ABSOLUTDATA

BrainWave

DATA SCIENCE DIGEST

3RD EDITION

Rajat Narang
Director, Absolutdata Labs

Dear friends,
Warm greetings from the Absolutdata Labs!
In our efforts to constantly learn, innovate, and share our learnings in the field of data science with our organization and industry alike, we've delved into another fascinating subject – fuzzy logic.

What Is Fuzzy Logic?

Humans, when posed with a question, often have replies that do not belong to a Boolean extreme of 'Yes' or 'No'. Instead, the answer lies somewhere in the continuum between truth and fallacy. Such degrees of truth and fallacy form the very basis of fuzzy logic. In contrast to Boolean logic, fuzzy logic caters to this flexibility in reasoning, addressing intermediate values like partial truths. A simple example could be asking 'Is it hot today?' The answer most likely would not be a direct 'Yes' or 'No'; it would be 'a little hot', 'extremely hot', etc.

When Do We Use Fuzzy Logic?

Across domains, we are often faced with such uncertainties while solving non-linear (not sequential or straightforward) narratives. Fuzzy logic is largely useful in scenarios where there are if-then possibilities instead of logical binaries. This helps us program machines to work in a more human way, i.e. by considering quite a lot of subjective attributes. Fuzzy logic does this by using intuitive linguistic variables to develop effective models for highly complex non-linear problems.

Application of Fuzzy Logic in Daily Life

Think of how an air conditioner (AC) controller works. Following binary logic, the controller would switch the AC on or off (depending on the temperature of the room and the temperature selected by the user). But what if the user changes the temperature too frequently? Should the AC controller just change the temperature slightly? Should direct sunlight hitting the room make the AC controller lower the temperature a bit more than usual? To put it simply, there are many if-then conditions that need to be taken into account. Fuzzy logic does that by ultimately emphasizing the degree of change (increase or decrease) based on various linguistic variables.

A rice cooker is another example. Here the mathematical program helps provide a range of customized cooking options. If it is a cold day and the rice is cooking too slowly, fuzzy logic enables the cooker to increase heat exposure. The degree of increase or decrease (in terms of 'little', 'too much'

Fuzzy logic in washing machines also has become quite popular; different features like water intake, wash time, spin speed, and water temperature are all auto-controlled by fuzzy logic. This has assisted in optimizing the machines' functionalities in several ways, like increasing their lifetime, making them more efficient, and reducing running costs.

Fuzzy logic has also been applied in numerous machine learning algorithms, particularly those for clustering and classification. For instance, in fuzzy clustering, a data point can belong to more than one cluster, with a certain degree of membership in each. This method is best suited for datasets that aren't clearly separable, where segments could overlap with one another to a certain extent. Fuzzy decision trees, on similar lines, make fuzzy rule-based decisions at each branching node.

Learning Fuzzy Logic

There is a whole slew of online courses available for learning fuzzy logic and its applications. These include courses by Udemy, the SWAYAM National Programme on Technology Enhanced Learning (NPTEL), etc. Since the application of fuzzy logic with machine learning has gained a lot of traction recently, data scientists and machine learning engineers can leverage such accessible opportunities to expand their knowledge.

Our Favourite: Fuzzy Logic in Python!

To apply fuzzy logic using Python, one can utilize the scikit-fuzzy package. This package's documentation presents problems like fuzzy C-means clustering and the basic tipping problem for reference. The tipping problem illustration provides a good way to simulate a fuzzy control system in Python. Fuzzy string matching is often seen in search engines, where users frequently enter words that partially match the actual word (or users misspell the actual word); it's based on approximate string matching. This can be done in Python using the FuzzyWuzzy package.

Current BrainWave Edition in a Nutshell

In this edition of BrainWave, we cover different aspects of fuzzy logic. We start with the basics, move to what it is really composed of, and discuss its application in different fields like medicine. We also shed some light on how conventional machine learning techniques can be blended with fuzzy logic to produce more generalized results.

Rajat Narang
Director, Labs at Absolutdata

Contents

Overview

“Fuzzy”, in one sense, refers to things that aren’t quite clear: fuzzy vision, fuzzy thinking, etc. In computer terms, “fuzzy” is the opposite of true/false Boolean logic; it exists on a continuum between one absolute and another.

We live with the concept of fuzziness every day: Clothes can be wet, dry, or of varying degrees of dampness; colors can be red, purple, or dozens of shades in between; lights can be bright, dim, or almost dark, and so on. So, fuzzy logic is the “something in between” logic that exists between absolute conditions. It’s based on degrees of truth rather than 100% true or 100% false (i.e. the Boolean logic so common in computer science). [1]

Fuzzy Logic in Real Life: Washing Machines

Fuzzy logic is leaving the lab and finding many practical applications. Let’s take just one: the washing machine. Washing machines with this feature not only have better performance, they are simple to use, generate less costs, and may even last longer.

What part does fuzzy logic play in the clothes-washing experience? From the Samsung website, we learn that fuzzy logic determines the washing process, including water temperature and levels, agitating time and type, rinse speed and time, and spin speed and time. The fuzzy logic system determines the optimal amount of detergent and water to use (along with many other factors) on each load based on the soil type (i.e. how dirty the clothes are). Optical sensors can even be used to detect the amount of dirt in the water, and a fabric sensor and weight sensor make additional adjustments. [2] A simplified version of this process is illustrated below:

Input: cloth type, soil level, soil type, combined mass of clothes, etc.	Fuzzification of input factors	Application of fuzzy inference rules	Defuzzification	Output: Wash time and type, rinse time, spin speed and ime, etc.

Figure 1: Fuzzy logic in a washing machine, simplified.

Interestingly, fuzzy logic mirrors how our brains tend to work, making it essential to the development of AI’s human-like capabilities (e.g. self-learning, problem solving, etc.). This influences how neural networks and other AI applications are utilized.

Advantages

The main advantages of fuzzy logic are:

- It's often preferable to purely mathematical or symbolic approaches, as most states exist on some kind of continuum.
- It allows us to measure uncertainty and vagueness .
- It's usually simpler, faster to develop, and less computationally demanding than the alternatives.
- Fuzzy algorithms tend to be more robust and less sensitive to change.

(Adapted from What are the main advantages of applying fuzzy-based decision making?[3])

Disadvantages

The main advantages of fuzzy logic are:

- It is often based on assumptions, making the results sometimes inaccurate.
- Fuzzy systems don't have the option to use machine learning or neural-network-type pattern recognition.
- Defining membership functions is difficult, as is setting exact rules.
- Fuzzy time logic terms can be confused with probability theory terms.
- The validation and verification of a fuzzy logic system requires extensive testing.

(Adapted from Fuzzy Logic Tutorial: What Is, Application & Example[4].)

References

[1] SearchEnterpriseAI. Definition: Fuzzy Logic

[2] Samsung. What is Fuzzy Logic in a Washing Machine?

[3] Researchgate. What are the main advantages of applying fuzzy-based decision making approaches in comparison with the non-fuzzy-based ones?

[4] Guru99. Fuzzy Logic Tutorial: What is, Application & Example

Figure 1: Agarwal, Manish. Fuzzy Logic Control of Washing Machine, Figure 1

Tutorialspoint. Fuzzy Logic - Control System

ResearchGate. State Model of a Washing Machine

-Authored by
Shivli Gupta,
Data Scientist at Absolutdata

Fuzzy Time Series Using the pyFTS Library

Coder's Cauldron

Overview

A friend asks you how the weather is. Your reply depends on the season, right? Hot or cold. But will it always be either hot or cold? No. Sometimes it will be “somewhat hot”; sometimes it will be “too cold”. Our mind does not see only two extremes. There is always some middle ground.

That’s exactly how fuzzy logic works. Each point in our universe of discourse can simultaneously belong to multiple categories, which can also be referred to as ‘fuzzy sets’. The association to each fuzzy set is defined by the membership grade, which depends upon the membership function. The triangular membership function is most commonly preferred.

Recently, there has been a lot of progress in the field of fuzzy time series forecasting. We have the introduction of some new libraries like pyFTS (developed by the Federal University of Minas Gerais (Brazil) with the Federal Institute of North of Minas Gerais and the Federal Institute of Minas Gerais). But why use fuzzy logic in time series forecasting?

The basic concept of fuzzy time series is to divide the universe of discourse into multiple intervals and then decipher the rules on how these intervals relate to each other with time. Let’s use the example of time series forecasting for Alabama University enrolment data to understand this in more detail. A link to the code can be found at the end of the article.

Methodology

The following steps are involved in the training:

1. Defining the universe of discourse

We first define our universe of discourse, which is generally the lower and upper value of the Y variable, with mar gins usually set at 20%. In this case, the bounds will be defined by the minimum and maximum enrolments across the years.

2. Partitioning

After defining the universe, we divide it into multiple overlapping intervals. These become our fuzzy sets. There are several methods of partitioning; the easiest one is grid partitioning, in which all intervals have the same length.

3. Fuzzification

This step involves transforming numerical values to the fuzzy sets that were defined in the last step. We should remember that each y value can belong to multiple fuzzy sets, but for simplification, we can assign the interval with the maximum membership grade, e.g. Y1 puts 0.8 in set A0 and 0.2 in set A1. We will assign it A0.

4. Creating patterns

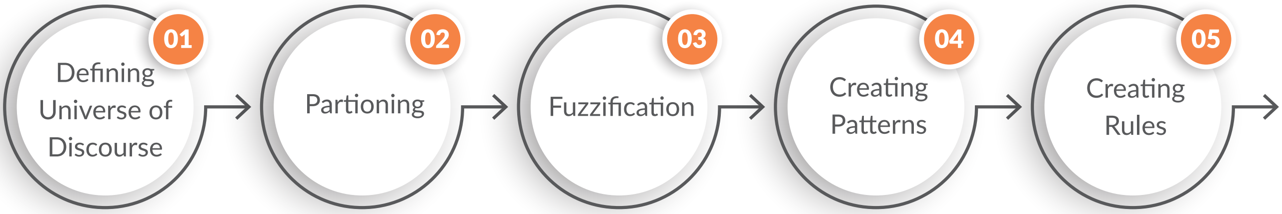
We follow the Precedent -> Consequent format. When all Y values are fuzzified, they will form a series, e.g. A0 – A1 – A1 – A2 – A3 – A4 – A4 – A5, etc. Whenever we get similar sets at f(t) and f(t+1), we can define some patterns.

5. Creating rules

Based on the patterns from the previous step, the model will create rules. For example:

A0 -> A1 | A1 -> A1, A2 | A2 -> A3 | A3 -> A4 | A4 -> A4, A5

These rules form the backbone of our model. They will be used to forecast unseen data.



After training the model, it's time to make predictions. The steps below will be followed:

1. Input value fuzzification

As discussed earlier, the numeric y value will be assigned to the fuzzy set. We have a y value for T=t and we want to predict for T=t+1. For example, f(t) belongs to A4.

2. Finding compatible rules

Now we will select the most appropriate rule from our list, e.g. since f(t) = A4, we will use the rule A4 -> A4, A5.

Defuzzification

Since we need the numerical values in the final output, we will convert these sets back to their original type. This will be equal to the mean of the centers of the fuzzy sets.

Now you can understand that rule readability makes it highly efficient. For example, this could tell an investor if "moving average" is low and "Relative Strength Index" is low, they should sell!

Illustration

Problem Statement:

Time series forecasting of Alabama University Enrolment Data using pyFTS library. More about the library at https://pyfts.github.io/pyFTS/build/html/index.html.

Data

We will use Alabama University's enrolment data over the past several years.

Albama University Enrollment Data.csv

Code
(Google Colab, taken from external source)

References:

Petrono Silvia, Ph.D. A Short Tutorial on Fuzzy Time Series

Chen Shyi-Ming. Forecasting enrollments based on fuzzy time series

-Authored by
Rhydham Gupta,
Data Scientist at Absolutdata

Deciphering the 4 main Components of Fuzzy Methodology

Vivid Visualization

Overview

Interpretation alone doesn't necessarily lead to an accurate or wise **judgement**. There are usually many options between **yes** and **no**. This broad spectrum of options, which are not fully yes and not fully no, can range anywhere between those two absolutes. This is the concept of fuzzy logic; the **degrees of our answers** makes it all **fuzzy** or vague.

The architecture of fuzzy logic is depicted in the figure below. The numbers corresponding to different features of this system will be referenced throughout this article.

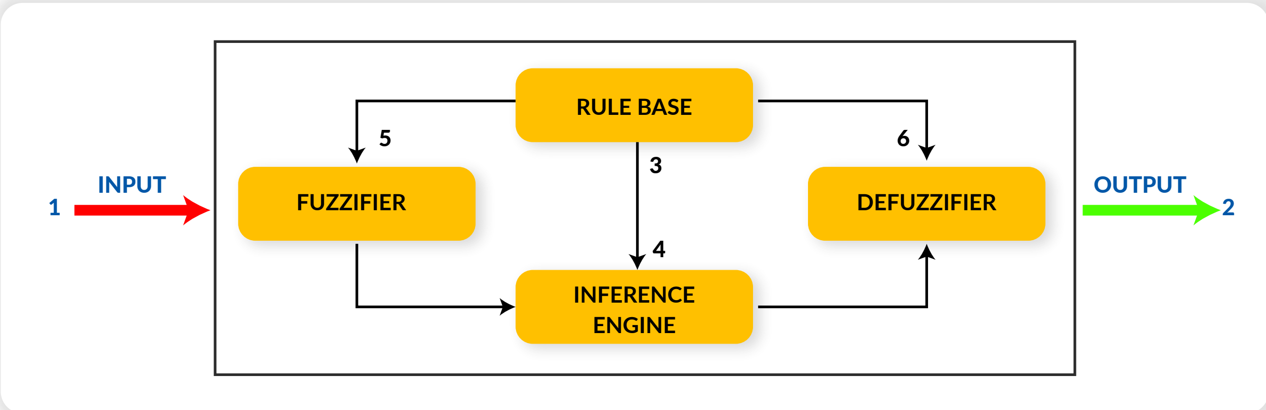
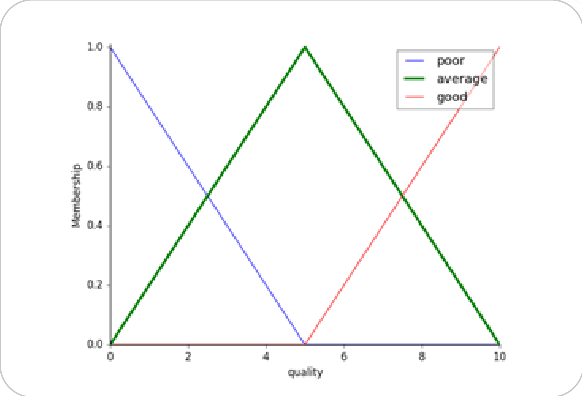


Image1 adapted from Geeks for Geeks [1]

Fuzzy logic helps us model the uncertainty and vagueness in the basic human responses which serve as our inputs [1]. All our inputs have a characteristic of **degree of membership**, in that we approximate the association of an input to a fuzzy set. The value of this measure is anywhere between 0 and 1; 0 indicating no association to a certain fuzzy set and 1 meaning that it belongs to this fuzzy set. Any value between 0 and 1 is treated with uncertainty [2]. The .view attribute in the **skfuzzy** Python library helps us view different elements modelled in fuzzy logic [3]. We will use this attribute to visualize our use case.

The Tipping Problem (Use Case)

Taking the example as implemented in the .ipynb file[3], we will visualize the functionality of a fuzzy logic system. Here we have to determine the 'tip' **(2)** with the help of ratings given on the 'service' and 'quality of food' **(1)**. The ratings for both our inputs could vary between 0 and 10, however we use **(5)** to convert our numeric values to fuzzy sets (labelled poor, average, and good). Then we can view the associated degree of membership in adherence to our varying inputs:



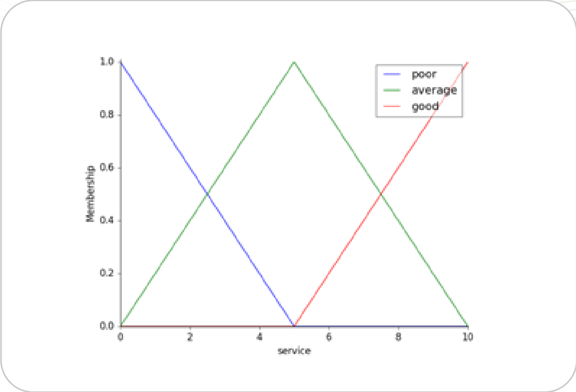


Image2, Image3 from skfuzzy [3]

Once our inputs have been defined, a relationship needs to be defined between **(1)** and **(2)**, which then forms our rule base **(3)**.

The rules for the tipping problem could be a good tip for good service and food, an average tip for average service, and a poor tip for either poor food or poor service. The complexity of rules really depends on how robust a controller we want to design.

Fuzzy logic is excellent at translating these imprecise rules into a form that gives the controller an actionable insight[1], i.e. a discrete value of the tip. The .view() attribute allows us to visualize the rules interpreted by the controller. See the figures below:

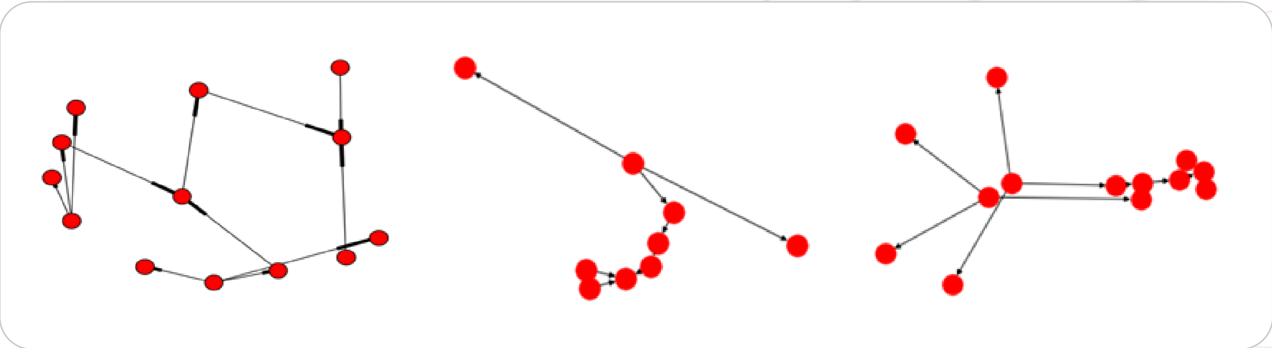


Image4 adapted from skfuzzy [3]

The interpretation made by the controller to learn the association of the inputs and the outputs is called the **inference engine (4)**. This plays a role in deciding the action made by the controller.

After the above steps have been followed, we have a complete fuzzy-enabled control system which can now be used to simulate a response to new numerical inputs, i.e. '9' for service and '6.8' for food. The black line shows the defuzzified **(6)** final output, i.e. the tip in numerical terms.

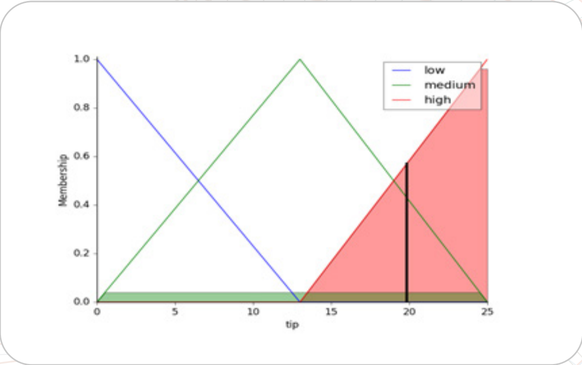


Image5 from skfuzzy [3]

'Click here for the Jupyter Notebook with the implementation of the same in Python.'

References

[1] Fuzzy Logic | Introduction

[2] The Fuzzy Systems Handbook A Practitioner's Guide to Building, Using, and Maintaining Fuzzy Systems

[3] Fuzzy Control Systems: The Tipping Problem

-Authored by
Mayank Lal,
Data Scientist at Absolutdata

Neuro-Fuzzy Systems

Thriving Traction

Overview

Neural networks are a set of algorithms inspired by the structure of the human brain's neural network; they attempt to mimic how our brains function. Neural networks have the potential to learn any complex relationship between input and output, as they are universal function approximators.

Thanks to their exceptional performance in solving complex problems (e.g. pattern recognition, image classification, video processing, speech recognition, and natural language processing), such networks are revolutionizing AI.

Next, we have fuzzy logic. This is an AI technique that simulates human reasoning. "Fuzzy" in this context refers to a human-like 'fuzziness' in decision-making, i.e. including all the intermediate states between 100% true and 100% false (like mostly true, partially true, partially false, and mostly false).

Building a high-performing fuzzy system is challenging: identifying membership function and inferential rules are dependent on domain experts (human intervention). Thus, the idea arose that neural networks can be used as alternative learning algorithms for the automation and development of fuzzy systems. These systems, where neural network and fuzzy logic are used together, are known as neuro-fuzzy systems or fuzzy neural networks.

There are several ways to integrate neural networks and fuzzy logic. They are divided into three main categories:

Cooperative Neuro-Fuzzy System

In a cooperative system, the neural network and the fuzzy system work independently, as separate blocks. The neural network is used as a preprocessor, using training data to deduce membership functions and/or inferential rules. After deducing the fuzzy inferential parameters, the neural network block is taken away and only the fuzzy system block is executed. One of the major drawbacks of cooperative systems is that they lack interpretability. This is due to the black-box nature of neural networks.

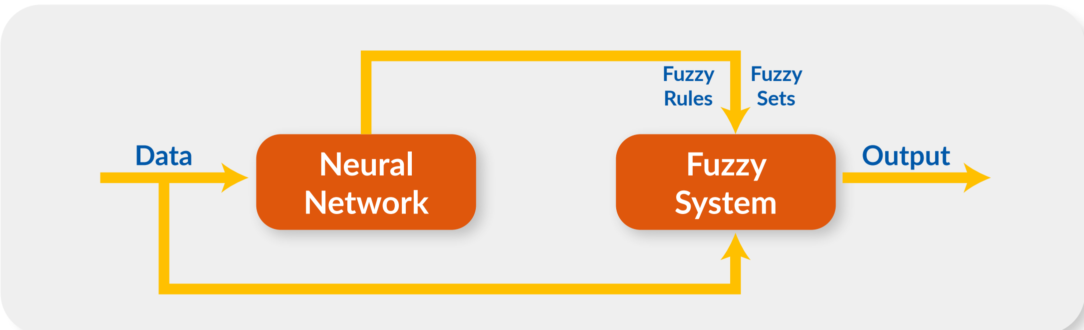


Figure 1. Cooperative System [1]

Concurrent Neuro-Fuzzy System

In concurrent systems, the neural network and the fuzzy system work together to continuously deduce the required set of parameters. Here the objective is to improve the performance of the entire system rather than optimizing the fuzzy system. In some cases, the neural network acts as an assistant to the fuzzy system by preprocessing the inputs. In other cases, it post-processes the fuzzy system's output. Like co-operative systems, concurrent systems are not fully interpretable due to the presence of the neural network.

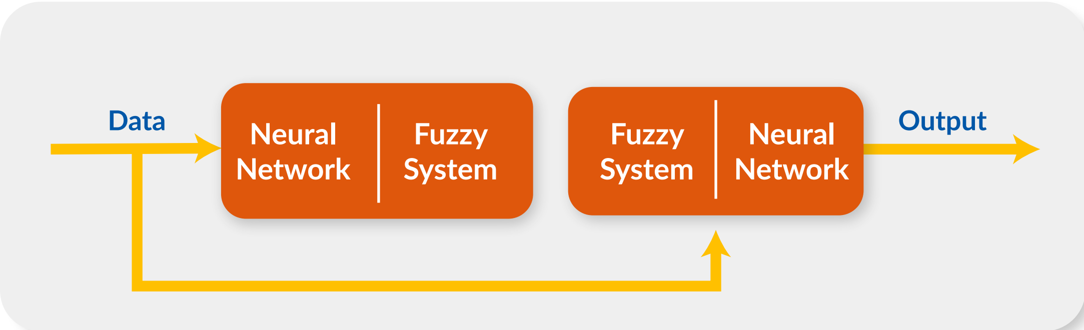


Figure 2. Concurrent System [1]

Hybrid Neuro-Fuzzy System

In hybrid systems, the fuzzy system is a special kind of neural network. In this system, the neural network and fuzzy system no longer work as separate blocks but are one fused entity: the neural network learning algorithm is used to ascertain the parameter of the fuzzy inferential system (fuzzy rules and fuzzy sets) from input data in an iterative way.

The hybrid system can be used to generate parameters in two ways – either by entirely using input-output data or by initializing with à priori knowledge (human experts). This fusion of neural networks and fuzzy systems has the advantage of learning in a supervised way. Also, their functionality can be easily interpreted.

Hybrid neuro-fuzzy systems can be created in several ways. Popular architectures include Fuzzy Adaptive Learning Control Network (FALCON), Adaptive Network-based Fuzzy Inference System (ANFIS), Generalized Approximate Reasoning-based Intelligence Control (GARIC), Neuronal Fuzzy Controller (NEFCON), and Self-Constructing Neural Fuzzy Inference Network (SONFIN).

Conclusion

In this section, we discussed different ways to learn fuzzy parameters using neural networks. Data acquisition and the preprocessing of input data is essential to the improved performance of neuro-fuzzy systems. For high-performing neuro-fuzzy systems, the major requirements include fast learning, on-the-go adaptability, reduced computational expense, and achieving a global minimum for error.

Credits:

[1] Image adapted from Neuro-Fuzzy Systems: A Survey

References

- Neuro-Fuzzy Systems: A Survey
- Adaptation of Fuzzy Inference System Using Neural Learning

-Authored by
Sunny Verma,
Data Scientist at Absolutdata

Is There a Need for Fuzzy Logic?

Folk-Wisdom's Fallacy

Since the conception of fuzzy logic, there have been many doubts about its usability. People have been raising questions like:

- Is there a need for fuzzy logic?
- Doesn't fuzzy logic contradict classical logic?
- Isn't fuzzy logic inconsistent?

In this section, we'll focus on the question of usability and the need for fuzzy logic.

Fuzzy Logic in Medicine

Fuzzy logic is a logic of imprecision and approximate reasoning. Put another way, fuzzy logic helps you provide reasoning for vague concepts.

Fuzzy logic is not a new idea; it is been with us for a very long time. In spite of this, the utility of fuzzy logic has always been questioned. Its usability is mostly restricted to linguistic variables and the associated machinery of fuzzy if-then rules. But, in fact, there are many other areas where fuzzy logic can potentially help us.

Below are some of the use cases where fuzzy logic is currently being used and could be of great importance.

Fuzzy Logic Use Cases

Linguistic variables and fuzzy if-then rules

A linguistic variable is a variable whose value is a word in a natural language. For example, "temperature" is a linguistic variable; it can take values like "very hot", "hot", "moderate", etc. Fuzzy if-then rules are simple conditions that associate values to a linguistic variable. The combination of a linguistic variable and fuzzy if-then logic is continuing to play an important role in control system and consumer product design. This is one of the areas where fuzzy logic is widely used and accepted.

FL generalization

Fuzzy logic has significantly higher generality than bivalent logic. Bivalent logic means that the system has exactly two values, e.g. true and false. The problem with bivalent logic is that human cognition has no clear separation between the classes/output, e.g. there is no clear boundary defining hot and moderate weather – they overlap. This problem can be resolved by using fuzzy logic to generalize bivalent logic. The generalization of bivalent logic helps us to construct a better model of reality.

Natural Language (NL) Computation, Computing with Words (CW)

There is a plethora of information available to us in the form of natural language, but the problem with natural language is that it is imprecise. Since bivalent logic is inflexible and unable to handle imprecision, fuzzy logic provides a much more reliable way to process this information.

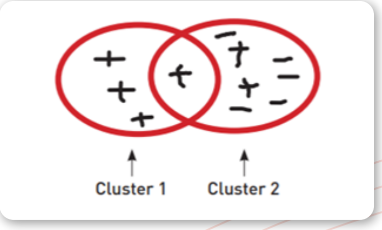
Possibility theory

Possibility theory, a subdivision of fuzzy logic, is an uncertainty theory devoted to the modeling of incomplete information. Possibility theory provides the basic framework for preference modeling. In the past, this theory has not been used very significantly. However – based on its connection to symbolic artificial intelligence, decision theory, and imprecise statistics – it will likely be of great importance in the near future.

Fuzzy logic as modelling language

Machine Learning algorithms are widely used for information extraction and inference; when used in combination with fuzzy logic, they provide much more flexible results. Clustering and classification are the two major use cases where fuzzy-based machine learning algorithms are already providing good results.

- **Fuzzy clustering algorithms –** In fuzzy-based clustering (also known as soft clustering) each data point or object can belong to more than one cluster (instead of belonging to one cluster, as in the classical clustering algorithm). There are already many fuzzy clustering algorithms available, i.e. fuzzy c-means clustering, possibilistic c-means clustering, etc.



- **Fuzzy classification** – In fuzzy classification, a supervised classification algorithm is used in conjunction with fuzzy logic to produce much more reliable output. Neuro-fuzzy models are an example of this approach, which has been applied in fields such as disease prediction, risk analysis, etc.

As mentioned above, there are several fuzzy logic implementations that cater to different use cases in solving analog and digital problems. This is owing to fuzzy logic's ability to approximate non-linear ambiguous behavior into numbers that make logical sense. More complex adaptations of fuzzy logic are powerful enough to see their usage in modern applications.

In other words, there is a need for fuzzy logic.

References:

PDF Fuzzy Based Machine Learning: A Promising Approach

PDF Is there a need for fuzzy logic?

PDF Possibility Theory and its Applications: Where Do we Stand ?

PDF linguistic-variable

-Authored by
Pranav Sharma,
Data Scientist at Absolutdata

Fuzzy String Matching

Experience Extended

Introduction

As a data scientist, you extract data from various sources: publicly available APIs, scraped from web pages, or directly asking for it. Data in the real world can be messy, and dealing with messy datasets can be painfully frustrating and time-consuming.

We can use fuzzy string matching to deal with the challenges of such datasets. First, let's define fuzzy matching, also known as approximate string matching. It is the process of calculating the similarity between strings, based on parameters like **Levenshtein distance, Jaccard distance, and cosine similarity.**

Why Fuzzy Matching is Important

Most businesses – over 90% -- have duplicate data. Most of these duplicates are non-exact matchings, which means they are not automatically detected. Using fuzzy matching and sophisticated matching techniques, it's possible to uncover typos, incomplete information, and other kinds of non-exact matches. The technical terms are deduplication (matching similar entries, such as 'yogurt' and 'yoghurt', into one entity) and record linkage (joining record for 'Tesla' with the URL of that Wikipedia page). [1]

While the applications of both data deduplication and record linkage vary widely, they are two sides of the same coin. This is because the underlying technology used to identify matching records for both is the same. Where might we use fuzzy matching? Here are just a few areas:

1. Identifying duplicate customer records
2. Fraud analytics
3. Price comparisons
4. Cleaning messy data
5. Integrating data sources
6. Finding cross-sell and upsell opportunities
7. Data enrichment and reference matching

Scalability Issues

There are many ways to perform fuzzy matching, but their performance takes a big hit as the size of the dataset increases. The reason is that each record is compared against all the other records in the dataset. As the size of our dataset increases, the time taken to perform fuzzy matching increases quadratically; this phenomenon is known as quadratic time.

The performance of fuzzy matching is also dependant on the length of the strings being compared. Therefore, we must think about scalability when using fuzzy matching. We can use TF-IDF (Term Frequency, Inverse Document Frequency) to deal with such problems. It filters useful words from the noise present in the text. This method generates features from the text by multiplying the frequency of a term in a document by the importance of the same term in an entire corpus.

TF-IDF is a very useful algorithm in text classification and text clustering, as it can be used to transform documents into numeric vectors that can easily be compared. You can also use n-grams to deal with the scalability issue. For further reading on fuzzy matching, TF-IDF, or n-grams, please see the links in the references section below.

References

[1] Elahi, Ehsan. Fuzzy Matching 101: Cleaning and Linking Messy Data Across the Enterprise

Rosa, Denis. What Is Fuzzy Matching and How to Use It Correctly

van den Berg, Chris. Super Fast String Matching in Python

Ostertag, Christoph. Build a Scalable Search Engine for Fuzzy String Matching

-Authored by
Arnav Agarwal,
Data Scientist at Absolutdata

Fuzzy Logic in Medicine

Food for Thought Experiment

Overview

Uncertainty is inherent in the field of medicine and bioinformatics. There are a lot of "fuzzy areas"; there may be little information (or mistaken information) about the patient, lab results can be wrong, family medical history can be imprecise, etc. Humans are also facing an increasing number of mental disorders and no structured classification. A single disease, e.g. COVID-19, can affect different patients with different intensities, depending on the geographic and demographic characteristics.

Since fuzzy logic plays with uncertainty, it appears to be an ideal way to deal with many of the shortcomings found in medical information.

Providing explicit descriptions of medical concepts and the relationship between concepts is very difficult. In quite a lot of cases, the partitions are fuzzy. Human thinking and decision-making are fuzzy, with a high degree of vagueness in proof and concept application. There is a high level of vagueness in many statements, e.g. "if an elderly person has severe headache, acupuncture therapy is administered for some time". This statement consists of vague terminologies: "elderly", "severe"," some". For these kinds of fuzzy statements to be deciphered, the use of soft computing (SC) methodologies is advised.

Fuzzy Logic in Medicine

According to the World Health Organization (WHO):

"Health is a state of complete physical, mental and social well-being and not merely the absence of disease or infirmity." [1]

The loss of health can be seen in three forms: disease, illness, and sickness.

We can paraphrase the explanation given by Torres and Nieto in their paper, Fuzzy Logic in Medicine and Bioinformatics:

'According to Aristotelian logic, there are only two logical values: true or false, black or white, etc. In real life, most things are not simply black or white; there are many shades of grey in between. Fuzzy logic accommodates these "partial" values, for example, between absolutely true and absolutely false.' [2]

In health as in other things, there is a range of conditions. A person without any disease can have a headache or a stomachache; does this mean they are no longer healthy? Or do we measure health only in terms of serious conditions, e.g. "A has heart disease, so A is not healthy; B does not have heart disease, so B is healthy"? Paraphrasing the paper Evaluation of the Risk of Drug Addiction with the Help of Fuzzy Sets:

'Everybody is healthy to some degree and sick to some degree ... If someone only has a headache, we may say they are 99% healthy and 1% sick. ... Uncertainty is an indispensable part [of science], and fuzzy logic is a way to represent and deal with [such uncertainty].'

How Can Fuzzy Logic Improve Medical Information?

Fuzzy logic is currently being explored for many bioinformatics and medical applications. For example, it could be used in the diagnosis of different diseases, such as tuberculosis, cancer, diabetes, heart disease, HIV, and others.

In addition to the ongoing implementation, there are several areas where implementation could soon begin, such as diagnosing malaria and prostate disease, dealing with hepatobiliary disorder, and differentiating between various syndromes.

There are continuous efforts to improve image processing techniques using fuzzy logic to support medical decision-making. Examples of FL-supported applications are:

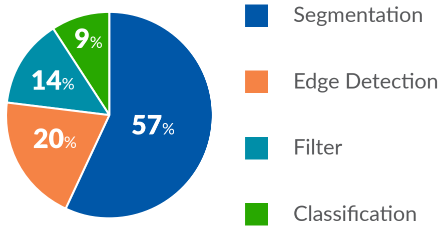


Fig. 1: Fuzzy Image Processing Applications

1. Radiographic Image Classification
2. Identifying Edges
3. Image Segmentation for Tumors
4. Motion Detection

Future work will also focus on the application of fuzzy artificial networks in medicine, evaluating parameters such as accuracy, sensitivity, and specificity.

References

[1] World Health Organization, “What is the WHO definition of health?”

[2] Torres, Angela and Nieto, Juan J., "Fuzzy Logic in Medicine and Bioinformatics"

[3] Singh et al., "Evaluation of the risk of drug addiction with the help of fuzzy sets"

Healthcare, Uncertainty, and Fuzzy Logic

Medical Applications on Fuzzy Logic Inference System: A Review

-Authored by

Manshi Poonia,
Data Scientist at Absolutdata

Data Science Competitions/ Seminars/Fora/Courses

Courses

Introduction to Fuzzy Set Theory, Arithmetic and Logic

Excerpt: "In most real-life applications of any decision making, one needs to face many types of uncertainty. As humans, we can deal with this uncertainty with our reasoning prowess. However, it is not clear how to deal with this uncertainty in a system. Fuzzy sets and fuzzy logic give us one way of representing this uncertainty and reasoning with them."

Topics covered:

1. Fuzzy sets, crisp vs. fuzzy types of fuzzy sets, membership functions, alpha cuts
2. Operation on fuzzy sets, t-norm, complements t-conorm, combination of operations
3. Fuzzy arithmetic interval, decomposition principle, extension principle, fuzzy equations
4. Relations, fuzzy relations projections, equivalence relation, transitive closure, compatibility relation
5. Propositional logic, Boolean algebra multi-valued logic
6. Inference from conditional and qualified fuzzy propositions
7. Fuzzy quantifiers, inference from quantified fuzzy propositions
8. Possibility vs. probability, belief and plausibility, Dempster's rule

Udemy: Practical Introduction to Fuzzy Logic with Matlab

Topics covered:

1. Fuzzy Logic Foundations: Fuzzy sets and membership functions; operations on fuzzy sets; union and intersection
2. Fuzzy Control vs. Classical Control: Necessity of fuzzy logic
3. Fuzzy Arithmetic Functions: discrete fuzzy function, continuous fuzzy function
4. Fuzzy Inference Systems and Fuzzy Rules: fuzzy linguistic variables, fuzzy ruled based systems

Fundamentals of Fuzzy Logic

Excerpt: "FL is a form of many-valued logic. It extends the truth values to an arbitrary degree of truth, formally a value in the interval [0, 1]. The aim of FL is to mimic human reasoning in an environment of uncertainty and imprecision (such as the real world). FL provides an intuitive approach to modelling human intelligence in machines, as it uses high-level linguistic inference."

Topics covered:

1. Motivations of fuzzy logic
2. Linguistic uncertainty
3. Fuzzy set theory
4. Fuzzy logic systems
5. Applications of fuzzy logic systems

NPTEL: Fuzzy Logic and Neural Networks

Topics covered:

1. Fuzzy sets and their applications
2. Fuzzy reasoning and optimization
3. Neuro-Fuzzy systems
4. Soft computing

Lecture Notes & Seminars

Fuzzy Logic and Systems

Topics covered:

1. Fuzzy logic
2. Fuzzy sets and systems
3. Fuzzy knowledge-based systems

Introduction to Fuzzy Logic

Topics covered:

1. Fuzzy concepts
2. Fuzzy propositional and predicate logic
3. Fuzzification
4. Defuzzification
5. Fuzzy control systems
6. Types of fuzzy algorithms
7. Applications of fuzzy logic

Applied Logic - Fuzzy Logic

PDF Applied Logic- Lecture 3 part 1 - Fuzzy logic

PDF Applied Logic- Lecture 3 part 2 - Possibilistic fuzzy logic

PDF Applied Logic- Lecture 3 part 3 - Truth-functional fuzzy logic

Topics covered:

1. Fuzzy concepts and fuzzy sets
2. Operations and linguistic rules
3. Fuzzy intersection (T-norm)
4. Fuzzy union (T- co-norm)
5. Fuzzy complement
6. Duality of T-norms and S-norms
7. Fuzzy relations
8. Cartesian products of fuzzy sets
9. Cylindrical extension and projection
10. Fuzzy logical connectives
11. Fuzzy implication
12. Possibilistic fuzzy logic
13. Fuzzy knowledge bases
14. Fuzzy theories
15. Fuzzy inference
16. Fuzzy theories

Knowledge-Based Control Systems

PDF Lecture 1: Introduction & Fuzzy Control I

PDF Lecture 2: Fuzzy Control II

Topics covered:

1. Fuzzy sets and set-theoretic operations
2. Fuzzy relations
3. Fuzzy systems
4. Linguistic model
5. Approximate reasoning
6. Singleton and Takagi–Sugeno fuzzy system
7. Knowledge-based fuzzy modelling
8. Direct fuzzy control
9. Supervisory fuzzy control

Neural Fuzzy Systems Lecture Notes

Topics covered:

1. Operations on fuzzy sets
2. Fuzzy relations
3. Fuzzy implications
4. Theory of approximate reasoning
5. Fuzzy rule-based systems
6. Fuzzy reasoning schemes
7. Fuzzy logic controllers
8. Neuro-fuzzy classifiers

-Authored by
Paresh Pradhan,
Data Scientist at Absolutdata

Thank You

For reading this edition of BrainWave from the Absolutdata Labs Team. This digest focuses on some technical angles of analytics and data science. BrainWave is published about 4 times a year. If you haven't already, please subscribe so you receive future editions.

Subscribe