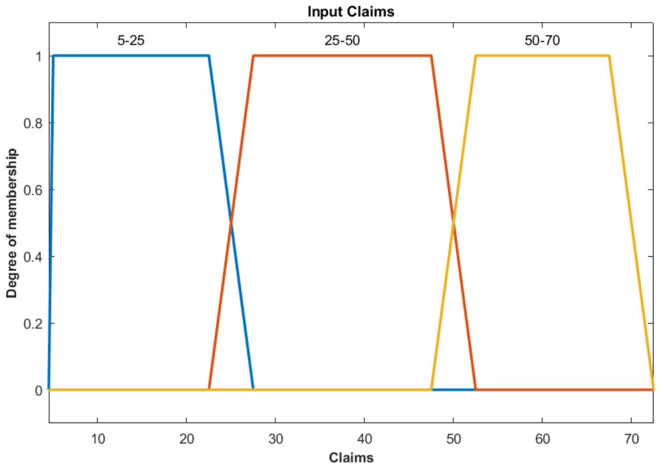


Figure 12. The input variable “Claims”

The second input variable, which refers to the liabilities of the insured company, is divided into two categories and therefore there are two respective membership functions equally distributed within the interval [0, 100].

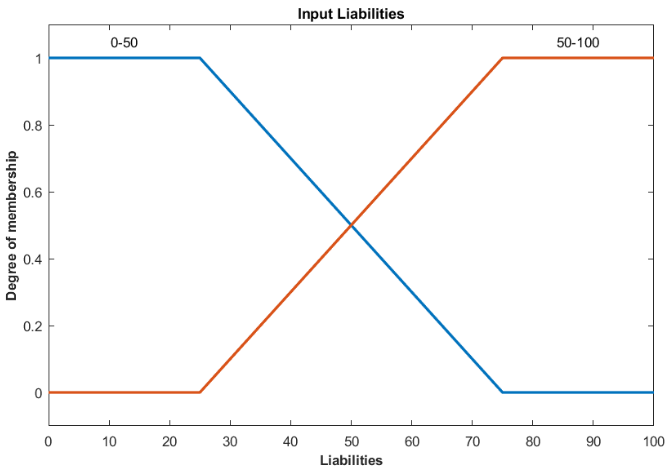


Figure 13. The input variable “Liabilities”

Regarding the third input variable of the fuzzy inference system, which refers to the turnover of the insured company, there are four categories which are not of equal size. This quality is depicted with the respective membership functions, which are distributed in a way that represents these unequal ranges of turnover.

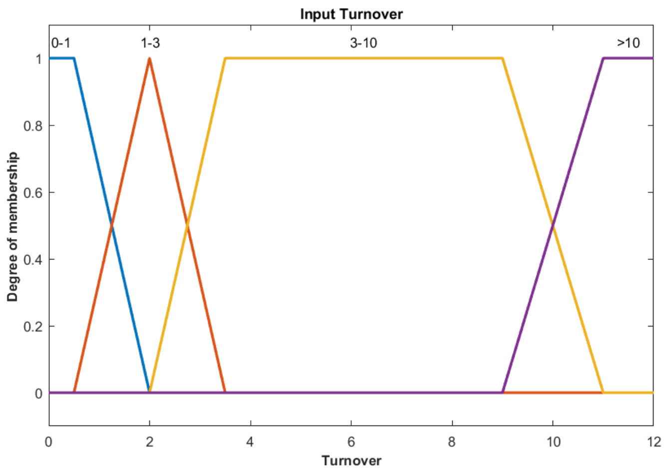


Figure 14. The input variable "Turnover"

The pricing factor, which is the output variable produced by the fuzzy inference system, is also divided in categories that are represented by membership functions that are equally distributed within the range [0, 1.1].

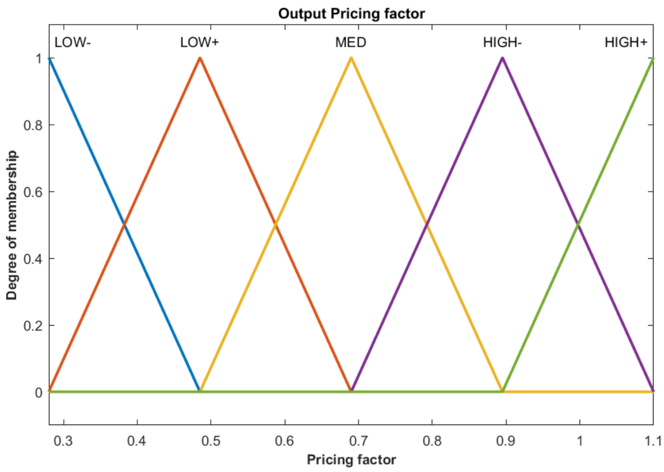


Figure 15. The output variable "Price factor"

For the correlation of the three input variables and the estimation of the pricing factor given a certain triplet of claims, liabilities and turnover, a set of logical rules has to be established. These rules are created based on the initial estimation of pricing factors conducted by the insurance broker expert (Table 3).

Based on this set of rules, the fuzzy inference system presents the correlation between different pairs of input variables and the output estimation produced. This correlation is depicted through surfaces distributed in the three dimensional space formed by the two input variables and the output variable.

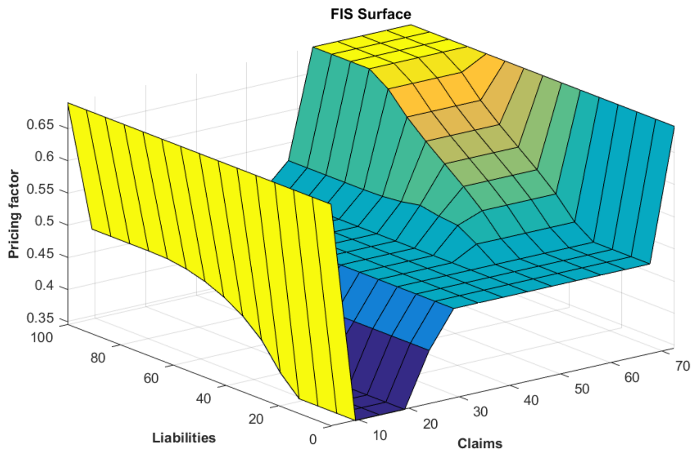


Figure 16. Correlation of "Price factor" with "Liabilities" and "Claims"

Apart from the visualization of surfaces, the fuzzy inference system also enables the visualization of rules. Using different combinations of input variable values, the user can observe the accuracy of the system regarding the approximation of the output price factor. The evaluation of results by an insurance expert is of vital importance, since the expert can decide whether the price factor produced has an appropriate and acceptable value that could be used in real-life problems or not. Different combinations of inputs and the results they produce will be presented in Chapter 6.

5.1.2. Fuzzy Inference System with 5 output categories using Gaussian membership functions

The next step towards optimizing the approximations produced by the Fuzzy Inference System with 5 output categories, is the use of Gaussian membership functions. The transformation of the triangular membership functions to simple Gaussian (gaussmf) and the trapezoidal membership functions to Two-point Gaussian (gauss2mf) enables a smoother distribution between the different categories of input and output variables. The scope of this transformation is to enhance the approximations produced by the system and increase its grade of fairness.

More specifically, the membership functions concerning the input variable of claims, liabilities and turnover are now formed as follows:

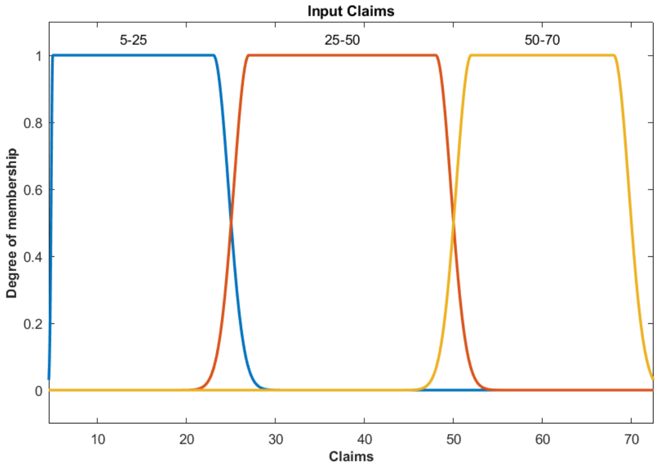


Figure 17. The input variable "Claims"

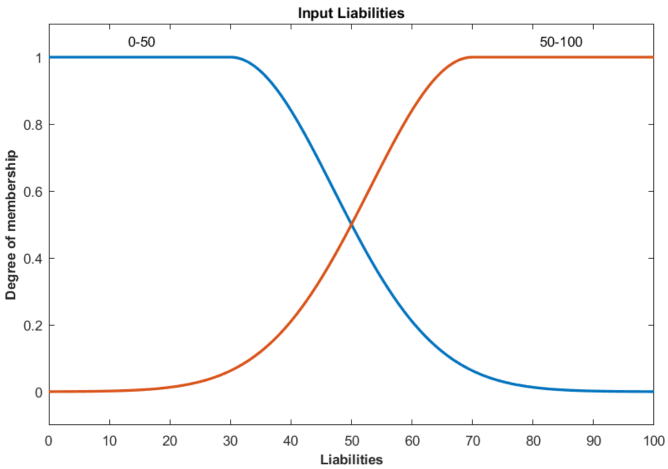


Figure 18. The input variable "Liabilities"

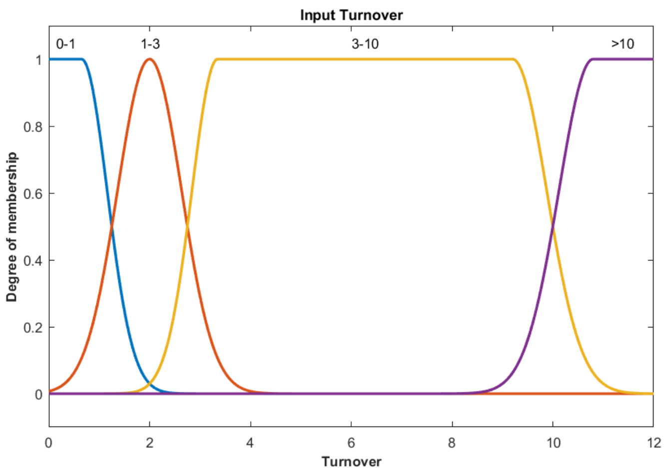


Figure 19. The input variable "Turnover"

The membership functions included in the output price factor take the following form:

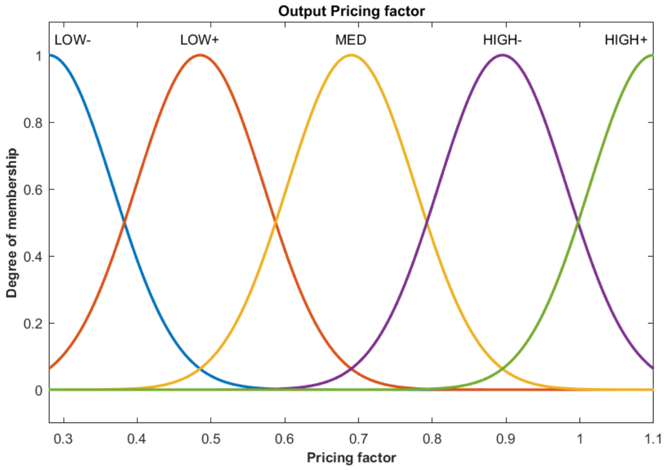


Figure 20. The output variable "Price factor"

As expected, the surface that correlates the dimensions of liabilities and claims with the output dimension of the price factor is smoother than previously due to the use of Gaussian membership functions. The following surface reveals monotony between the two input variables and the output variable. This monotony is in some cases used as an indicator of the grade of consistency between the rules that concern the three variable and is usually desirable in fuzzy inference systems.

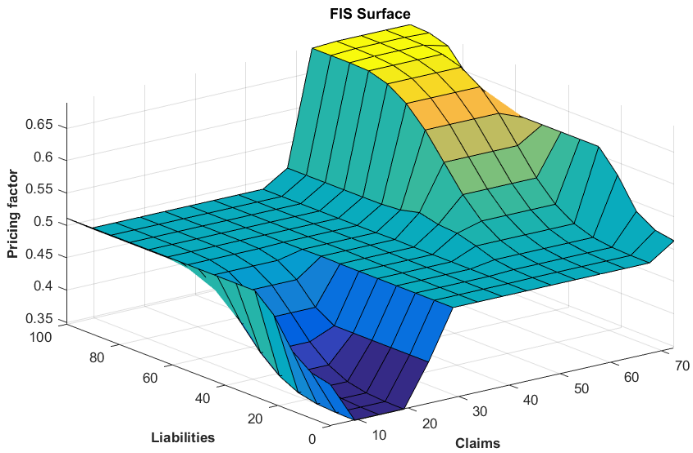


Figure 21. Correlation of "Price factor" with "Liabilities" and "Claims"

5.2. Fuzzy Inference System with 8 output categories

In the scope of increasing the accuracy of the pricing factor approximations produced, the output is divided into a larger number of categories. As described above, according to Table 1 the minimum value of this factor is 0 whereas the maximum value of the factor is 1.1. Based on these values, the eight output categories are the following:

Membership function label	Range
LOW-	[0.28, 0.3825]
LOW	[0.3825, 0.485]
LOW+	[0.485, 0.5875]
MED-	[0.5875, 0.69]
MED+	[0.69, 0.7925]
HIGH-	[0.7925, 0.895]
HIGH	[0.895, 0.9975]
HIGH+	[0.9975, 1.1]

Table 4. Fuzzy System with 8 output categories

The problem of insurance policy pricing is therefore modelled as follows:

	T=0 to 1 mil.		T=1 to 3 mil.		T=3 to 10 mil.		T>10 mil.	
	L=0-50%	L=50-100%	L=0-50%	L=50-100%	L=0-50%	L=50-100%	L=0-50%	L=50-100%
C=5-25%	MED-	HIGH	LOW+	MED+	LOW-	LOW	LOW-	LOW-
C=25-50%	HIGH-	HIGH+	MED-	HIGH-	LOW	LOW+	LOW-	LOW-
C=50-70%	HIGH	HIGH+	MED+	HIGH	LOW+	MED-	LOW-	LOW

Table 5. Model 2

5.2.1. Fuzzy Inference System with 8 output categories using triangular and trapezoid membership functions

In this case, the fuzzy inference system is constructed with input membership functions that are identical to the model created in paragraph 5.1.1. For the sake of completeness the input membership functions are once again presented below.

As far as the first input of the fuzzy inference system is concerned, namely the variable that refers to claims, there are three categories and therefore three membership functions that are equally distributed within the interval [0,72.5] (Fig. 22). The second input variable, which refers to the liabilities of the insured company, is divided into two categories and therefore there are two respective membership functions equally distributed within the interval [0,100] (Fig. 23).

Regarding the third input variable of the fuzzy inference system, which refers to the turnover of the insured company, there are four categories which are not of equal size. This quality is depicted with the respective membership functions, which are distributed in a way that represents these unequal ranges of turnover (Fig. 24).

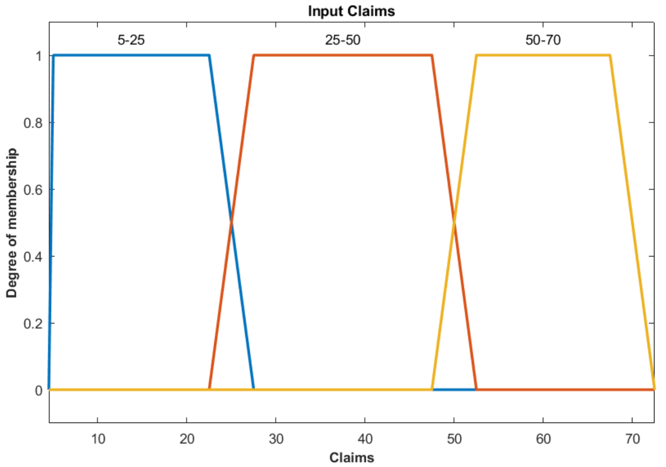


Figure 22. The input variable "Claims"

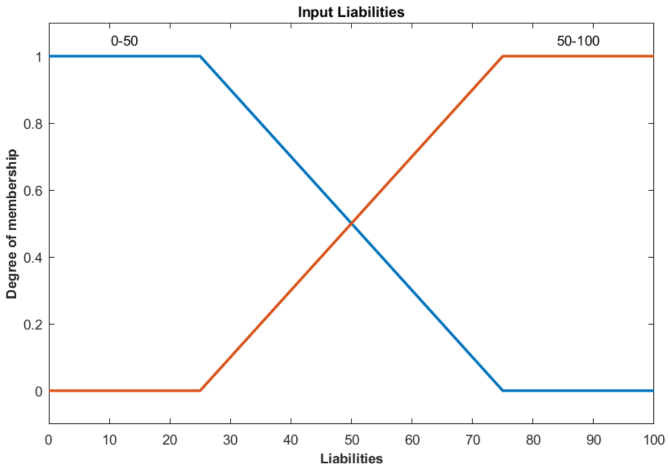


Figure 23. The input variable "Liabilities"

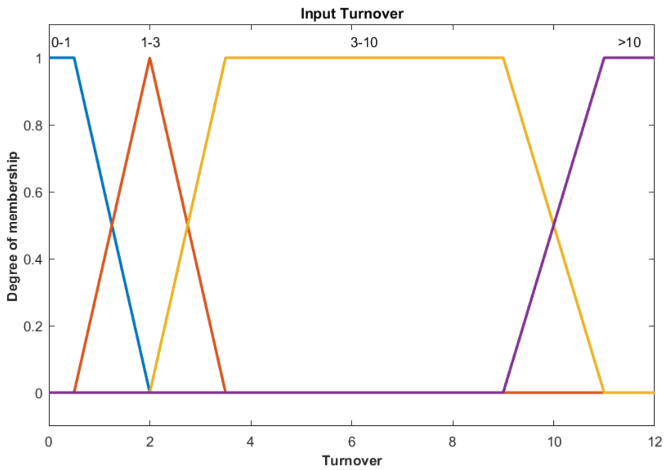


Figure 24. The input variable "Turnover"

The difference lies in the way in which the output variable is formed. In this case, the interval [0.28, 1.1] will be divided into eight equally distributed categories (Table 4).

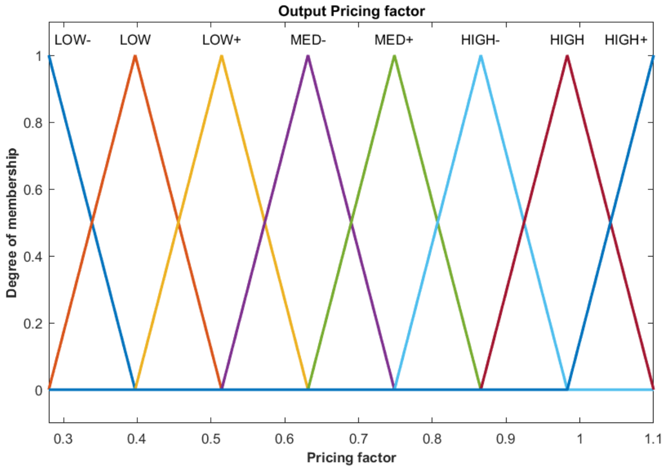


Figure 25. The output variable "Price factor"

The correlation between the input variables of claims and liabilities with the output variable is shown below.

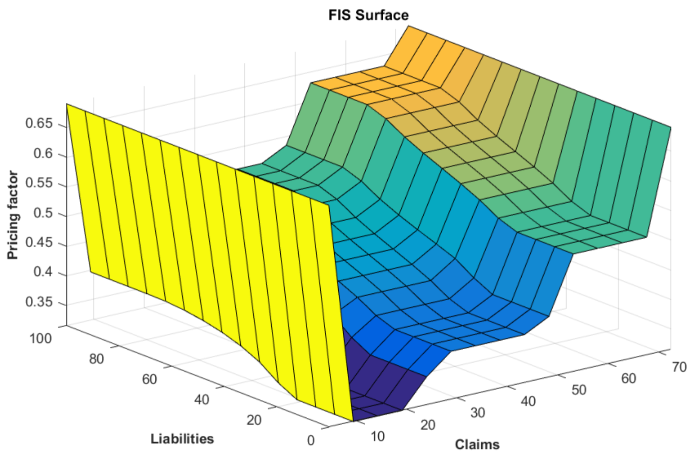


Figure 26. Correlation of "Price factor" with "Liabilities" and "Claims"

5.2.2. Fuzzy Inference System with 8 output categories using Gaussian membership functions

Respectively, the next step towards optimizing the approximations produced by the Fuzzy Inference System with 8 output categories is the use of Gaussian membership functions.

More specifically, the membership functions concerning the input variables of claims, liabilities and turnover are now formed as follows:

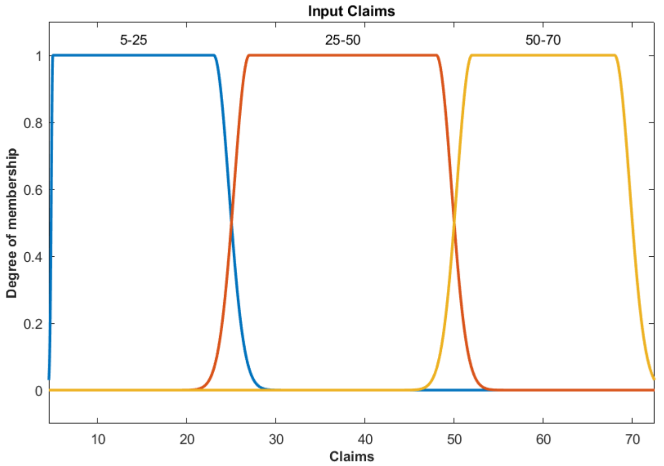


Figure 27. The input variable "Claims"

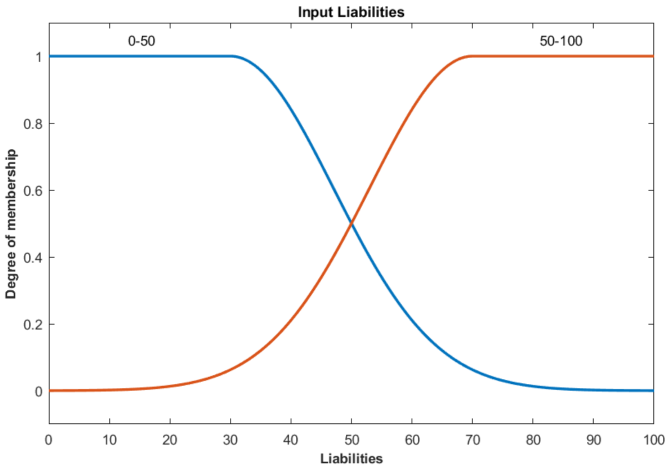


Figure 28. The input variable "Liabilities"

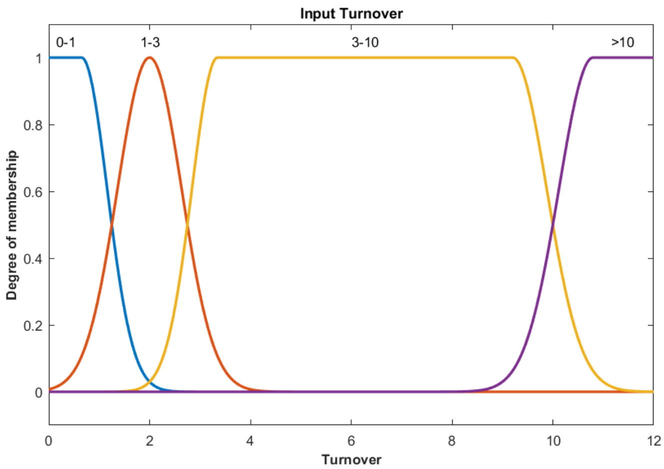


Figure 29. The input variable "Turnover"

The membership functions included in the output price factor take the following form:

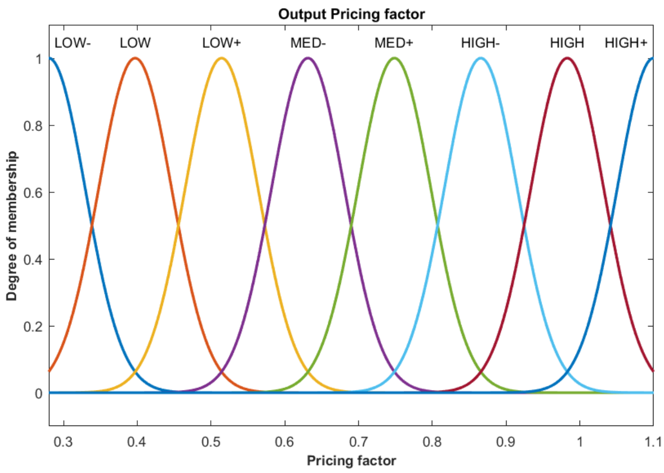


Figure 30. The output variable “Price factor”

The correlation between the input variables of claims and liabilities with the output variable is shown below.

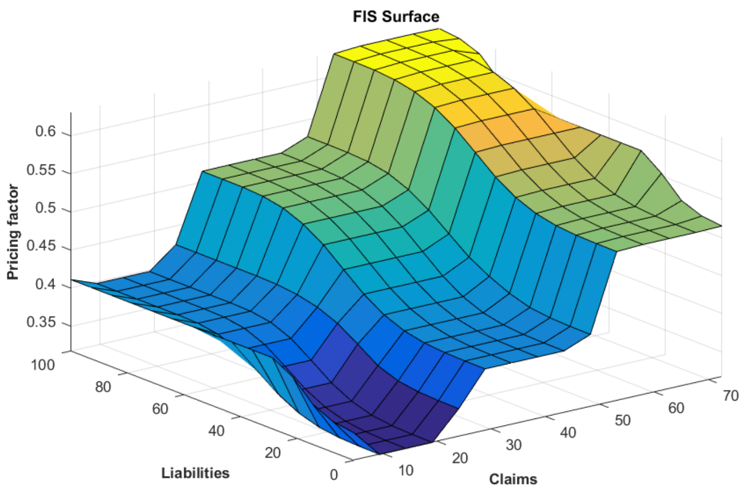


Figure 31. Correlation of “Price factor” with “Liabilities” and “Claims”

5.3. Adaptive Neurofuzzy Inference System

As previously analyzed, an Adaptive Neurofuzzy Inference System addresses the same problem that Fuzzy Inference Systems targets to solve. The most important difference between the two systems lies on their basic principles.

A Fuzzy Inference System is based on a number of rules that simulate the human cognitive process of decision making. This system simulates the way in which a decision should theoretically or ideally be reached and this is the reason why Fuzzy

Inference Systems are expected to produce results that do not precisely approximate actual values of pricing factors that are used in real life. When constructed properly, a Fuzzy Inference System indicates how pricing factors should ideally be attributed based on the theoretical pricing criteria of an insurance expert. On the contrary, an Adaptive Neurofuzzy Inference System receives triplets of actual input values -claims, liabilities and turnover of the insured company- as well as the output attributed by the credit insurance company, from a large dataset of anonymous registrations.

After processing a large number of these quadruplets coming from the anonymous dataset provided, the Adaptive Neurofuzzy Inference System divides the dataset, namely the group of clients with their unique characteristics, into an optimal number of clusters. From a more technical perspective, the Adaptive Neurofuzzy Inference System, given the anonymous registrations of the dataset, creates its own complex rules and membership functions whose form cannot be initially perceived by the human brain.

Before creating and training the Adaptive Neurofuzzy Inference System, a file including all the training data has to be created with the use of a simple code in MATLAB Environment. In this case, four vectors of the same size are going to be created. These vectors correspond to the three input variables and the output variable of the system. The size depends on the number of quadruplet registrations selected as training data. After the creation of four equally sized column vectors, a table containing all four column vectors is produced and registered into a .dat file. The anonymous dataset provided by the insurance company contains 150 quadruplets of three inputs -claims, liabilities and turnover- and one output -pricing factor attributed to each insured company- corresponding to the 150 companies included in the sample. 140 registrations are used for the creation of a training set and the remaining 10 will be used for testing the effectiveness of the system. The “training.dat” file created therefore contains 4x140 elements.

For the construction of an Adaptive Neurofuzzy Inference system, a Sugeno type Fuzzy Inference system has to be created. Then, a number of input and output variables have to be added to the system depending on the number of inputs and outputs included in the dataset provided. Contrary to the procedure followed during the creation of a

Mamdani type Fuzzy Inference System, the membership functions added into the Sugeno type Fuzzy Inference System will not be edited.

After the addition of input and output variables, the choice of editing the Adaptive Neurofuzzy Inference System is selected. In the respective window, the choice of loading training data from the "training.dat" file is selected. The Fuzzy Inference System is generated through the method of subtractive clustering, which creates an optimal number of clusters, based on the training data provided. For the process of training the Fuzzy Inference System, hybrid optimization method is selected with zero error tolerance and ten training epochs. The plot formed by the system depicts the training data loaded. The clusters created by the system for the clusters of the input variables are the following.

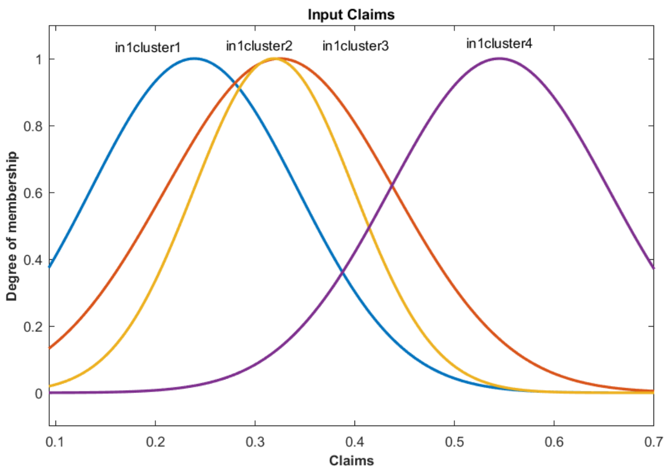


Figure 32. The input variable "Claims"

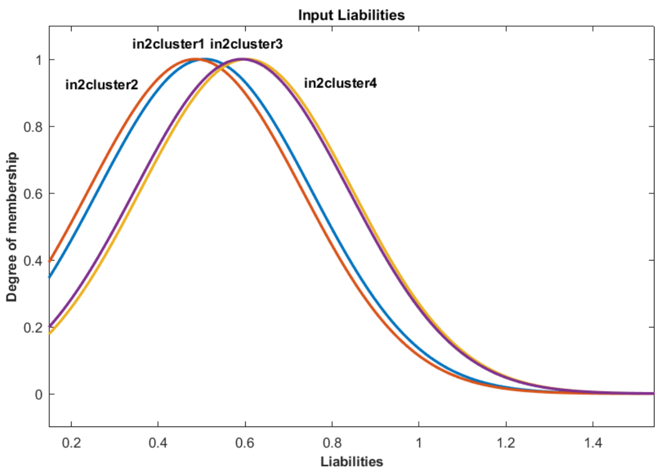


Figure 33. The input variable "Liabilities"

Regarding the input of turnover, the membership functions created by the system are the following.

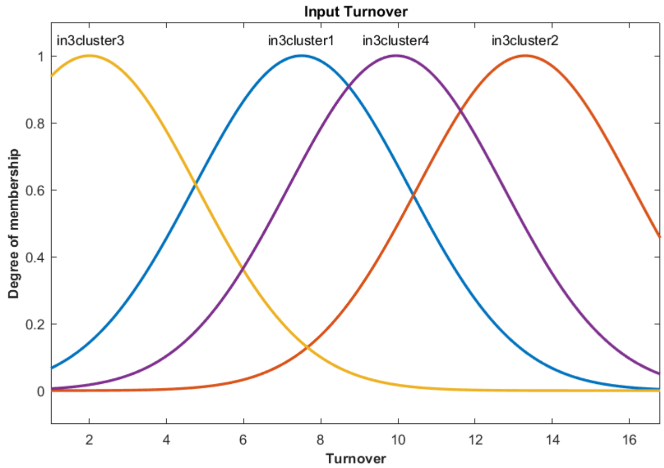


Figure 34. The input variable "Turnover"

The surface created between the input variables and the output variable pricing factor as constructed by the Adaptive Neurofuzzy Inference System is the following.

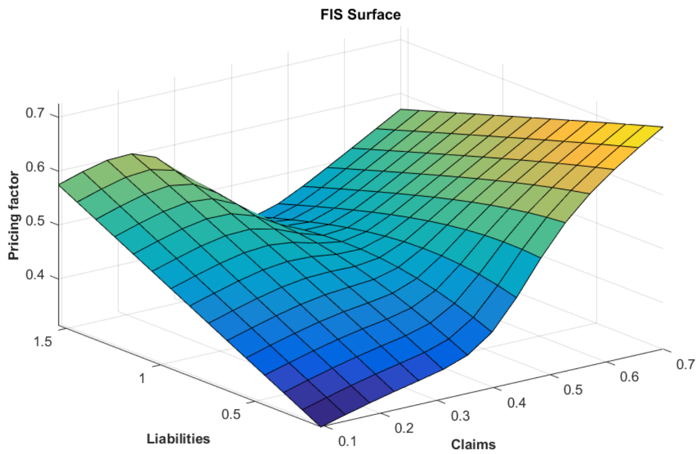


Figure 35. Correlation of "Price factor" with "Liabilities" and "Claims"

6. Optimization

After the construction of a fuzzy inference system based on the information provided by the insurance expert and the creation of a neurofuzzy inference system based on the available credit insurance policies, the question of how the former could be improved so that it approximated the latter in the closest way is now raised. As previously discussed, the fuzzy inference system reflects the way in which an insurance expert would address the problem of pricing credit insurance policies under theoretically normal circumstances. In other words, given the details of a prospective client the insurance expert attempts to predict the exact pricing factor that will define the cost of an insurance coverage. In real time conditions, each company has its own different characteristics and this is why the numbers produced by the neurofuzzy inference system are not identical to the numbers produced by the fuzzy inference system. The problem is how the theoretically designed fuzzy inference system could be optimized in order to approximate more closely the real time conditions faced by prospective client companies.

6.1. Basic concepts of optimization and description of the problem

The attempt of optimizing the already existing fuzzy inference system took place in Matlab environment. A description of the optimization problem is provided below.

The optimal values of design variables that would produce enhanced results when set as parameters into the objective function were calculated by the genetic algorithm of Matlab. This algorithm contains a set of parameters which can be defined by the user. For example, the number of generations, the population size, the crossover and mutation probabilities and other parameters, can be selected by the designer of the system. By default, the population contains 50 individuals in case that the number of design variables is lower or equal to 5 and 200 members if this number is greater. The creation function for the initial population was chosen to be the linear feasible population function. The number of generations is 100 multiplied by the number of variables. The tolerance of the algorithm is also an important factor that can be selected within a genetic algorithm. This factor denotes the permitted relative change of the objective function. In case that the relative change of the fitness function is lower that the Function Tolerance parameter, the algorithm stops. The tolerance of the algorithm is set to 10^{-6} by default and the penalty factor to 100. For the modification of the population through generations, tournament selection method is used by default. Regarding the main operators of the genetic algorithm, Gaussian mutation and scattered crossover with possibility of 0.8 were used. Another important parameter included in the parameters of this genetic algorithm is called MaxStallGenerations. It refers to the maximum number of generations after which the algorithm will stop if the objective function is not improved within the desired tolerance.

The objective function is defined in this thesis as the Mean Square Error of two vectors. The first vector represents the actual insurance policy prices found in the anonymous dataset provided. The second vector refers to the output values produced by the optimized fuzzy inference system, given certain input values. The design variables are the possible positions of points a-α, a, a+β and a-α, a, b, b+β that define the membership functions of both inputs and output, for triangular and trapezoidal functions as shown in the figures below.

The optimization problem in this thesis is to minimize the Mean Square Error between the previously mentioned vectors, thus to maximize the approximation of real pricing factors achieved by the optimized fuzzy inference system.

Minimize:

$$Mean\ Square\ Error = \frac{1}{\mathrm{n}} \sum_{i=1}^{n} \left(\mathrm{Y_i} - \widehat{\mathrm{Y}}_\mathrm{i}\right)^2$$

where $\mathrm{Y_i}$ are the actual values, $\widehat{\mathrm{Y}}_\mathrm{i}$ are the optimized fuzzy inference system outputs, and n is the number of registrations within the dataset.

Subject to:

$$x(i) < x(i+1)$$

The membership functions are discretized in i, i+1 points as shown in the following figures.

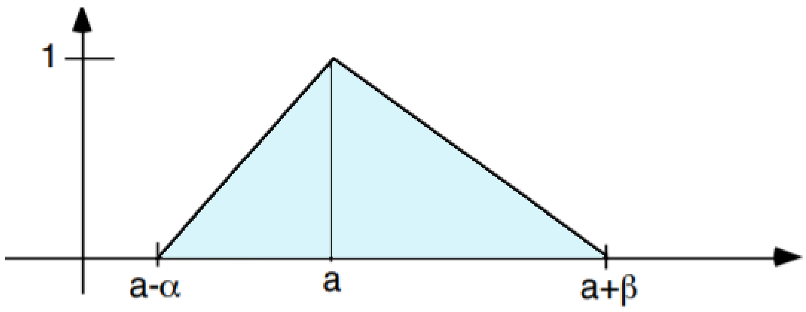


Figure 36. Trimf

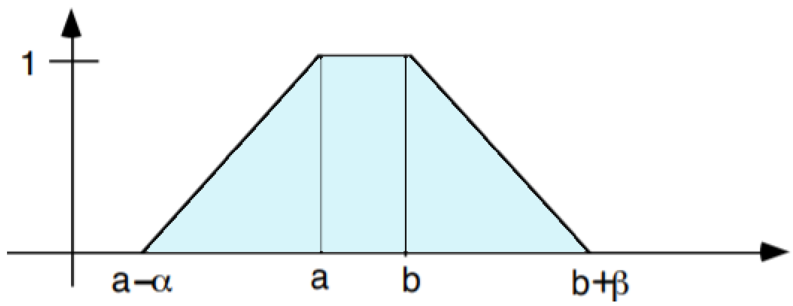


Figure 37. Trapmf

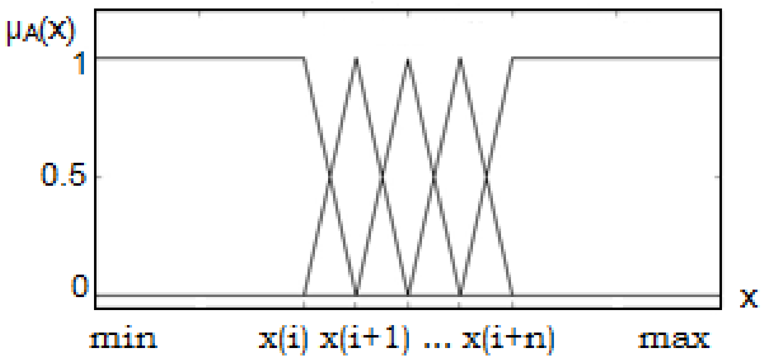


Figure 38. Discretization of membership functions for optimization (Tairidis et al. 2016)

Five different cases, that is, the initial system and four optimization cases are considered as described in 6.2.

The results for several cases of membership functions modification are going to be presented for different values of these parameters. By default, the genetic algorithm of Matlab produces solutions starting from random initial solutions within the population.

A pseudo code of the implementation of the genetic algorithm is provided below.

Generation=0

Initialize

Evaluate

Keepbest

do generation=1, MaxGens

Select

Crossover

Mutate

Report

Evaluate

Elitist

enddo

There are two stopping criteria which are responsible for terminating the implementation of a genetic algorithm. The first one refers to the case in which the algorithm has reached the initially defined maximum number of generations. The second criterion refers to the case in which the value of the objective function between two generations is no longer improved across generations (Stall limit) (Tairidis et al. 2016).

6.2. Implementation of genetic algorithm in Matlab environment

6.2.1. Initial and optimized fuzzy systems

Case 1: In this case all characteristics and parameters of the initial fuzzy inference system are set. A Matlab file creates a fuzzy inference system that is identical to the one created within the fuzzy graphic environment, which was described in the previous chapters.

Case 2: The differentiation of this version to the case 1 is that the fuzzy system that is created includes design variables for the optimization of the membership functions of the first input. Instead of mapping the exact points at which membership functions are formed, 6 design variables need to be defined by the genetic algorithm in order for the fuzzy inference system to be formed.

Case 3: In this case, the fuzzy system that is created includes design parameters as far as the membership functions of the second input are concerned. Instead of mapping the exact points at which membership functions are formed, 2 values need to be defined by the genetic algorithm in order for the fuzzy inference system, that is, the membership functions, to be formed.

Case 4: The differentiation of this case to the initial system is that the fuzzy inference system that is created here includes design parameters as far as the membership functions of the third input are concerned. Namely, instead of mapping the exact points at which membership functions are formed, 5 design parameters need to be defined by the genetic algorithm in order for the fuzzy inference system to be formed.

Case 5: In this latter case, the fuzzy system that is created includes design parameters as far as the membership functions of the output are concerned. Instead of mapping the exact points at which membership functions are formed, this file contains 6 design parameters that need to be defined by the genetic algorithm in order for the fuzzy inference system to be formed.

6.2.2. Objective function

The proper selection of the objective function is the most substantial part of the optimization. In our problem, the fuzzy system with its design parameters is considered and an optimization criterion is set. This optimization criterion is also called objective function. The objective function needs to be set with great consideration since the genetic algorithm produces solutions in the scope of finding the parameters that create the best value of this objective function. The optimization algorithm should produce solutions that either maximize or minimize the value of the objective function. In this particular thesis, the target was to minimize the difference between the actual pricing factor results produced by the neurofuzzy inference system and the outputs produced by the various fuzzy inference systems created in each iteration based on the parameters provided by the genetic algorithm.

6.2.3. Constraints of the design variables

A set of constraints, in the way in which the fuzzy inference system is optimized, should be set. These constraints have to be taken into consideration by the genetic algorithm in order to ensure that the solutions respect the physical (geometrical) and technical rules of the fuzzification process, that is, the proper creation of the membership functions. For instance, regarding case 2 in 6.1.2, the optimized set of parameters should follow an ascending order. In case that the genetic algorithm provides a set of solutions that do not obey to these constraints, the membership functions cannot be graphically presented in the new optimized fuzzy inference system, due to geometric inconsistency.

6.2.4. Genetic Algorithm in Matlab

The genetic algorithm of the Optimization toolbox in Matlab is used in the present thesis. The scope of the whole procedure is to take into consideration the constraints which were described above and create a vector that contains the optimized design parameters which, in turn, optimize the fuzzy inference system. This means that the scope of the particular genetic algorithm is to change the values included in the design variable vector and compare the results of the fuzzy inference system that is created every time to the already provided results of the training dataset. The scope of this process is to optimize the way in which the fuzzy inference system approximates pricing factors under real time circumstances.

6.3. Hindering Factors

During the search for potential solutions that would enhance the total value of the objective function, various problems occurred. During the pursuit of optimized factors the definition of the most suitable objective function was of vital importance, since this would be the criterion according to which the genetic algorithm would function. In the beginning, the difference between actual values provided by the dataset and the output of the optimized fuzzy system was the objective function that would evaluate the effectiveness of the optimization algorithm. This criterion did not prove to be the most appropriate one. As discovered within the documentation of Matlab, one of the most suitable criteria to be used regarding optimization with a genetic algorithm is the Mean Square Error of this difference. The function of Mean Square Error has been therefore used for calculating the difference of the formerly mentioned vectors.

Another obstacle faced during the optimization process was the fact that the genetic algorithm was initially not respecting the constraints set for the optimization problem. Various ways of posing constraints in an optimization problem using the genetic algorithm are presented in the documentation of Matlab. Under the initial conditions, the constraints were set in form of a matrix and the genetic algorithm did not take into consideration the constraints as expected. The alternative way of setting constraints in the form of two separate vectors as presented in 6.1.3. proved to be the most suitable one, as the genetic algorithm eventually respected the rules of the optimization problem.

Another vitally important factor that hindered the process of finding an enhanced solution was the fact that the initial fuzzy system that was designed by the insurance broker expert functioned in intervals that were relatively limited compared to the actual characteristics of the companies. This difference fell unnoticed for a long period of time and it was impeding the whole optimization process. The warnings produced by Matlab stated that values were constantly exceeding the intervals of the initially designed fuzzy inference system. Ostensibly, no mistakes were observed in the parameters of the initial fuzzy inference system and the difficulty of recognizing the fact that the human brain of an expert failed to model all possible scenarios rendered the total process exceptionally challenging. This flaw was eventually fixed and the whole fuzzy inference system had to be redesigned with broader sets of values in order to be optimized.

The last barrier that occurred during the process of optimizing design parameters of the initial fuzzy inference system was setting the lower and upper bounds of the design variable vector. The elements contained in this vector were expressed as percentages so the lower and upper bound of every respective element within the vector should be 0.001 and 0.999 under normal circumstances. Nevertheless, the algorithm presented a remarkable difficulty in converging to the final values of the design variables when the dimensions of the vector exceeded the number of two. The alternative solution of restricting the lower and upper bound of design variables was eventually implemented. In this case, the intervals within which the potential solutions could be found became narrower. This improved the speed and convergence of the genetic algorithm and the results produced were satisfactory. Great attention should be drawn in the final values of the vector produced by the genetic algorithm. In case that final values produced by the optimization algorithm are identical or close to the lower or upper bound initially set, the algorithm has to be implemented again with a broader interval given to this specific element of the design variables. There is a high possibility of further enhancing the final value of the objective function with this alteration. All final elements within the vector produced by the genetic algorithm of Matlab should preferably be found away from the respective lower and upper bounds set.

6.4. Optimization results

After creating all necessary files and fixing the parameters, different cases of the initial fuzzy system are optimized. There are various combinations of membership function parameters that had to be defined and their outcomes were compared. Membership functions of every separate input of the original fuzzy inference system were initially optimized. The output variable of the system was then optimized with given input parameters. The optimized values that occurred for every respective design variable are shown in Table 6. The results presented in the following table have been produced with genetic algorithm parameters set by default as described in the beginning of this Chapter.

Input 1	0.1213	0.1512	0.4644	0.5100	0.6711	0.9540
Input 2	0.1902	0.4388				
Input 3	0.0366	0.0811	0.2494	0.6101	0.8100	
Output	0.2442	0.3259	0.4800	0.6500	0.6600	0.8179

Table 6. Optimized Design Values

Two methods were used in order to find the design variables that optimize the value of the objective function. The first one was to combine values illustrated in Table 6 in order to investigate how the objective function value is altered. The second method refers to setting unknown design parameters simultaneously for more variables and observing their total impact on the objective function value.

Initially, different combinations of design variable vectors were considered and all outcomes were compared based on the value of the objective function. All comparisons are made considering the original objective function value of the initial fuzzy inference system before any optimization effort. The results are shown in Table 7.

Optimized variables	Evaluation Function (MSE)	Optimization Percentage
Without Optimization	0.0037	0%
Input 1	0.0033	10.8 %
Input 2	0.0035	5.40%
Input 3	0.0034	8.10%
Inputs 1,2	0.0031	16.2%
Output	0.0031	16.2%
Inputs 1,3	0.0034	8.10%
Inputs 2,3	0.0031	16.2%
Inputs 1,2,3	0.0032	13.5%
Inputs 1,2-Output	0.0030	18.9%
Inputs 1,3-Output	0.0039	-5.40%
Inputs 2,3-Ouput	0.0033	10.8%
Inputs 1,2,3-Output	0.0037	0%

Table 7. Combinations of separately optimized design parameters

In the majority of combinations presented, there is an improvement in the way in which fuzzy inference system approximations function. The combinations of inputs 1,3-output and inputs 1,2,3-output present deviant results, which can be attributed to the way in which these variables are related to each other.

A second method, which refers to the actual optimization of different design variables combinations, was also tried. The results that occur by using different numbers of variable combinations, initial population sizes and generations are illustrated in the following tables.

After observing the results produced by trials presented in Table 7, the question of whether the fitness function value could be further improved or not occurs. The best fitness function value occurs by combining the first two inputs with the output variable. Taking into consideration the formerly mentioned observation, an effort of optimizing all three design variables at once is the next step. Optimization results for different population sizes and generations are illustrated in Table 8.

Population Size	Generations	Function Evaluation (MSE)
30	50	0.0030
70	50	0.0024
100	80	0.0025

Table 8. Optimization of inputs 1-2, output

From the numbers presented above, it can be inferred that increasing the population size can enhance the results produced by the genetic algorithm. Nevertheless, a larger population size and number of generations does not always guarantee an enhanced fitness function value.

There have been several efforts of optimizing different combinations of design variables with 70 initial solutions within the population and 50 generations. A representative sample of these efforts is presented in Table 9.

Optimized variables	Evaluation Function (MSE)	Optimization Percentage
Inputs 1,2	0.0032	13.5%
Inputs 1,3	0.0032	13.5%
Inputs 2,3	0.0031	16.2%
Inputs 1,2,3	0.0029	21.6%
Inputs 1,2-Output	0.0024	35.1%
Inputs 1,3-Output	0.0029	21.6%
Inputs 2,3-Output	0.0027	27%
Inputs 1,2,3-Output	0.0026	29.7%

Table 9. Optimization results for different initially created combinations of design variables

7. Results and discussion

7.1. Fuzzy Inference System Results

As mentioned in Chapter 5, four different fuzzy inference system versions were designed. The characteristics of every system are summarized as a reminder in the following table.

FIS name	Input Variables	Output Variable
5_simple	3 with trimf, trapmf	5 with trimf
5_gauss	3 with gaussmf ,gauss2mf	5 with gaussmf
8_simple	3 with trimf, trapmf	8 with trimf
8 gauss	3 with gaussmf,gauss2mf	8 with gaussmf

Table 10. Four Fuzzy Inference System versions

After the construction of these four different versions, the model that most effectively approximates the real pricing factor values should be selected. In Table 11 there is a comparison between the output pricing factors produced by the four respective versions of the fuzzy inference system and the real pricing factors that correspond to actual triplets of input values, namely the claims, liabilities and turnover of the insured company.

The desired behaviour of these models will be a fair attribution of output pricing factor values, especially in the case that an input parameter is found on the boundary between two categories of the respective input. In the first three columns of the table presented below, the actual input values of ten anonymous insured companies are examined. The numbers below the inputs of claims and liabilities denote the percentage of claims to turnover and the percentage of liabilities to turnover of the insured company. The third parameter, namely the turnover, is translated into millions of Euros.

The results produced by the four versions of the fuzzy inference system compared to the real output pricing factor are also presented in the following table. Finally, the mean square error compared to actual values is used as a means for the evaluation of each fuzzy model presented.

claims	liabilities	turnover	5_simple	5_gauss	8_simple	8_gauss	actual
0.3000	0.5400	1.00	0.754	0.768	0.837	0.846	0.8
0.1891	0.3361	2.38	0.522	0.515	0.533	0.531	0.55
0.5000	0.6500	4.00	0.588	0.591	0.542	0.559	0.63
0.3126	0.3614	5.95	0.485	0.487	0.428	0.419	0.45
0.1130	0.1087	6.90	0.346	0.348	0.316	0.318	0.3
0.3123	0.6325	8.55	0.485	0.487	0.482	0.494	0.53
0.3570	0.6340	10.00	0.458	0.460	0.446	0.450	0.55
0.5439	0.6326	11.95	0.349	0.348	0.393	0.397	0.45
0.2969	0.3814	14.92	0.350	0.349	0.319	0.318	0.33
0.2713	0.6244	16.00	0.349	0.348	0.319	0.318	0.37
M.S.E.			**0.0029536**	**0.0028641**	**0.0029233**	**0.0025756**	**0**

Table 11. Comparison of four fuzzy models

From the results illustrated in the table above, it can be inferred that the fuzzy inference system with Gaussian membership functions and eight output categories is the one that provides the closest approximation of actual pricing factors. The 8_gauss model is going to be selected for comparison to the adaptive neurofuzzy inference system in the following paragraph.

7.2. Adaptive Neurofuzzy Inference System Results

Based on the registrations provided by the training dataset, the Adaptive Neurofuzzy Inference System created a number of output pricing factors regarding the ten triplets of inputs -claims, liabilities and turnover- included in the testing dataset. The precision of approximations produced by the system depends highly on the quality of the dataset as well as on the quantity of instances provided by the expert, namely the insurance company.

Due to the great confidentiality that governs credit insurance companies, the provision of anonymous data has been proven a challenging procedure. In the scope of training an Adaptive Neurofuzzy Inference System properly, thousands of instances may have to be included in the training dataset in order for the system to approximate actual pricing factors accurately. In order to construct all possible rules that depict the complex relationships between the inputs and outputs, the Adaptive Neurofuzzy Inference system needs a large number of training instances.

Nevertheless, the accuracy of the Adaptive Neurofuzzy System with the use of 140 anonymous instances is of relevance to the way in which Fuzzy Systems function and is therefore examined in this thesis. In the table presented below, a comparison between the pricing factors produced by the most accurate Fuzzy Rule-based Inference System of the previous paragraph, namely the 8_gauss model, and the Adaptive Neurofuzzy Inference System is provided.

claims	liabilities	turnover	8_gauss	ANFIS	actual
0.3000	0.5400	1.00	0.846	0.677	0.80
0.1891	0.3361	2.38	0.531	0.566	0.55
0.5000	0.6500	4.00	0.559	0.596	0.63
0.3126	0.3614	5.95	0.419	0.520	0.45
0.1130	0.1087	6.90	0.318	0.294	0.30
0.3123	0.6325	8.55	0.494	0.462	0.53
0.3570	0.6340	10.00	0.450	0.430	0.55
0.5439	0.6326	11.95	0.397	0.445	0.45
0.2969	0.3814	14.92	0.318	0.337	0.33
0.2713	0.6244	16.00	0.318	0.366	0.37
		M.S.E.	**0.0025756**	**0.0040575**	**0**

Table 12. Comparison of ANFIS to a Fuzzy Model

It can be observed that the Mean Square Error calculated for the Adaptive Neurofuzzy Inference System is much higher than the selected 8_gauss Fuzzy model. This can be attributed to the small training dataset available. Given a sufficiently large training dataset, the Adaptive Neurofuzzy Inference System is expected to produce actual pricing factor approximations that are more accurate compared to the ones produced by the Fuzzy Inference System, since the latter is based on ideal theoretical rules.

7.3. Optimization Results

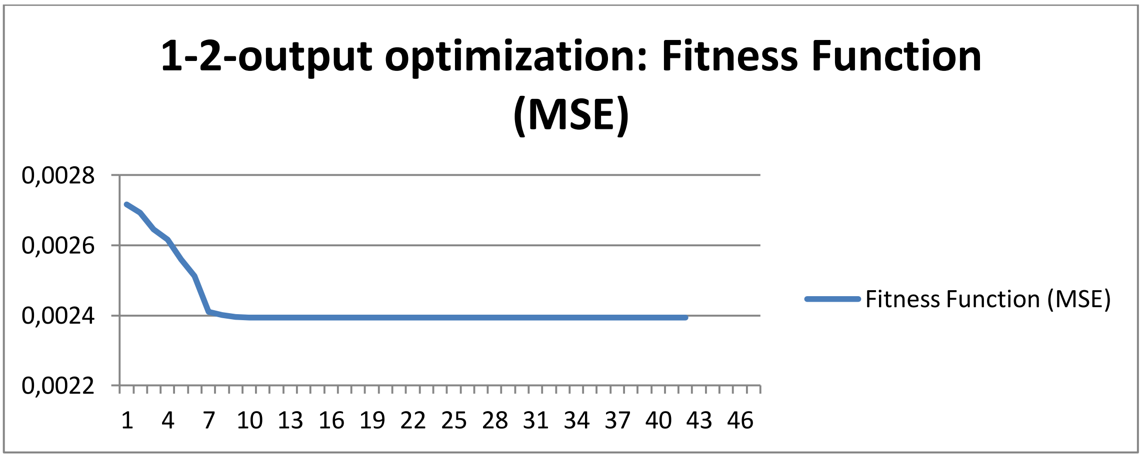
In the last part of this thesis, the 8_simple fuzzy inference system version was selected for optimization. The optimization criterion required that the Mean Square Error between output values produced by the initially selected fuzzy inference system and the 150 actual values found in real credit insurance policies should be as close as possible, given the same input values. Remarkable difference between cases of separately and collectively optimizing design parameters has been noticed. This difference is presented in the following table.

Optimized variables	Collectively combined (MSE)	Separately combined(MSE)	Improvement Percentage
Inputs 1,2	0.0031	0.0031	-3.2%
Inputs 1,3	0.0032	0.0034	5.8%
Inputs 2,3	0.0031	0.0031	0 %
Inputs 1,2,3	0.0029	0.0032	9.4%
Inputs 1,2-Output	0.0024	0.0030	20%
Inputs 1,3-Output	0.0029	0.0039	25.6%
Inputs 2,3-Output	0.0027	0.0033	15.2%
Inputs 1,2,3-Output	0.0026	0.0037	29.7%

Table 13. Comparison of collectively and separately optimized combinations of design variables

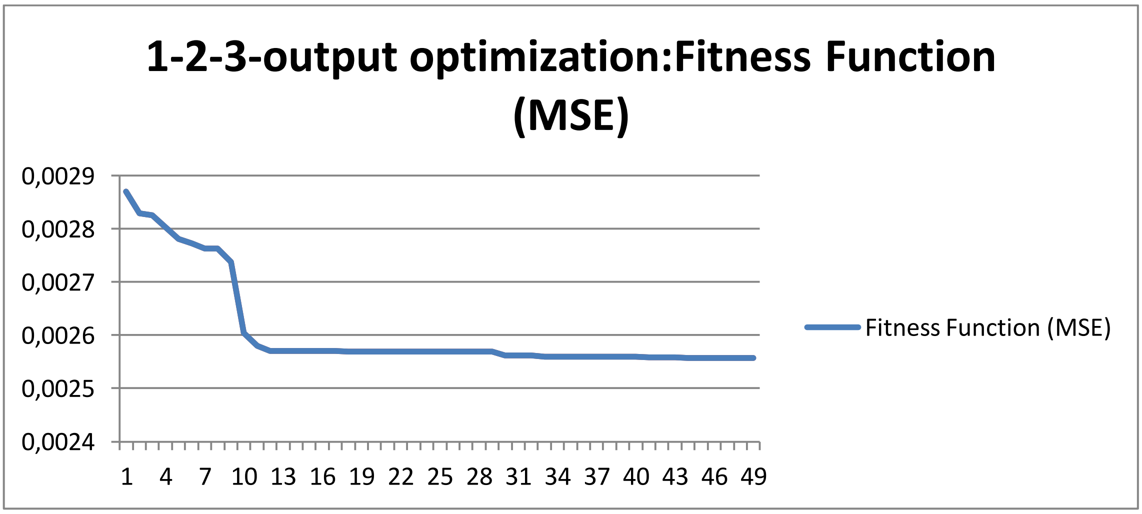
The results presented in Table 13 illustrate that the larger the number of design variables collectively combined, the greater impact they have in minimizing the fitness function. There is also one remarkable phenomenon regarding the optimization process. The best fitness function value appears for the same combination of design variables in both cases presented in the previous table. The combination of input variables 1 and 2 with the output variable produces the minimum fitness function value regardless the technique followed.

The speed at which the algorithm converges to these fitness function values is related to the population size set in the code. The number of generations, which does not have an impact on the speed, allows the algorithm to find the best set of design variables within the population. Eleven identical numbers in the last section of the graph illustrate that the algorithm has reached an optimal solution. The reason behind this number lays on the parameters of tolerance and generation limit. The stall generation limit has been set as 10 whereas the genetic algorithm tolerance is 10^{-15}. Namely, if the improvement of the algorithm is below 10^{-15} for 10 generations, the algorithm stops. The way in which the genetic algorithm converges through generations to a minimum fitness value is illustrated in the graphs below.



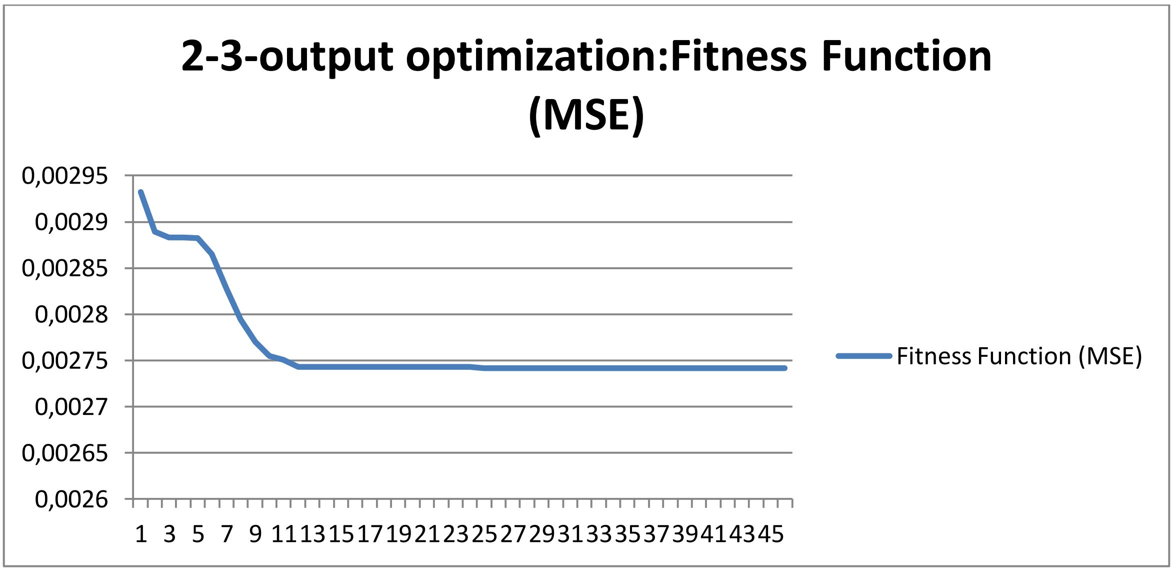
Graph 1. Output optimization convergence

The results presented above represent the best effort of optimization achieved during this thesis. The genetic algorithm started satisfactory, with a fitness function value of 0.00271619. The algorithm stopped when the fitness function converged to the value 0.00239333 during the last 10 generations.



Graph 2. Output optimization convergence

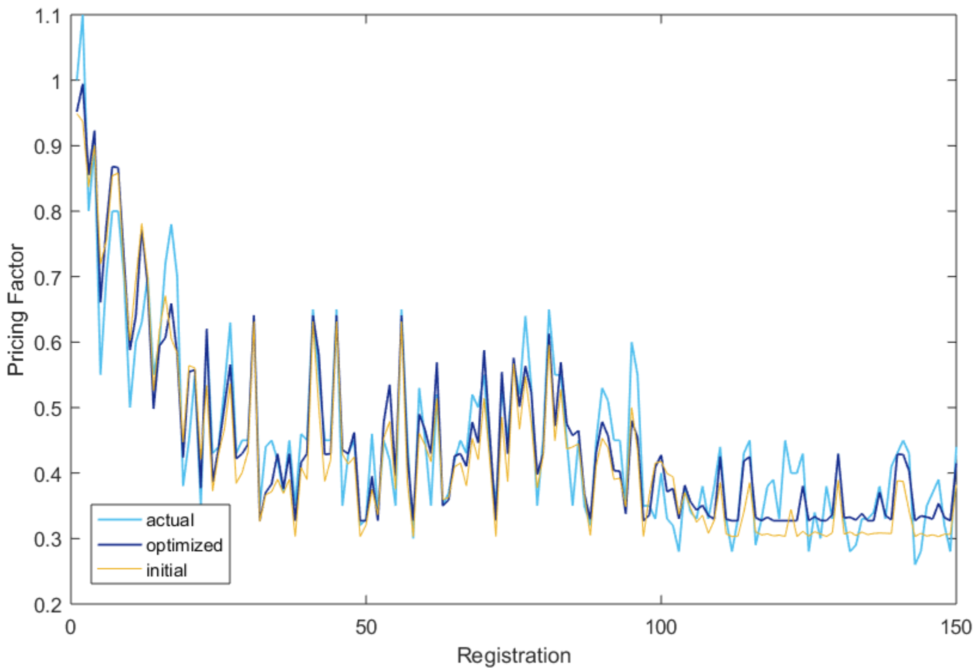
The shape of each graph is unique, as it represents the path followed by the genetic algorithm during the search of an optimal solution. The way in which each algorithm converges is slightly different even in cases of optimizing the same design variables. This phenomenon is attributed to the random nature of the genetic algorithm. The following graph, which is also unique in terms of shape, depicts the optimization of inputs 2 and 3 combined with the output.



Graph 3. Output optimization convergence

7.4. Comparison between initial and optimized fuzzy systems

The following graph illustrates the outputs produced by the best fuzzy inference system created by the optimization process along with the actual output values of the initially provided real-time anonymous insurance policy pricing dataset.



Graph 4. Comparison between initial and optimized fuzzy approximations

As illustrated in the graph, there is noticeable difference in the way in which the two fuzzy inference systems approximate actual values provided by the dataset. The optimized fuzzy inference system approximates actual insurance policy pricing factors with greater precision. The variations between the optimized and the initial fuzzy inference system may seem negligible. Nevertheless, a small alteration in the way in which credit insurance price factors are approximated can have an enormous impact on the prices that a prospectively insured company will pay. The new optimized fuzzy inference system offers 35.1% greater approximations compared to the initially designed fuzzy inference system.

8. Conclusions

In the beginning of this thesis, various questions were raised regarding calculations of credit insurance price factors. The first question raised was whether a control system could be used in the field of credit insurance. According to the illustration of how fuzzy controllers work in Chapters 1 and 2, the answer was positive. Next, the concern of whether the way in which the human brain functions can be depicted in a fuzzy inference system occurred. The answer to this question lies in Chapter 5, where different versions of fuzzy inference systems were created. The question of how accurately the insurance broker expert human brain could predict pricing factors according to which insurance policy costs were issued was then raised. For the evaluation of the fuzzy inference system another type of model was necessary, namely the neurofuzzy inference system that was also created in Chapter 5. The difference between the two types of systems is that the fuzzy inference system is designed considering the information provided by credit insurance experts whereas the neurofuzzy inference system is designed based on a dataset with real-life credit insurance policy prices. In Chapter 7 comparisons of four different fuzzy versions with the adaptive neurofuzzy inference system that was automatically designed based on the dataset provided were presented. The evaluation of the fuzzy inference systems designed showed that a fuzzy system with more output categories and smoother membership functions is more likely to produce accurate approximations of real-time conditions than fuzzy systems with a different design. The neurofuzzy inference system was expected to produce more accurate results compared to the fuzzy inference system, however this was not observed in this particular thesis. This can be attributed to the limited data available in the field of credit insurance. More data could be used for the improvement of the neurofuzzy inference system; however, this was beyond the scope of the present thesis. The last question was whether the theoretically designed fuzzy inference system could be improved in order to approximate all 150 registrations provided in the dataset more accurately. This was examined during the optimization process, whose basic concepts are introduced in Chapter 3. All parameters and hindering factors faced during the optimization process were analyzed in Chapter 6. Different efforts of optimizing the initial fuzzy inference system variables were made. The answer to whether the initial, theoretically designed by insurance experts system could be improved in order to approximate real-time conditions faced by companies was positive. The optimized fuzzy inference system offers 35.1% greater precision to the way in which pricing factors are calculated and it can prove a valuable tool in the hands of credit insurance brokers.

References

Brkic, S., and Hodzic, M. and Dzanic, E., 2017. Fuzzy Logic Model of Soft Data *Analysis* for Corporate Client Credit Risk Assessment in Commercial Banking. In: Fifth Scientific Conference with International Participation “Economy of Integration” ICEI 2017, December 2017. MPRA Paper No. 83028 (10 pages)

Calibo, D.I., Ballera, M.A., 2017. A Fuzzy Logic Model for Risk Analysis and Recommendation System of a Government Financial Granting Project, *International Journal of Simulation Systems*, Paper 19: (7 pages). DOI: 10.5013/IJSSST.a.20.S2.19

Dorf, R.C. and Bishop, R.H., 2017. Modern Control Systems, 13th Edition. Pearson, London.

DeWit, G.W., 1982. Underwriting and Uncertainty. *Insurance: Mathematics and Economics*, 1:277-285

Fullér Robert, 1995. Neural Fuzzy Systems, Abo Akademi University

Jang, J.-S. R., Sun, C.-T., 1995. Neuro-fuzzy modelling and control. *Proceedings of the IEEE*, 83(3):378-406

Jang, J.-S. R., Sun, C.-T., 1997. Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997.

Marinakis Y., Evolutionary and nature-inspired algorithms, University lectures in Greek, Technical University of Crete, School of Production Engineering and Management., Chania, 2019.

Michalewicz, Z., 1996. Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, Berlin Heidelberg.

Moorcraft, B., 2018. What is trade credit insurance? *Insurance Business Asia Magazine*. Available at: https://www.insurancebusinessmag.com/asia/news/breaking-news/what-is-trade-credit-insurance-116058.aspx (Accessed 15 July, 2020)

Pfeifer, R., 2013. Crash course in Fuzzy Logic. In “Real-world computing” Lecture Notes, University of Zurich, AI lab.

Shapiro, A.F., 2005. Insurance applications of Fuzzy Logic, Institute of Actuaries 2005 Biennial Convention, 8-11 May, Paper 050215 (33 pages).

Shapiro A.F., 2007. An Overview of Insurance Uses of Fuzzy Logic. In: Chen SH., Wang P.P., Kuo TW. (eds) Computational Intelligence in Economics and Finance. Springer, Berlin, Heidelberg

Sokolovska, O., 2017. Trade Credit Insurance and Asymmetric Information Problem, *Scientific Annals of Economics and Business*, 64(1):123-137 DOI: 10.1515/aicue-2017-0008

Sreekantha, D.K., Kulkarni, R.V., 2010. Expert system design for credit risk evaluation using neuro-fuzzy logic. *Expert Systems*. DOI: 10.1111/j.1468-0394.2010.00562.x

Tairidis, G.K., 2016. *Optimal design of smart structures with intelligent control*. Ph.D. Dissertation, Technical University of Crete.

Tairidis, G.K., Foutsitzi, G., Koutsianitis, P. and Stavroulakis, G.E., 2016. Fine tuning of a fuzzy controller for vibration suppression of smart plates using genetic algorithms. *Advances in Engineering Software*, 101:123-135

Tairidis, G.K., and Stavroulakis, G.E., 2019. Fuzzy and neuro-fuzzy control for smart structures. In M. Khakifirooz, M. Fathi and P. Pardalos (Eds.), *Computational intelligence and optimization methods for control engineering*. Springer: Cham

Widrow, B., Lehr, M. A., 1990. 30 years of adaptive neural networks: Perceptron, madline, and backpropagation. *Proceedings of the IEEE*, 78:1415-1442.

Yazdani, H. and Kwasnicka, H., 2012. Fuzzy Classification Method in Credit Risk. In: Nguyen NT., Hoang K., Jędrzejowicz P. (eds) Computational Collective Intelligence. Technologies and Applications. ICCCI 2012. Lecture Notes in Computer Science, vol 7653. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-34630-9_51

Zadeh, L.A., 1965. Fuzzy Sets, *Information and Control*, 8(3):338-353