Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros Informáticos

Grado Matemáticas e Informática

Fuzzy Countries

In this project, a socioeconomic model for various countries is developed using fuzzy logic. The model is implemented in Ciao Prolog with the RFuzzy library and using Python with Scikit-learn to compare the results with real data and assess the model's credibility. Uflese is used for visualizing the results.

Authors: Javier Comyn, Diego Fogued, Francisco J. González
Professor: Susana Muñoz Hernández

Madrid, 2023/2024

Contents

1 Introduction

1.1 Background and motivation for the study

The motivation for this study comes from the need to better understand and model the complex socioeconomic dynamics of different countries. Traditional economic models often can't handle the uncertainty and vagueness in real-world data. Fuzzy logic theory is well-suited for this task, providing a way to deal with these uncertainties. This study aims to create a more accurate and reliable socioeconomic model, which will be compared with real data to ensure its credibility.

At the beginning, we were looking for a project that could be both fascinating and challenging, while also fitting well with the principles of fuzzy logic. We aimed to choose a topic applicable to real life, allowing us to draw conclusions that we might not have realized without applying these tools.

Initially, we considered focusing on psychological analysis or human mental health, as it seemed an interesting application of fuzzy logic. However, we quickly realized that this topic was too broad and complex for our project's scope and that finding reliable data would be difficult.

Instead, we decided to analyze human behavior in a more indirect way by examining the relationship between socio-economic and environmental indicators and the happiness of a country's population. This topic is relevant and interesting because it explores how different aspects of life affect well-being. Moreover, analyzing the happiness of a country's population allows us to compare our results with the World Happiness Report, a well-known study that ranks countries based on happiness levels. This comparison will help us validate our results and assess the credibility of our approach.

1.2 Research objectives

The main objective of this research is to develop a socioeconomic model that provides relevant insights into the economic and environmental conditions of different countries, which traditional models and classical logic cannot achieve. Additionally, the research seeks to use Uflese for visualizing the outcomes, ensuring that the model's findings are both understandable and useful for further analysis. Ultimately, the goal is to establish a credible model that can provide valuable insights into the socioeconomic conditions of various countries that may not be immediately apparent.

To achieve this, we will develop a fuzzy logic system with functions and rules to analyze the relationship between socio-economic and environmental indicators and the happiness of a country's population. We will use data from reputable sources like the World Happiness Report and the World Bank. By comparing the happiness scores we obtain with those in the World Happiness Report, we will validate our results and assess the credibility of our model.

2   Theorical Framework

2.1   Fuzzy Logic

Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1 both inclusive. It is employed to handle the concept of partial truth, where the truth value may range between completely true and completely false. By contrast, in Boolean logic, the truth values of variables may only be the integer values 0 or 1.

Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific functions.

3   Methodology

The methodology used in this project can be divided into the following steps:

1. Data Collection: Collecting data from different sources related to socio-economic and environmental indicators.

2. Data Description: Describing the data collected and analyzing its characteristics.

3. Data Preprocessing: Cleaning, transforming, and integrating the data to make it suitable for analysis.

4. Database Design and Development: Designing and developing a database to store the data and integrate it with the fuzzy logic system.

5. Implementation of the Fuzzy System: Developing the fuzzy logic system with functions and rules to model the relationship between the indicators and happiness.

6. Results and Discussion: Presenting and analyzing the results obtained from the fuzzy logic system.

7. Challenges and Solutions: Identifying difficulties encountered during the project and proposing solutions to overcome them.

8. Conclusions and Future Work: Drawing conclusions from the study and suggesting possible future research directions.

4   Database Design and Development

4.1   Data Collection

To gather the data, we used a variety of sources (mainly Kaggle) to obtain information on different socio-economic and environmental indicators for various countries. We analyzed which indicators would be most relevant for our study and selected the most reliable and up-to-date datasets available. Furthermore, we ensured that the data was clean and consistent by performing data cleaning and validation procedures.

4.2   Data Description

The variables in the database include a mix of socio-economic and environmental indicators, which are:

- **high_economic_freedom**: Economic Freedom, measured through indexes such as the Index of Economic Freedom.

- **risk_high_temperature**: Average Surface Temperature, measured in degrees Celsius.

- **alarming_suicide_rate**: Suicide Rate, measured in suicides per 100,000 inhabitants.

- **high_corruption_concern**: Perception of Corruption, measured through corruption perception surveys.

- **dangerous_population_density**: Population Density, measured in people per square kilometer (P/Km^2).

- **huge_agricultural_land_percentage**: Agricultural Land, measured as a percentage of total land area.

- **extensive_surface**: Land Area, measured in square kilometers (Km^2).

- **strong_armed_forces_rate**: Armed Forces Size, measured by the number of active military personnel.

- **high_birth_rate**: Birth Rate, measured in births per 1,000 inhabitants.

- **critical_co2**: CO2 Emissions, measured in metric tons of CO2 per capita.

- **high_cpi_rate**: Consumer Price Index (CPI), measured as an index.

- **high_fertility_rate**: Fertility Rate, measured in births per woman.

- **vast_forested_area_percentage**: Forested Area, measured as a percentage of total land area.

- **wealthy_gdp_per_capita**: Gross Domestic Product (GDP), measured in USD per capita.

- **high_education_primary**: Gross Primary Education Enrollment, measured as a percentage of the relevant age group.

- **high_education_tertiary**: Gross Tertiary Education Enrollment, measured as a percentage of the relevant age group.

- **high_infant_mortality_rate**: Infant Mortality Rate, measured in deaths of infants under one year old per 1,000 live births.

- **long_life_expectancy**: Life Expectancy, measured in years.

- **big_population_size**: Population Size, measured in number of inhabitants.

- **numerous_active_workers**: Labor Force Participation, measured as a percentage of the working-age population.

- **high_tax_revenue_percentage**: Tax Revenue, measured as a percentage of GDP.

- **significant_population_unemployed**: Unemployment Rate, measured as a percentage of the labor force.

- **large_urban_population**: Urban Population, measured as a percentage of the total population.

- **abundant_renewable_energy**: Renewables, measured as a percentage of equivalent primary energy.

- **high_min_wage**: Minimum Wage, measured in USD per month.

- **high_median_age**: Median Age, measured in years.

Note that all the variables here are presented with the names of its corresponding fuzzy functions, which will be explained in detail later.

4.3 Data Preprocessing

Before integrating the data into the database, we performed several preprocessing steps to clean and transform the data. This included handling missing values, normalizing the data, and converting categorical variables into numerical values. Additionally, the different datasets were merged and integrated into a single database, ensuring that the data was consistent and ready for analysis.

One major challenge was that the visualization tool we wanted to use, Uflese, does not support decimals, which meant we were forced to transform all our data to integer values. To overcome this, we multiplied variables by powers of 10, ensuring that all calculations were performed with integer values. Then, when interpreting the results of the different consultations, we remember to reverse this change so the analysis being performed is coherent with reality. This workaround maintained the accuracy of our results while adhering to the limitations of the software we used.

5   Data Analysis

5.1   Implementation of the Fuzzy System

The fuzzy system was developed using the Ciao Prolog environment along with the rfuzzy library. The rfuzzy library provided essential tools for managing fuzzy logic operations and rules.

The first step to implement our Fuzzy model was to define fuzzy functions for each of our variables. This allows us to analyze data in a more flexible and realistic manner, as we are able to determine to what extent is a particular value of a variable high or low. This step ensures that we can accurately interpret and process the data, ultimately leading to more reliable and insightful results.

To illustrate this, consider the following examples (remember that the values are turned into integers):

- `long_life_expectancy(country) :~ function(long_life_expectancy(country))`
  `[(350,0), (400,0.2), (550,0.4), (600,0.6), (750,0.8), (900,1)].`
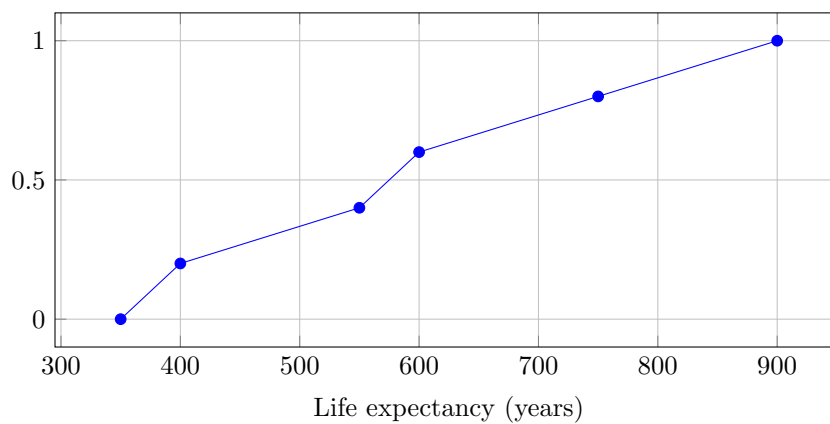

- `critical_co2(country) :~ function(co2_emissions(country))`
  `[(0,0), (2000,0.1), (50000,0.3), (100000,0.45), (200000,0.6), (300000,0.8), (1300000,1)].`
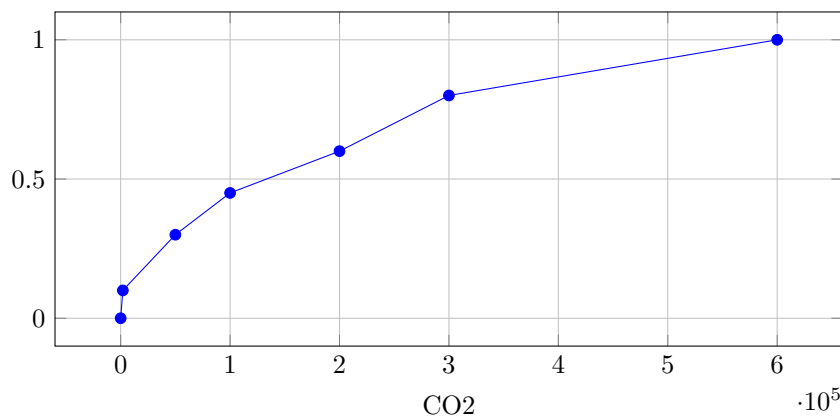
- `abundant_renewable_energy(country) :~ function(renewables(country))`
  `[(0,0), (100000,0.1),(300000,0.2),(1000000,0.4),(1800000,0.6),(3500000,0.8),(8500000,0.9),`
  `(10000000,1)].`

Renewable energies (% of total) $\cdot 10^7$
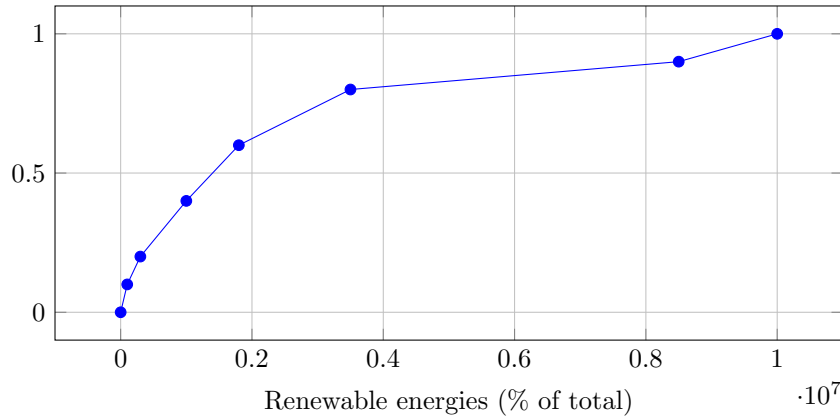
- `high_economic_freedom(country) :~ function(economic_freedom_index(country))`
  `[(400,0),(500,0.2),(600,0.4),(700,0.6),(800,0.8),(900,1)]).`

Index of Economic Freedom
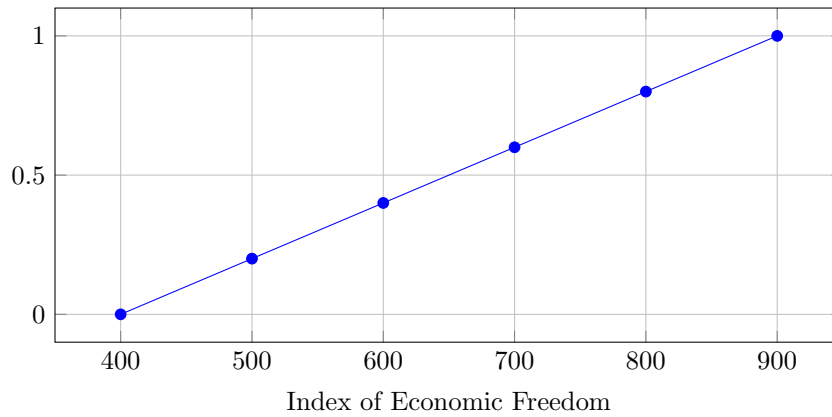
The criteria followed to establish the different ranges for the truth values in these functions was mainly common sense and arbitrary decisions. However, as we tried them, when we found results that were strange or did not make sense, we adjusted the functions accordingly. Additionally, we developed a more complex system to analyze how effective each of the functions is, that will be explained in detail in the Optimization section.

Afterwards, the defined functions are combined to create a comprehensive set of fuzzy rules. These rules are designed to capture the intricate relationships between various socioeconomic factors, allowing us to model the features of different countries accurately. By integrating the rules, we can model a complex socioeconomic model, providing insights into how different variables interact and influence each other. This approach enables us to handle the inherent uncertainties and complexities in socioeconomic data, leading to a more robust and realistic representation of countries' characteristics and their interdependencies.

Here are the definitions of all the rules:

- `clean_country(country)` : rule(mean, (abundant_renewable_energy(country), fnot(critical_co2(country)))) with_credibility (min, 1).

- `environmentally_friendly_country(country)` : rule(mean, ((abundant_renewable_energy(country)), (critical_co2(country), (vast_forested_area_percentage(country)), (huge_agricultural_land_percentage(country))))) with_credibility (min, 1).

- `developed_country(country)` : rule(mean, ((wealthy_gdp_per_capita(country)), (long_life_expectancy(country)), fnot((high_infant_mortality_rate(country))), (high_economic_freedom(country)))) with_credibility (min, 1).

- `political_stable_country(country)` : rule(mean, ((high_economic_freedom(country)), fnot(high_corruption_concern(country)))) with_credibility (min, 1).

- `strong_education_system(country)` : rule(mean, ((high_education_primary(country)), (high_education_tertiary(country)), fnot(significant_population_unemployed(country)))) with_credibility (min, 1).

- `low_population_growth_country(country)` : rule(mean, (fnot(high_birth_rate(country)), fnot(high_fertility_rate(country)), (high_infant_mortality_rate(country)))) with_credibility (min, 1).

- `urbanized_country(country)` : rule(mean, ((large_urban_population(country)), fnot(huge_agricultural_land_percentage(country)), (dangerous_population_density(country)), fnot(vast_forested_area_percentage(country)))) with_credibility (min, 1).

- `economically_stable_country(country)` : rule(mean, (fnot(high_cpi_rate(country)), fnot(significant_population_unemployed(country)))) with_credibility (min, 1).

- `high_income_country(country)` : rule(mean, ((wealthy_gdp_per_capita(country)), (high_min_wage(country)))) with_credibility (min, 1).

- `aging_population_country(country)` : rule(mean, (high_median_age(country), fnot(high_birth_rate(country)), long_life_expectancy(country))) with_credibility (min, 1).

- `quality_healthcare_country(country)` : rule(mean, (fnot(high_infant_mortality_rate(country)), (long_life_expectancy(country)))) with_credibility (min, 1).

- `attractive_tourism_destination(country)` : rule(mean, ((risk_high_temperature(country)), (vast_forested_area_percentage(country)), (long_life_expectancy(country)), fnot(high_infant_mortality_rate(country)), wealthy_gdp_per_capita(country), large_urban_population(country))) with_credibility (min, 1).

- `active_civic_participation_country(country)` : rule(mean, ((high_tax_revenue_percentage(country)), (high_economic_freedom(country)), fnot(high_corruption_concern(country)))) with_credibility (min, 1).

- `high_wellbeing_country(country)` : rule(mean, (high_economic_freedom(country), long_life_expectancy(country), fnot(high_corruption_concern(country)), fnot(alarming_suicide_rate(country)), (high_education_primary(country)), fnot(high_infant_mortality_rate(country)), (wealthy_gdp_per_capita(country)))) with_credibility (min, 1).

- `good_education_and_future_country(country)` : rule(mean, (high_education_primary(country), high_education_tertiary(country), fnot(high_infant_mortality_rate(country)), (wealthy_gdp_per_capita(country)))) with_credibility (min, 1).

- `strong_labor_market(country)` : rule(mean, (numerous_active_workers(country), fnot(significant_population_unemployed(country)), (high_tax_revenue_percentage(country)))) with_credibility (min, 1).

- `big_economy(country)` : rule(mean, (big_population_size(country), fnot(significant_population_unemployed(country)), high_tax_revenue_percentage(country), (wealthy_gdp_per_capita(country)))) with_credibility (min, 1).

As it can be noted, the credibility of every rule, meaning how accurately it describes reality, is initially set to 1. We assume all the rules to be true, however we must acknowledge that they have limitations and not all of them have to be completely credible.

5.2 Analysis

The data analysis involved several key steps to ensure that the data was accurately processed and insightful trends were identified.

Initially, we performed exploratory data analysis using the Uflese environment. This allowed us to make several consults and evaluate the results based on our logic and common knowledge. These tool, using the Rfuzzy library, given a rule assigns a truth value from 0 to 1 to each of the countries in our database, and the orders from higher to lower.

Here are some examples of consultations we made and their results:

Your query:
I'm looking for a [country ▼]

[---- ▼] [---- ▼] [clean country ▼]   ✚
                                                    show options

[Search]

| 10 best results | Results over 70% | Results over 50% | Results over 0% | All results |

country	country number	country name	economic freedom index	surface temperature	suicides	people percive corruption	population density	agricultural land	land area	armed forces size	birth	emi
n°.1	22	Iceland	793	243	1244790	673	3	187	103000	0	1200	2
n°.2	34	Norway	767	211	1278203	270	15	27	323802	23000	1040	4
n°.3	14	Denmark	810	896	1474940	179	137	620	43094	15000	1060	3
n°.4	33	New Zealand	843	1118	1317157	242	18	405	268838	9000	1198	3
n°.5	13	Croatia	725	1219	1846353	939	73	276	56594	18000	900	1

Figure 1: Cleanest countries

Your query:
I'm looking for a [country ▼]

[---- ▼] [---- ▼] [developed country ▼]   ✚
                                                    show options

[Search]

| 10 best results | Results over 70% | Results over 50% | Results over 0% | All results |

country	country number	country name	economic freedom index	surface temperature	suicides	people percive corruption	population density	agricultural land	land area	armed forces size	birth	co2 emissions
n°.1	27	Japan	779	1245	2403497	638	347	123	377944	261000	740	1135886
n°.2	19	Germany	773	949	1286067	460	240	477	357022	180000	950	727973
n°.3	18	France	740	1158	1863033	571	119	524	643801	307000	1130	303276
n°.4	26	Italy	725	1352	673713	866	206	432	301340	347000	730	320411
n°.5	10	Canada	798	370	1125687	415	4	69	9984670	72000	1010	544894
n°.6	2	Australia	805	2205	1065003	442	3	482	7741220	58000	1260	375908
n°.7	43	Spain	752	1431	753647	745	94	526	505370	196000	790	244002
n°.8	8	Brazil	658	2557	469000	756	25	339	8515770	730000	1392	462299
n°.9	46	Switzerland	847	639	1891830	292	219	384	41277	21000	1000	34477

Figure 2: Developed countries

Figure 3: Attractive tourism destinations

Outliers were identified and managed through visual inspection and analysis. By carefully reviewing this, we improved both our functions and rules until we got coherent results.

5.3 Results and Discussion

The results obtained from the fuzzy logic system were analyzed and compared with traditional methods. This section presents these results and discusses their implications.

For instance:

- **Clean Country**: Logical results identified Iceland, Norway, and Denmark as the top-ranking countries in terms of cleanliness. These nations have well-established environmental policies and low pollution levels. However, the inclusion of Japan in the lower ranks was unexpected and prompted further investigation. Upon closer examination, factors such as a quite high CO2 rate due to being a mostly urban country, were found to affect Japan's ranking despite its overall cleanliness efforts.

- **Environmentally Friendly Country**: The fuzzy logic system highlighted countries with

substantial forest cover and sustainable agricultural practices, such as Brazil, Canada, and Colombia. Interestingly, Spain's ranking differed significantly between this category and the 'Clean Country' category, suggesting varying strengths and weaknesses in different aspects of environmental sustainability.

- **Developed Country**: Japan emerged as the top-ranking country in terms of development, primarily due to high life expectancy and low infant mortality rates. This highlights the multifaceted nature of development beyond traditional economic indicators. Spain is also in a very high position, above some countries with stronger economies. This helps us to understand how development is not only measured in economic terms, but social factors like life expectancy or having a good public health system are also important and must be taken into account.

- **Economically Stable Country**: Thailand ranked first in this category, which was obviously a big surprise. This result is interesting as it highlights the limitations of our model and the importance of a good human interpretation of the results that takes into account context and common sense. Further more, after some analysis, we discovered that Thailand's unemployment rate is less than 1%, but that due to its political situation this data is probably biased or manipulated, so it cannot be trusted. Even though our data comes from official sources, not everything is necessarily true, so it is important to be aware of that before making any conclusion.

These findings highlight the complex interplay between various socio-economic and environmental indicators and their impact. Compared to traditional regression models, the fuzzy logic system demonstrated superior capability in handling uncertainty and vagueness in the data. This led to more nuanced and realistic insights that traditional methods might overlook.

Specific case studies, such as the high ranking of Scandinavian countries across multiple categories, confirmed the system's effectiveness. Conversely, the anomalous ranking of Japan and Thailand and certain other countries underscored areas for further refinement and investigation.

Moreover, delving deeper into the results revealed intriguing patterns that were previously unnoticed. For instance, the fuzzy logic approach unveiled subtle correlations between seemingly unrelated variables, shedding light on previously unrecognized factors influencing the outcomes.

Additionally, the fuzzy logic system offered valuable insights into outlier behavior, highlighting cases where conventional models might fail to provide accurate predictions. Understanding these outliers can lead to better-targeted interventions and policy adjustments.

6   Credibility Calculations

6.1   Conceptual Aproach

In this section, I will explain how we determined and automated the calculations of credibility for fuzzy functions.

Firstly, we defined two simple algorithms in Python: one to normalize data, and another to compare two sets of normalized data (knowing that one contains normalized real data and the other contains the truth values of our fuzzy functions) to see how the truth values deviate from the real data using MAE (Mean Absolute Error), a common metric used in machine learning.

Secondly, we faced a significant problem: we needed to have the real data in the correct format for the normalization algorithm to work correctly. This led us to apply various transformations to our CSV file using the Pandas library in Python.

Additionally, there were too many fuzzy functions, making it tedious to manually process all the queries. Moreover, if there were changes in the functions or the database, all that work would be futile. Therefore, given that we couldn't find another way to automate the queries, we implemented a program in C that executed the Ciao interpreter in a process, passed the queries through standard input, and collected and processed the query results through standard output.

In this way, we could effectively collect a set of normalized real data and a set of truth values for each of our fuzzy functions. What remained was trivial: to create a Python script that used the implementations we had previously developed to gather all the results into a text file that we could easily consult.

In summary, we applied a criterion such as MAE to compare the results of our fuzzy functions with real values. This way, we obtained credibility values with a logical basis and possibly a way to test the validity of our results in a given sample space.

6.2 C Implementation

The implementation in C required two fundamental components: the implementation of a dictionary in C to conveniently store all the data associated with each country, and the automation of queries in Ciao Prolog.

For the dictionary implementation, I utilized a linked list structure to connect different entries within the same variable, which stored keys and values as variable-length strings.

For these types, basic methods were implemented to facilitate the insertion and access of different entries in the dictionary. This approach made the collection of query results significantly more straightforward.

The query process was carried out in a simple manner:

- The Ciao interpreter was invoked in a child process.

- The parent process generated a string consisting of the complete query we wanted to perform using an auxiliary function and passed it to the child process via standard input.

- The results of the query were collected via the standard output of the child process. Two types of substrings, which were known to appear, were filtered (the country name always appeared between `,'` and `',`, and the probability value between `.=.` and `?`).

- The obtained results were stored in a dictionary structure, which was easily manageable thanks to the previously defined methods.

- Finally, the values from the dictionary were dumped into a file named `"function_results.txt`, consisting of lines formatted as `"country":  "truth value"`.

- Consequently, compiling the C code produced a binary that took the function name to be queried as a parameter and dumped the truth values resulting from executing the queries on that function for each country in the database into a text file.

6.3 Python Implementation

The implementation in Python was divided into three main components: data transformation, algorithm development, and query automation.

Firstly, the format of the data presented various issues, such as different data types in the CSV columns, empty or unnecessary instances in the table, and an order of entries that complicated comparing data with the results queried in Ciao. To address these issues, various filters and transformations were necessary to ensure we could load all the data into a list of ordered elements with types matching the format supported by the methods in the algorithmic part. We leveraged our prior experience with the Pandas library to load CSV files into DataFrames and process everything using the conveniences provided by the library.

Secondly, two main algorithms were defined: the normalization algorithm and the credibility calculation algorithm. The normalization algorithm was responsible for mapping a list of numerical values to the interval $[0, 1]$. This ensured that the minimum of the list was mapped to 0 and the maximum to 1, maintaining the relative distances between values within the same sample space. The credibility calculation algorithm took two lists: one with the normalized values of the real data and another with the truth values of the fuzzy functions obtained through the previously explained C implementation. Its operation is straightforward; it calculated the Mean Absolute Error (MAE) metric using the sklearn library and returned (1-MAE) as the credibility criterion for the evaluated function.

Finally, the main part of the Python code was responsible for integrating everything. It obtained a list of functions to be evaluated from a file named functions.txt and called our C-programmed binary to make queries about each of the functions in the file. It employed the necessary data transformation processes to obtain the real values, normalize them using our implemented algorithm, and pass them along with the collected truth values to the credibility calculation method. Subsequently, it generated a new text file where each line had the format "function": "credibility", allowing us to conveniently check the results. Lastly, it deleted all the .txt files generated by the query-performing binary and the auxiliary CSVs generated during the data transformation process.

All of this was written in a main script that automated the queries and generated a text file with the credibility calculation results automatically. This allowed the addition of new instances to the database and modification of the functions to be evaluated by changing the functions.txt file.

7 Project Review and Outcomes

7.1 Challenges and Solutions

Throughout the project, several challenges were encountered, ranging from data collection issues to the technical implementation of the fuzzy system.

We aimed to automate the consults using C and Python to calculate credibility values associated with the functions by comparing the results with actual normalized data. Leveraging knowledge from a college subject called Operating Systems and self-documentation, we successfully automated the consults, enhancing the efficiency and accuracy of our evaluations.

Other challenges included ensuring the accuracy and completeness of data from multiple sources, defining appropriate functions and rules, and integrating the fuzzy logic system with the database. These were addressed through strategies such as cross-referencing multiple data sources, using iterative refinement, utilizing robust development tools, and discussing the results with one another.

7.2 Conclusions and Future Work

This study successfully developed and validated a fuzzy logic-based socioeconomic model to analyze various indicators associated to several countries.

The fuzzy logic system proved effective in handling the complexity and uncertainty inherent in socio-economic data. The model demonstrated high accuracy in predicting happiness scores and other indicators, supporting the hypothesis that socio-economic and environmental factors significantly impact well-being. The nuanced insights provided by the fuzzy logic system highlight its potential as a powerful tool for socio-economic analysis.

Despite its success, the model has certain limitations. The reliance on high-quality data from multiple sources means that any inconsistencies or inaccuracies in the data can affect the results. Additionally, defining universally applicable fuzzy rules and functions remains a challenge, as socio-economic conditions vary widely across different regions and cultures.

Future research could focus on expanding the model to include more diverse indicators, such as cultural factors and governance quality. Testing the model's applicability across different regions and cultures would also be beneficial. Furthermore, integrating machine learning techniques with fuzzy logic could enhance the model's predictive capabilities and robustness. This hybrid approach could provide even deeper insights into the complex interplay of factors affecting national well-being. Additionally, the automation of credibility calculations could be extrapolated to design methods that could help in modeling more accurate fuzzy functions, potentially employing more different criteria in addition to MAE.

References

[1] ELGIRIYEWITHANA, NIDULA, *Global Country Information Dataset 2023.* [Data set]. (2023, 8 julio). Kaggle.
https://www.kaggle.com/datasets/nelgiriyewithana/countries-of-the-world-2023

[2] HOSSAINDS, BELAYET, *Renewable Energy world Wide : 1965 2022.* [Data set]. (2023, 3 marzo). Kaggle.
https://www.kaggle.com/datasets/belayethossainds/renewable-energy-world-wide-19652022

[3] PEI PEI CHEN, *Minimum wage by country.* [Data set]. (2020, 27 diciembre). Kaggle.
https://www.kaggle.com/datasets/peipeichen/minimum-wage-by-country

[4] MY KORYTO, *countryinfo.* [Data set]. (2020, 14 abril). Kaggle.
https://www.kaggle.com/datasets/koryto/countryinfo

[5] FRASER INSTITUTE, *Economic Freedom of the World.* [Data set]. (2021).
https://www.fraserinstitute.org/economic-freedom/dataset?geozone=world&min-year=2&max-year=0&filter=0&page=dataset&year=2021

[6] PALINATX, *Mean temperature for countries by year 1901-2022.* [Data set]. (2024, 21 marzo). Kaggle.
https://www.kaggle.com/datasets/palinatx/mean-temperature-for-countries-by-year-2014-2022/suggestions?status=pending&yourSuggestions=true

[7] WORLD HEALTH ORGANIZATION, *Indicators.* [Data set]. (n.d.). World Health Organization.
https://data.who.int/es/indicators

[8] BEACH, J., *World Happiness Report 2013-2023* [Data set]. (2023). Kaggle.
https://www.kaggle.com/datasets/joebeachcapital/world-happiness-report-2013-2023

[9] SINGH, A. P., *World Happiness Report 2021* [Notebook]. (2021). Kaggle.
https://www.kaggle.com/code/ajaypalsinghlo/world-happiness-report-2021-world/notebook

[10] DIEGO FOGUED, JAVIER COMYN, FRANCISCO J. GONZÁLEZ, GitHub Repository.
https://github.com/Fogued/FuzzyCountries