A Comparative Analysis of Soft Set Classifiers and a Fuzzy Classifier as Diagnostic Tools for Diabetes Prediction

Kamil Giziński[1], Bartosz Rolnik[1] and Dominik Sigulski[1]

[1]Faculty of Applied Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland

Abstract

There are lots of different machine learning algorithms, and every one of them has its use. However, not every one of those algorithms will be a good choice for every scenario that we want to test - some are simply superior for one case, while others are more suitable for another. In this short paper, we compare the accuracy for predicting diabetes in Pima Indians patients of two soft set classifiers and one fuzzy classifier.

Keywords

Machine learning, Disease detection, Soft sets, Fuzzy sets

1. Introduction

One of the most important tools of artificial intelligence are neural networks [1]. They are an indispensable tool in identifying certain searched features in the examined objects [2, 3, 4]. No less important role in modern computer science is played by heuristic algorithms, which are often inspired by the observation of the animal world [5, 6]. They are used wherever an optimal solution is sought when the functional describing the optimization goal is difficult to define, we do not know about its mathematical properties. a very interesting application is the reduction of electricity consumption by optimizing the connection of [7, 8, 9] transformers.

Disease detection is becoming a very important field in machine learning [10, 11, 12]. Since we are becoming more and more dependent on the technology we have developed, it is only natural we also implement new technologies [13, 14, 15] in the process of diagnosing certain diseases in patients; or at least marking those who we suspect are sick, so that a healthcare [16, 17] professional can examine their cases more closely. The usage of machine learning algorithms can both improve the patients' outcomes and save valuable time of both doctors and patients.

Since not every machine learning algorithm is a good fit for every problem, it is important to test different algorithms to see how they fit in our scenario. David H. Wolpert and William G. Macready have themselves indicated that the first theorem in their paper "state[s] that any two algorithms are equivalent when their performance is averaged across all possible problems"[18, 7]. That is why we have decided to focus on the comparison between algorithms, two soft set classifiers and a fuzzy classifier to be exact, to see how they perform - both in terms of their accuracy and the time required to produce a result.

Fuzzy sets [19, 20, 21, 22, 23] are sets whose elements have degrees of membership. They are an extension of the classical set definition, where the membership of an element to a set is described in a binary way - an element either belongs to a set or it does not; there is no "in between". The fuzzy set theory allows defining the membership of an element using a membership function valued in the real unit interval [0,1]. One of the biggest difficulties in the usage of fuzzy sets is the uncertainty regarding the membership function: we can never really know whether our choice of the membership function is the optimal one. In our case, the membership functions are, for the most part, based on medical norms.

Soft sets are a generalization of fuzzy sets, and they deal with the uncertainty in a parameterized way. On the contrary to fuzzy sets, they describe the reality more extensively. As described in Khan and Herawan 2021, "Soft set describes fuzzy data in term of each parameter presence or absence while fuzzy set describe it in term of all parameter's accumulative weight only." This approach has its advantages and disadvantages, of course, as it may be harder for a human to make a decision based on many parameters, while it is certainly much easier to make a choice based on a single, crisp value produced by a classifier using a fuzzy set. However, it also means that a decision made based on a soft set reasoning can be much more complete, with more parameters available.

2. Mathematical model

2.1. Data normalization

In select cases, data has been normalized using one of two types of normalization:

SYSYEM 2022: 8th Scholar's Yearly Symposium of Technology, Engineering and Mathematics, Brunek, July 23, 2022

kamigiz359polsl.pl (K. Giziński); bartrol050polsl.pl (B. Rolnik); domisig095polsl.pl (D. Sigulski)

Standard deviation normalization

$$\frac{x - \bar{x}}{\sigma} \tag{1}$$

where:

- x - sample value
- σ - column standard deviation
- $\bar{x}$ - mean column value

Min-max normalization

$$\frac{x - min}{max - min} \tag{2}$$

where:

- x - sample value
- min - minimum column value
- max - maximum column value

2.2. Fuzzy classifier

We have also used a handful of equations in our fuzzy classifier.

Triangular membership function

$$\mu_{triangular}(x; a, b, c) = \begin{cases} 0, & \text{if } x \leq a. \\ \frac{x-a}{b-a}, & \text{if } a < x \leq b. \\ \frac{c-x}{c-b}, & \text{if } b < x \leq c. \\ 0, & \text{if } c < x. \end{cases} \tag{3}$$

where:

- a - the beginning of the base of the triangle, membership takes the value 0
- b - the center of the triangle, membership takes the value 1
- c - the end of the base of the triangle, membership takes the value 0

Trapezoidal membership function

$$\mu_{trapezoidal}(x; a, b, c, d) = \begin{cases} 0, & \text{if } x \leq a. \\ \frac{x-a}{b-a}, & \text{if } a < x \leq b. \\ 1, & \text{if } b < x \leq c. \\ \frac{d-x}{d-c}, & \text{if } c < x \leq d. \\ 0, & \text{if } d < x. \end{cases} \tag{4}$$

where:

- a - the beginning of the lower base of the trapezoid, membership takes the value 0
- b - the beginning of the upper base of the trapezoid, membership takes the value 1
- c - the end of the upper base of the trapezoid, membership takes the value 1
- d - the end of the lower base of the trapezoid, membership takes the value 0

Antecedent fulfillment degree

$$\mu_i = max\{afd\} \tag{5}$$

where:

- afd - antecedent fulfillment degrees (i.e.: afd=(0.6, 0.75))

Rule fulfillment degree for all antecedents

$$\mu_{all}(x) = min\{\mu_1, \mu_2, ..., \mu_i\} \tag{6}$$

where:

- μ_i - antecedent fulfillment degree

Center of gravity of a triangle

$$S = \frac{a + b + c}{3} \tag{7}$$

where:

- s - the x-axis coordinate for the center of gravity
- a, b, c - x-axis coordinates of the triangle vertices

Defuzzification

$$h = \frac{\sum_i \mu_i A_i c_i}{\sum_i \mu_i A_i} \tag{8}$$

where:

- h - crisp, defuzzified value
- A_i - the area of the ith set
- μ_i - membership degree of the ith set
- c_i - center of gravity of the ith set

3. Dataset

The dataset used for this project was the Pima Indians Diabetes Database [24]. According to the source, "This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. [...] In particular, all patients here are females at least 21 years old of Pima Indian heritage." The dataset consists of several medical predictor variables and one target variable. Here is a short description of each column.

1. **Pregnancies** — number of times pregnant;
2. **Glucose** — plasma glucose concentration after 2 hours in an oral glucose tolerance test;
3. **BloodPressure** — diastolic blood pressure (mm Hg);
4. **SkinThickness** — triceps skin fold thickness (mm);

5. **Insulin** — 2-Hour serum insulin (μ IU/ml);
6. **BMI** — Body mass index (weight in kg/(height in m)^2);
7. **DiabetesPedigreeFunction** — a function that describes the likelihood of diabetes based on family history;
8. **Age** — age in years;
9. **Outcome** — class variable (0 or 1). 268 of 768 are 1 and 500 are 0;

3.1. Statistical analysis

Table 1
Initial dataset summary

Column	Non-null	Data type
Pregnancies	768	Integer
Glucose	768	Integer
BloodPressure	768	Integer
SkinThickness	768	Integer
Insulin	768	Integer
BMI	768	Float
DiabetesPedigreeFunction	768	Float
Age	768	Integer
Outcome	768	Integer

Table 2
Initial columns' statistics

	Pregnancies	Glucose	BloodPressure
count	768	768	768
mean	3.84	120.89	69.10
std	3.36	31.97	19.35
min	0	0	0
25%	1	99	62
50%	3	117	72
75%	6	140.25	80
max	17	199	122

Table 3
Initial columns' statistics - continuation

	SkinThickness	Insulin	BMI
count	768	768	768
mean	20.53	79.79	31.99
std	15.95	115.24	7.88
min	0	0	0
25%	0	0	27.3
50%	23	30.5	32
75%	32	127.25	36.6
max	99	846	67.1

Table 4
Initial columns' statistics - continuation

	DiabetesPedigreeFunction	Age	Outcome
count	768	768	768
mean	0.47	33.24	0.34
std	0.33	11.76	0.47
min	0.078	21	0
25%	0.24	24	0
50%	0.37	29	0
75%	0.62	41	1
max	2.42	81	1

However, we had to clean the data first - as seen in Tables 2, 3 and 4, there were many incomplete records which in turn would result in the classifiers producing incorrect predictions. Which is why we removed all zero values outside the Pregnancies and Outcome columns. The column statistics after removing the incomplete records can be seen in Tables 5, 6 and 7.

Table 5
Columns' statistics (without zero values)

	Pregnancies	Glucose	BloodPressure
count	392	392	392
mean	3.30	122.63	70.66
std	3.21	30.86	12.50
min	0	56	24
25%	1	99	62
50%	2	119	70
75%	5	143	78
max	17	198	110

Table 6
Columns' statistics (without zero values) - continuation

	SkinThickness	Insulin	BMI
count	392	392	392
mean	29.15	156.06	33.09
std	10.52	118.84	7.03
min	7	14	18.20
25%	21	76.75	28.40
50%	29	125.50	33.20
75%	37	190	37.10
max	63	846	67.10

We have also shuffled the dataset, normalized it in select cases (either min-max or standard deviation normalization) and divided it into a training set and a validation set (70:30 split).

The values for both sick (marked with yellow) and healthy patients (marked with red) are not too distinct, apart from the glucose column - as seen in Figure 1. As

Table 7
Columns' statistics - continuation

	DiabetesPedigreeFunction	Age	Outcome
count	392	392	392
mean	0.52	30.86	0.33
std	0.35	10.20	0.47
min	0.085	21	0
25%	0.27	23	0
50%	0.45	27	0
75%	0.69	36	1
max	2.42	81	1

shown in Figure 2, the correlation between the outcome values and other columns is not particularly strong either - only the glucose column shows a correlation above 0.5, with almost all other correlations not even exceeding 0.3.

3.2. Medical terms and norms

In this subsection, we are going to describe the columns in the database from the medical point of view.

Glucose is a simple sugar, which is the basic source of energy in the human body. Measuring blood glucose levels on an empty stomach can help determine if a person has diabetes.

The approximate norm of blood glucose concentration is:

- **70-99 mg/dL** — correct values;
- **100-125 mg/dL** — abnormal values, oral glucose tolerance test is required;
- **above 126 mg/dL** — abnormal values, repeating the test is required, patient is diagnosed with diabetes after getting such a result twice.

The next column is **blood pressure**. Blood pressure is the amount of pressure that flowing blood exerts against the walls of your arteries. The measurement is made with a pressure gauge and the obtained values are given in mm Hg, i.e., millimeters of mercury.

Blood pressure in terms of the norms is divided into:

- **normal**: <80 mm Hg;
- **elevated phase 1**: 80–89 mm Hg;
- **elevated phase 2**: >90 mm Hg.

The thickness of the skin fold on the triceps is one of the determinants of body fat level. We have not managed to find any information regarding the accepted norms.

Insulin is a hormone responsible for regulating blood sugar (glucose) in the body. The norm 2 hours after a meal is **up to 30 mIU/ml**.However, in our dataset the median for insulin was 125.5 μ IU/ml - 4 times over the norm, if we assume the units are the same exact ones - which suggests that there was an error in the dataset - or perhaps a different testing method was used.

Diabetes pedigree function
Family history of diabetes was shown to be a significant predictor of diabetes prevalence.There is no additional information about this function in the dataset, apart from the short explanation that it is "a function that describes the likelihood of diabetes based on family history", so we can only assume that the higher the value, the higher the risk is for a patient.

BMI is the body mass index. It is calculated by comparing height with weight. Its value is helpful in assessing the risk of overweight-related diseases such as atherosclerosis, diabetes or ischemic heart disease. The lower the BMI value, the lower the risk of disease development.

Norms for BMI:

- **underweight**: <18.5;
- **health weight**: 18.5 - 24.9;
- **overweight**: 25.0 - 29.9;
- **obese**: >30.

Age
According to the American CDC agency, patients are at risk if they are **45 years or older**.

4. Implementation

All the algorithms were written and executed in Python 3.9.5 using Jupyter Notebooks.

4.1. Soft set classifier - mean

The first soft set classifier uses a mean column value to create the weights table.

Firstly it gets the mean value, then it counts how many values in the column were below and above the mean. If there were more values above the mean, the program adds a pair $[0, 1]$ to the table, otherwise it adds a pair $[1, 0]$. If there is a draw, we randomly pick one of the pairs and append it to the weights table.

4.2. Soft set classifier - percentage

The second soft set classifier uses minimum and maximum column values to calculate how a sample value compares to the minimum and maximum values, to then create the weights table.

Firstly it gets the minimum and maximum values, then it calculates how far they are from minimum to maximum in percentage - i.e. if a sample value is equal to minimum, the algorithm appends $[0, 1]$ to the weights table and if a value is right in the middle, it appends $[0.5, 0.5]$.

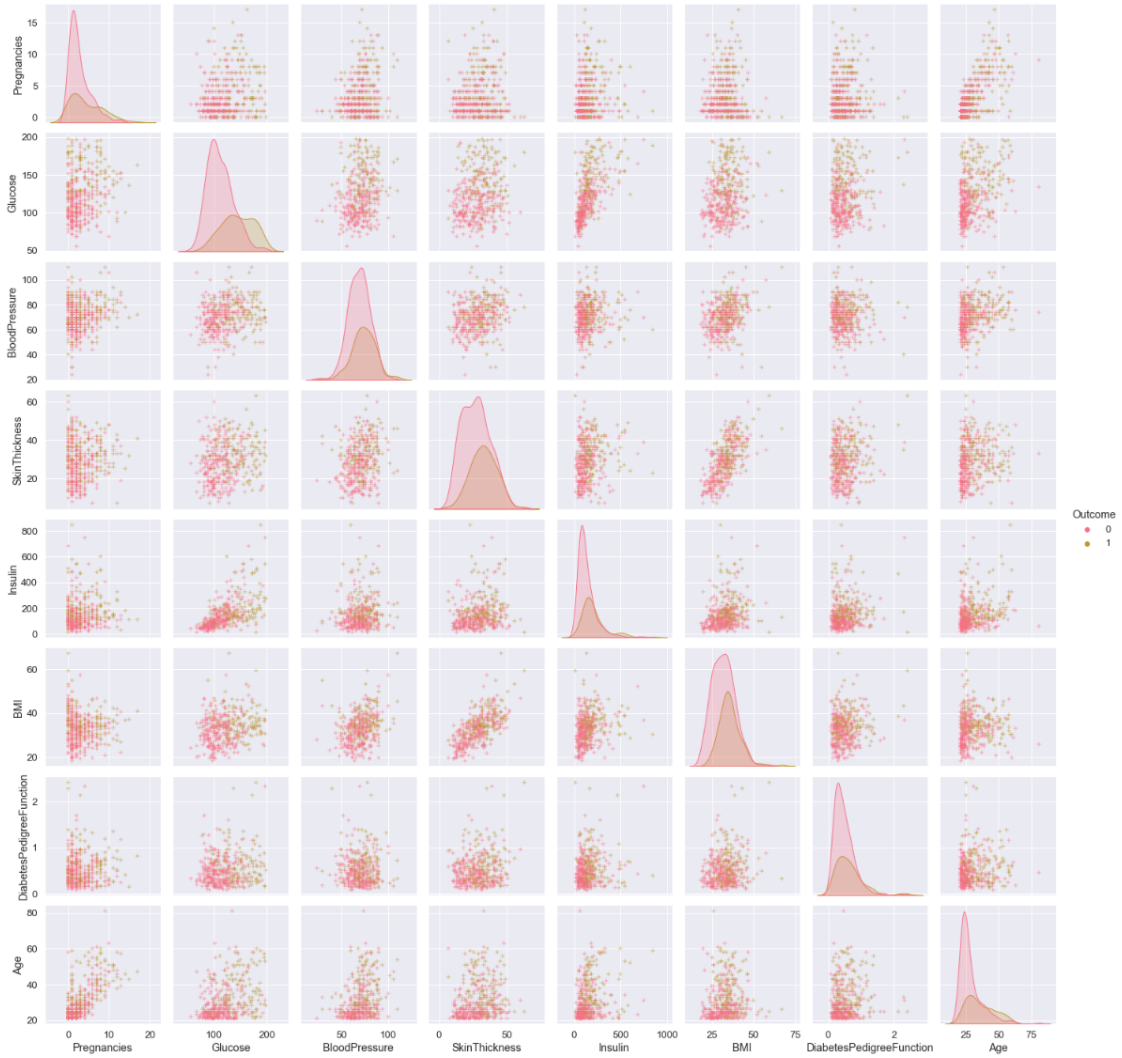

Figure 1: Distribution chart of results for pairs of columns

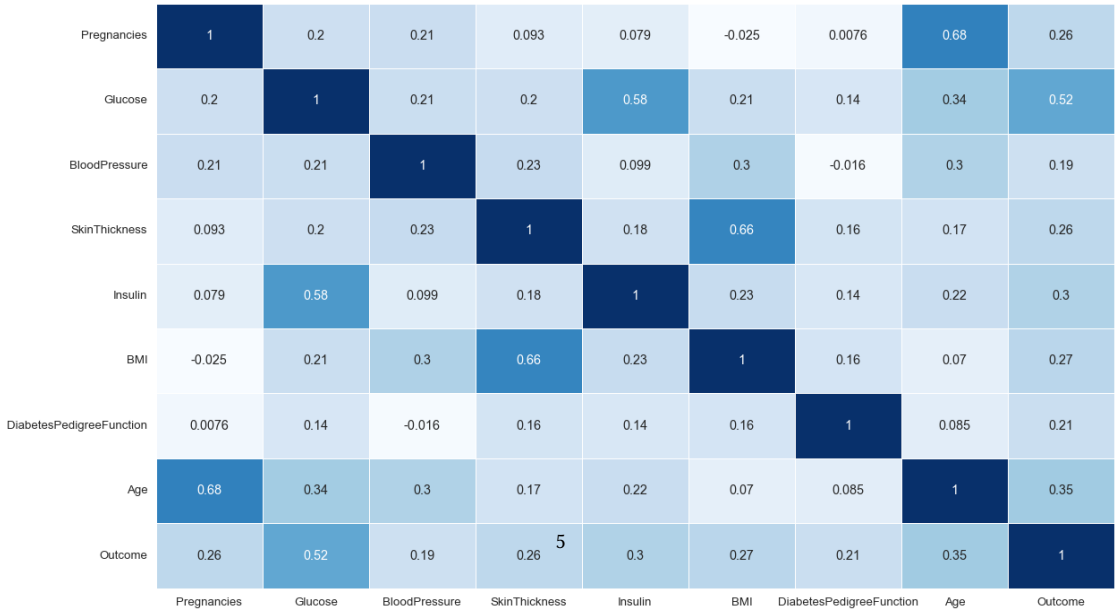

Figure 2: Columns' correlation chart

4.3. Fuzzy classifier

Firstly, we need to explain how we chose the membership functions for the antecedents and the consequent.

There is no clear-cut threshold for the number of pregnancies, so we have decided to pick 2 as our middle threshold and with the rest of the domain divided based on the middle triangle, as seen in Figure 3.

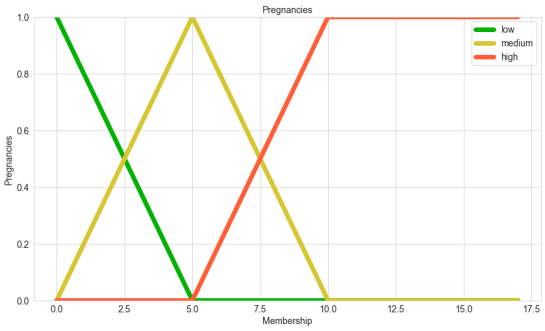

Figure 3: Membership function of the Pregnancies column

We have decided to divide the glucose column domain according to the glucose norms. with the middle being at 112, since this is the middle of the abnormal range of values that do not yet warrant a diabetes of diagnosis. The chart of its membership function can be seen in Figure 4.

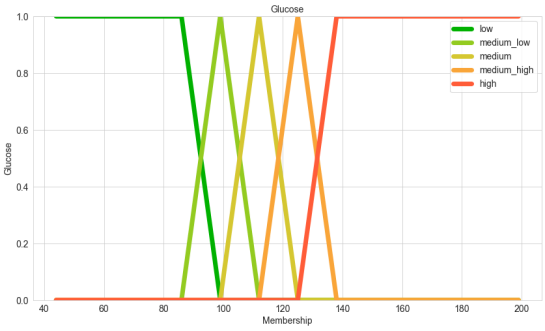

Figure 4: Membership function of the Glucose column

The column containing blood pressure values is divided in accordance with the blood pressure norms mentioned in the "Medical terms and norms" section, as seen in Figure 5.

Since we could not find any norms regarding skin thickness on triceps, we have divided its membership function based on the median value, as seen in Figure 6.

The values inside the insulin column did not seem to match any medical norms we have managed to find,

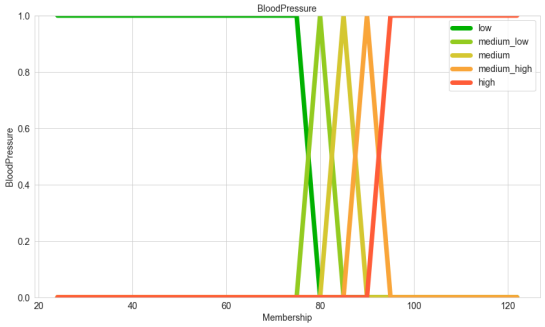

Figure 5: Membership function of the BloodPressure column

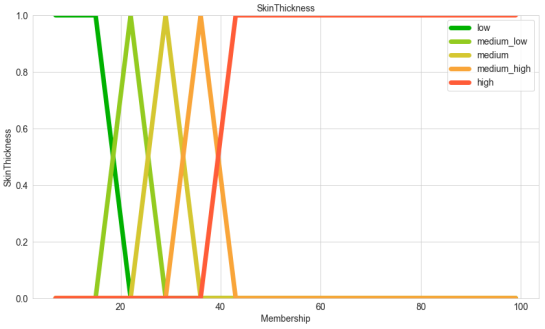

Figure 6: Membership function of the SkinThickness column

so we had to divide them using the quartiles as rough division points, as seen in Figure 7.

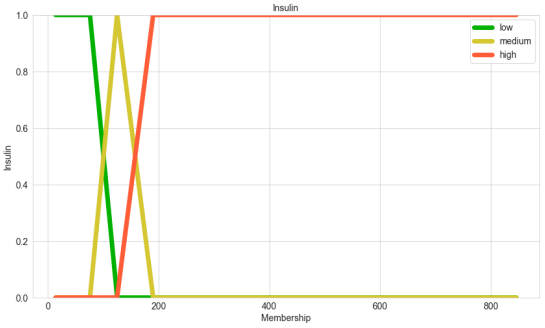

Figure 7: Membership function of the Insulin column

BMI has quite clear norms, so we used them to divide the values in the BMI column - as seen in Figure 8.

As seen in Figure 9, the diabetes pedigree function values were divided using its median, since no medical norms were available.

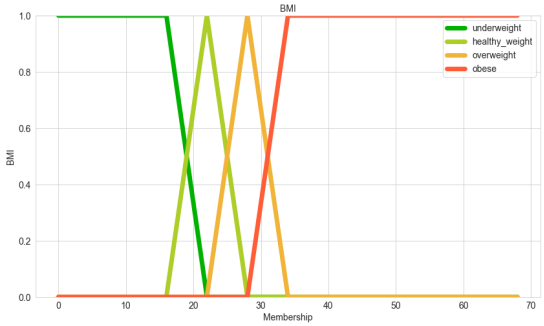

Figure 8: Membership function of the BMI column

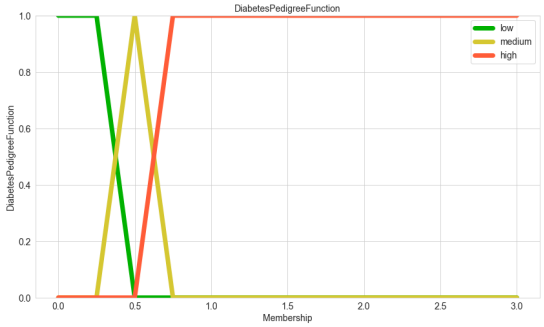

Figure 9: Membership function of the DiabetesPedigree-Function column

As seen in Figure 10, age values were divided according to CDC's norm that people who are 45 years or older are at risk of developing type 2 of diabetes.

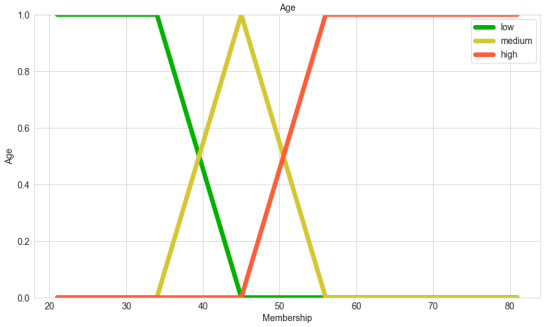

Figure 10: Membership function of the Age column

Finally, the outcome consequent was divided in half, with values that fall in the middle being marked as 'sick', which can be seen in Figure 11.

After creating the membership functions, we wrote the rules connecting all the antecedents to the consequent. An example of such a rule can be seen in Table 8

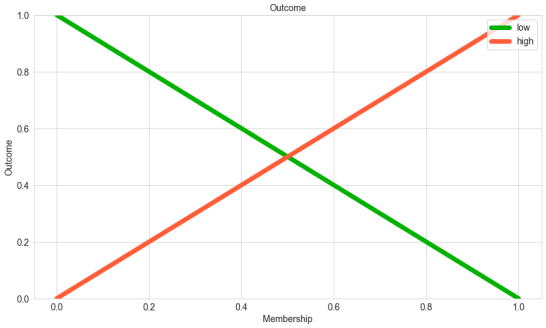

Figure 11: Membership function of the Outcome column

Table 8
An example of a fuzzy rule

Column	Label
Pregnancies	low or medium
Glucose	medium or medium-high or high
Blood Pressure	medium or medium-high or high
Skin Thickness	low or medium-low or medium
Insulin	low or medium
BMI	underweight or healthy weight
Diabetes Pedigree Function	low or medium
Age	low or medium
Outcome	low

The algorithm starts by getting the membership degrees for each column and then fuzzifying the sample values based on these degrees. Once we have all the fuzzy values, we can calculate the rule fulfillment and then pick the highest fulfillment degrees for each consequent label. Then out of these labels we pick the one with the highest degree - if there is a "draw", i.e., the degrees are equal, we pick the worst case - so in our dataset we mark the patient as sick, since it is obviously better to mark one too many, than one too few.

5. Performance

As seen in Figure 12, the time performance of the classifiers varied quite a lot, with the soft set classifier using percentages being more than three times slower than the soft set classifier using mean values and almost twice as slow as the fuzzy classifier. The time performance is an average of 50 timed runs to average out any abnormal situations.

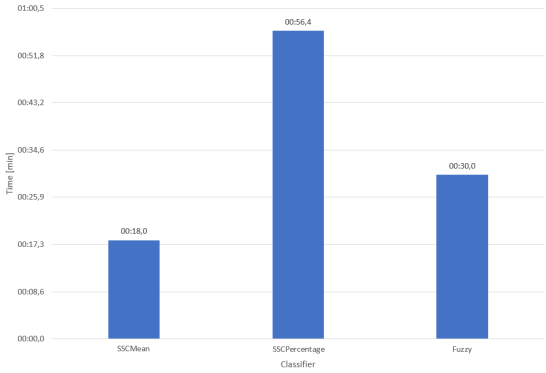

Figure 12: Performance summary of the classifiers

6. Results

The soft set classifier using mean values was the worst performing one, with less than 50% accuracy for data normalized with standard deviation and around 50% accuracy for data normalized with percentages and unnormalized data.

The soft set classifier using percentages did not perform much better, either, with its accuracy slightly exceeding 50%. As seen in Table 9, the results were quite in all cases. On the contrary to the mean soft set classifier, it performed best with data normalized using standard deviation - but the difference was not stark at all.

Table 9
SoftSetClassifierMean and SoftSetClassifierPercentage average results.

	SSCMean	SSCPercentage
std normalized	48.44%	55.12%
minmax normalized	50.08%	53.32%
unnormalized	50.12%	52.93%

The fuzzy classifier came out on top, with its accuracy averaging out at more than 60%. As seen in Table 11 and Figure 13, its accuracy did not drop below 55% - so in almost all cases it performed better than other classifiers did on average. Even taking the standard deviation into account, ass seen in Table 11, the results were significantly better.

7. Conclusion

In conclusion, the soft set classifiers were not a good choice for this problem and dataset. They probably would have worked better with a more uniform outcome distribution - but with more samples being marked as 'sick', the results were skewed.

The fuzzy classifier produced superior results, and with improved, more fine-tuned rules it could achieve an even better accuracy. Similarly to the other soft set classifier, it would have benefitted from a more uniform data distribution.

As seen in Figure 14, the fuzzy classifier did evidently better, but the results can definitely be improved further.

Table 10
Fuzzy Classifier accuracy summary (20 samples)

	Accuracy
1	68.64%
2	62.71%
3	59.32%
4	55.08%
5	58.47%
6	68.64%
7	61.86%
8	61.86%
9	62.71%
10	66.95%
11	67.80%
12	63.56%
13	57.63%
14	61.86%
15	68.64%
16	61.86%
17	60.17%
18	63.56%
19	68.64%
20	68.64%

Table 11
Fuzzy Classifier accuracy summary (20 samples) - mean and standard devitation

	Value
mean	63.43%
std dev	4.19

References

[1] V. S. Dhaka, S. V. Meena, G. Rani, D. Sinwar, M. F. Ijaz, M. Woźniak, A survey of deep convolutional neural networks applied for prediction of plant leaf diseases, Sensors 21 (2021) 4749.

[2] O. Dehzangi, M. Taherisadr, R. ChangalVala, Imu-based gait recognition using convolutional neural networks and multi-sensor fusion, Sensors 17 (2017) 2735.

[3] H. G. Hong, M. B. Lee, K. R. Park, Convolutional neural network-based finger-vein recognition using nir image sensors, Sensors 17 (2017) 1297.

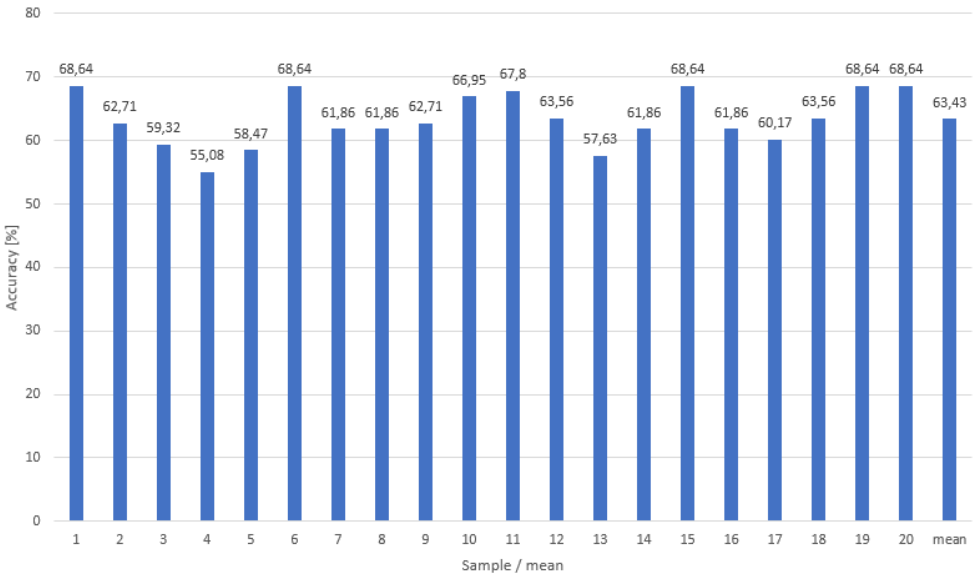

Figure 13: Fuzzy Classifier accuracy summary chart (20 samples)

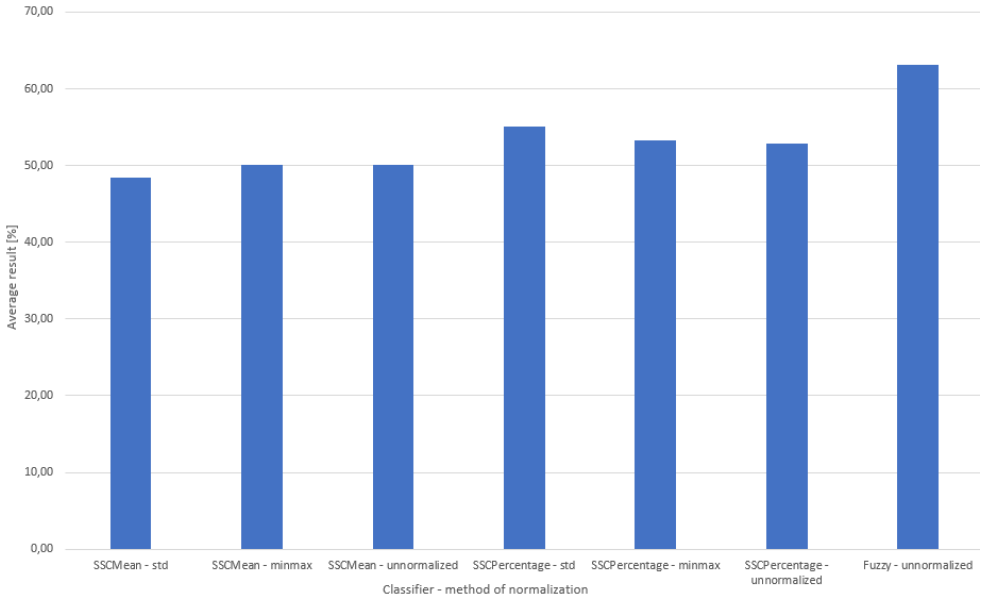

Figure 14: Summary of the accuracy of the classifiers

[4] G. Capizzi, G. Lo Sciuto, C. Napoli, E. Tramontana, M. Woźniak, A novel neural networks-based texture image processing algorithm for orange defects classification, International Journal of Computer Science and Applications 13 (2016) 45 – 60.

[5] Y. Zhang, S. Cheng, Y. Shi, D.-w. Gong, X. Zhao, Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm, Expert Systems with Applications 137 (2019) 46–58.

[6] M. Ren, Y. Song, W. Chu, An improved locally weighted pls based on particle swarm optimization for industrial soft sensor modeling, Sensors 19 (2019) 4099.

[7] F. Bonanno, G. Capizzi, G. Sciuto, C. Napoli, G. Pappalardo, E. Tramontana, A novel cloud-distributed toolbox for optimal energy dispatch management

from renewables in igss by using wrnn predictors and gpu parallel solutions, in: 2014 International Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM 2014, IEEE Computer Society, 2014, pp. 1077–1084. doi:10.1109/SPEEDAM.2014.6872127.

[8] G. Capizzi, G. Lo Sciuto, C. Napoli, E. Tramontana, An advanced neural network based solution to enforce dispatch continuity in smart grids, Applied Soft Computing Journal 62 (2018) 768 – 775.

[9] M. Woźniak, A. Sikora, A. Zielonka, K. Kaur, M. S. Hossain, M. Shorfuzzaman, Heuristic optimization of multipulse rectifier for reduced energy consumption, IEEE Transactions on Industrial Informatics 18 (2021) 5515–5526.

[10] A. T. Özdemir, B. Barshan, Detecting falls with wearable sensors using machine learning techniques, Sensors 14 (2014) 10691–10708.

[11] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, D. Bochtis, Machine learning in agriculture: A review, Sensors 18 (2018) 2674.

[12] G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Matta, M. Re, F. Silvestri, S. Spanò, Efficient ensemble machine learning implementation on fpga using partial reconfiguration, in: Applications in Electronics Pervading Industry, Environment and Society: APPLEPIES 2018 6, Springer, 2019, pp. 253–259.

[13] M. A. Sanchez, O. Castillo, J. R. Castro, Generalized type-2 fuzzy systems for controlling a mobile robot and a performance comparison with interval type-2 and type-1 fuzzy systems, Expert Systems with Applications 42 (2015) 5904–5914.

[14] N. Dat, V. Ponzi, S. Russo, F. Vincelli, Supporting impaired people with a following robotic assistant by means of end-to-end visual target navigation and reinforcement learning approaches, in: CEUR Workshop Proceedings, volume 3118, CEUR-WS, 2021, pp. 51–63.

[15] Q.-b. Zhang, P. Wang, Z.-h. Chen, An improved particle filter for mobile robot localization based on particle swarm optimization, Expert Systems with Applications 135 (2019) 181–193.

[16] M. Woźniak, M. Wieczorek, J. Siłka, D. Połap, Body pose prediction based on motion sensor data and recurrent neural network, IEEE Transactions on Industrial Informatics 17 (2020) 2101–2111.

[17] S. Russo, C. Napoli, A comprehensive solution for psychological treatment and therapeutic path planning based on knowledge base and expertise sharing, in: CEUR Workshop Proceedings, volume 2472, CEUR-WS, 2019, pp. 41–47.

[18] D. H. Wolpert, W. G. Macready, Coevolutionary free lunches, IEEE Transactions on Evolutionary Computation 9 (2005) 721–735. doi:10.1109/TEVC.2005.856205.

[19] M. Woźniak, A. Zielonka, A. Sikora, M. J. Piran, A. Alamri, 6g-enabled iot home environment control using fuzzy rules, IEEE Internet of Things Journal 8 (2020) 5442–5452.

[20] D. Połap, M. Woźniak, C. Napoli, E. Tramontana, Real-time cloud-based game management system via cuckoo search algorithm, International Journal of Electronics and Telecommunications 61 (2015) 333–338. doi:10.1515/eletel-2015-0043.

[21] M. Woźniak, A. Zielonka, A. Sikora, Driving support by type-2 fuzzy logic control model, Expert Systems with Applications 207 (2022) 117798.

[22] A. Carpenzano, R. Caponetto, L. Lo Bello, O. Mirabella, Fuzzy traffic smoothing: An approach for real-time communication over ethernet networks, in: 4th IEEE International Workshop on Factory Communication Systems, IEEE, 2002, pp. 241–248.

[23] N. Brandizzi, S. Russo, R. Brociek, A. Wajda, First studies to apply the theory of mind theory to green and smart mobility by using gaussian area clustering, in: CEUR Workshop Proceedings, volume 3118, CEUR-WS, 2021, pp. 71–76.

[24] U. M. Learning, Pima indians diabetes database, 2016. URL: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database.