International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

Vol. 34 No. 2 September 2022, pp. 205-234

The Decision Tree Aided Neuro-Fuzzy Inference Characterization of the Stochastic Hydrology of the Tana Alluvial Aquifer

Dr. Meshack Owira Amimo[1] and Dr K.S.S. Rakesh[2]

[1]Research Scholar, Livingstone International University of Tourism & Business Management (LIUTEBM), Zambia,
bmoamimo@gmail.com

[2]CEO, Gradxs,
India
kssrakesh@gmail.com

Abstract—The Tana Alluvial Aquifer is the name given to the little-understood aquifer which is active in the areas bordering the River Tana Flow course as the river weaves its way through the sedimentary plains of Balambala, Garissa, Fafi and Ijara and, finally, into the Tana Delta areas, with the common denominator being the proximity to the Lower Tana catchment, especially the riparian corridor of the River itself, and beyond. The aquifer may extend to between five to fifteen kilometers away from the river channels course way, and at times, it may be felt even 20 kilometers away. The geology of the locality is heterogeneous and comprise sediments whose soil mechanics may not be easily deciphered, since some areas close to the river have very fresh water while others are saline (Bura East in Fafi Sub County easily comes to mind here). There are areas far from the river but bearing fresh water (Mulanjo comes to mind). In some areas, sites close to the river discharge low yield figures, whereas those located farther afield discharge favorably. The water quality and discharge are therefore stochastic variables, subject to chance occurrence. In view of this inconsistency, and on the account of data scarcity, the neuro-fuzzy inference algorithm was developed to map the Universe of Discourse of the Tana Alluvial Aquifer, aka the T.A.A., as it relates to the longitudes, latitudes, depths, and discharges of the aquifers in the study area. The mapping was with respect to aquifer discharge, the variable used to characterize an aquifer, in terms of Transmissivity and Hydraulic Conductivity, thereby defining aquifer recharge propensity. Membership functions were developed using the trapezoidal membership family, and fuzzy rules were appropriately evolved from the fuzzified aquifer data, before finally employing the Sugeno inference engines (in Python) to make predictions of discharge, at each of the T.A.A. aquifer subsets mapped for fresh, saline, hard and blackish water species. The accuracy in the outputs achieved in the areas mapped vindicated the power of the neuro-fuzzy inference systems, as the accuracy oscillated between 92 and 99 percent, when the discharge values predicted were compared with the actual known discharge values of the wells mapped. The water quality class characterization was then undertaken using the decision tree (DT) algorithm in python which gave rise to a 100 percent prediction accuracy. The same DT algorithm could not successfully predict the discrete values of aquifer discharge or EC values, with as much accuracy (but performed excellently with salinity class data), and that was why fuzzy logic was employed. The study vindicated the use of the DT and Fuzzy Logic Algorithms as simple, yet powerful analytical tools, in characterizing the Stochastic Hydrology of the Tana Alluvial Aquifer.

Key Words: Stochastic Hydrology, Fuzzy Logic, Decision Tree, Tana Alluvial Aquifer, Gini Index, Information Gain, Membership Function, Universe of Discourse, Defuzzification

Corresponding Author: Dr. Meshack Owira Amimo

I. Background And Project Location

A. Location

The Tana Alluvial aquifer is a phrase coined to describe the sedimentary aquifer suite dominating the riparian corridor of the Lower Tana catchment systems in Tana River and Garissa County Land portions enjoying relative proximity to the River Tana Flow Course.

Very limited technical literature is available to help with aquifer characterization of the Tana Alluvial Aquifer.

The project area lies in BOTH Tana and Garissa Counties located in Northern Kenya within **the localities specified using longitudes and latitudes stated.** The zones mapped are located on the both sides of the River as this same river acts as administrative boundary between Garissa and Tana River counties.

B. Geology and Stratigraphy

The topography is generally flat, and is clayey rich, supporting vegetations that comprise mainly thorny shrubs, undergrowth savannah grass, equatorial weeds typical of the coastal strips and acacia family trees. The vegetation is mainly of the desert xerophyte species and thrives even in extremes of drought when the rains at times fail continuously for a set of three to four consecutive seasons. In fact, some of these vegetation has been successfully used to help map areas that have promising aquifer potential and whose water is fresh from the water quality perspectives. A case in point is the several Doum palm species that colonize the Raya-Shabaha-Sankuri areas located within the Riparian corridor of the Tana flow course and whose aquifers are not only shallow but also happen to have very fresh water quality, ideal for human domestic usage.

The area is apparently a buried Quaternary Lake, with evidence of fresh water systems fish populations. Quaternary Tectonics probably resulted into the landmass uplift with the results that some of the waters were emptied the surrounding Coastal plains. There is more evidence to suggest a coastal climate than a desert climate for the Project Area.

The geology is defined by **dark to light toned sandy clayey sediments, the Mansa Guda formation,** which overlies the carbonates – namely corallites, aragonitic sediments and calcite. The sandy clayey species are mainly the Mariakani Sandstones.

Sedimentary beds dominate both surface and subsurface geology in the hydrostratigraphic information availed during the study. The Jurassic limestone carbonates are fairly fractured and have been known to harbor appreciable quantities of water from past aquifer mapping projects, exhibiting shallow aquifer storage, albeit with anomalous levels of mineralization via the numerous cracks and fissures as well as the pore spaces available to aid transmissivity from one point to the other in a laminar flow environment which is the major recharge flow pathway at the shallow aquifers levels.

Groundwater has also been known to form at the contact points between the lithological units defined by carbonates and the sandstones at appreciable depths. The discontinuity contact points have been known to be the major storage structures of water in the shallow and deep-seated zones in terms of the TAA's hydrogeology. The storage of aquifer water in the upper geologic material has been known to benefit from the annual rainfall recharge through the recharge sinkholes and fractures that characterize the study area. It is therefore safe to opine that upper shallow aquifers will be replenished via direct infiltration during the overland runoffs flow, whenever it rains, whereas the deeper zones of the study area in the depth brackets ranging between 80m to 250m will be recharged through regional flow. These are slow recharge dynamics coming in from far-flung areas like isiolo and Kitui areas from where the Tana partly originates as it traverses the watersheds. These shall be enhanced and augmented by the karstification channels and plate tectonic structures that did form in the Jurassic – cretaceous period. Evapo transpiration rates of up to 3,000mm per annum over shadow the annual rains of up to 600mm per annum.

C. Physiography

The area stands at an average altitude of **77-265m** above sea level within a gently dipping terrain punctuated with several ant hills and flood plains both on the south eastern and north western flanks. The river flows in the northwest-southeastern azimuth.

II. Hydrology, Hydrochemistry And Structural Geology

A. *Recharge Mechanisms within the Lower Tana Aquifer Systems*

Evidences abound of jointing and fracturing of the carbonate sediments on the surface, alluding to intense forces of fracturing, carbonation and quaternary tectonic faulting. Much of the south westerly – north easterly directed stress fields helped sculpture the terrain into its present geological state.

Owing to the relatively high fractions of clays in the beds, there is no sufficient time available for maximum catchment input infiltrations into the sub surface zones lying on the adjacent aquifer units in the proposed well sites. This explains the anomalous salinity levels of the boreholes done to great depths in the area.

B. *Drainage*

Owing to the relative flat nature of the terrain, there is flood rampancy. The permanent civil structures on the ground to stand the risk of destruction added to the occasional loss of lives for both livestock and human persons. Most of the housing units are constructed through shrubs and dry acacia trees locally available, lightening the task of evacuation in the event of impending flood disasters.

C. *Climate*

The project area falls within zone 7 of the classification of climatic/ecological zones of Africa, that is to say arid to semi-arid with temperatures averaging 30 to 34 degrees per day and occasioning evapo transpiration rates of up to 3000mm per annum. The rainfall average falls well below 800mm per year.

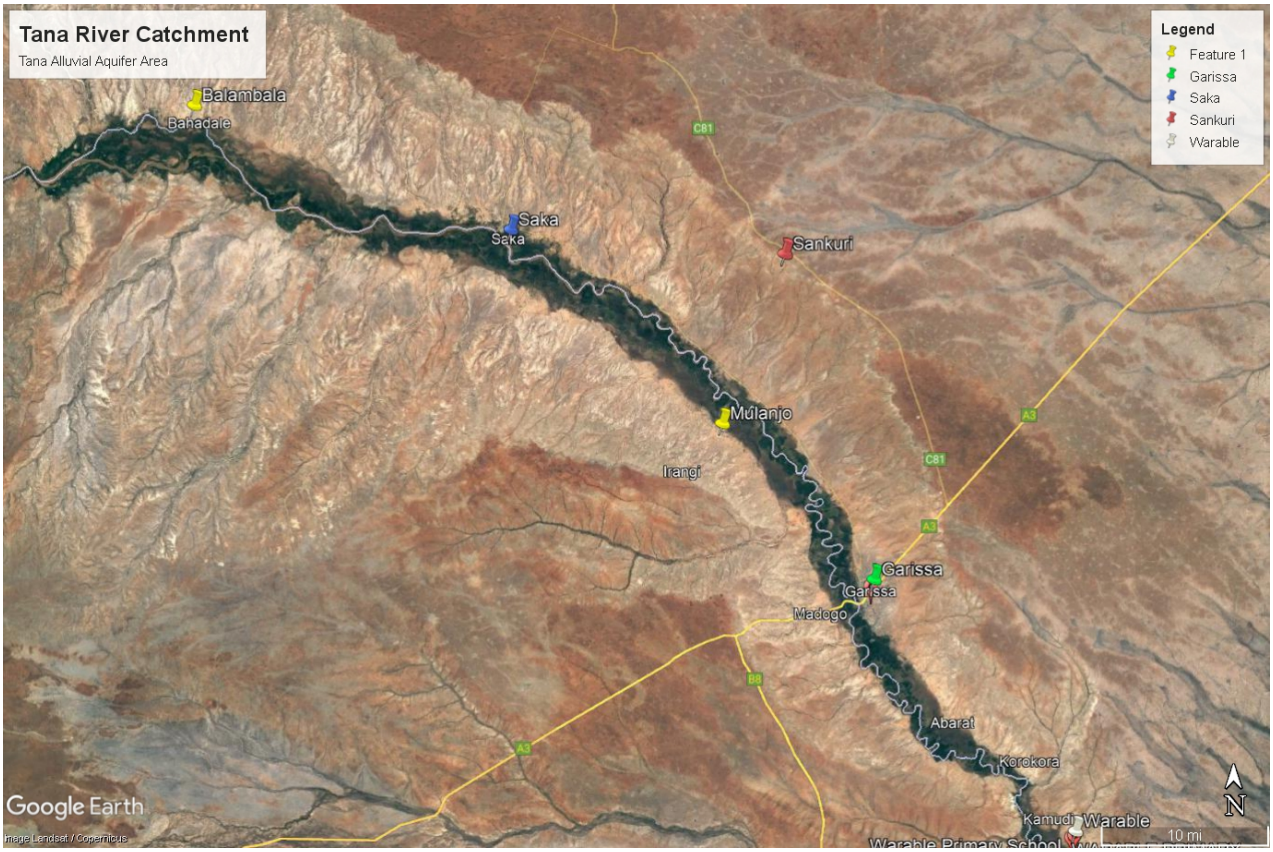


Figure 1: Satellite Imagery of the Tana Catchment

III. HYDROGEOLOGY, GEOPHYSICS AND PROBLEM STATEMENT:

The present study is motivated by the desire to build a simple predictive Hydrological Model for Decision Making in terms of siting the new and replacement wells proposed in the study area. It is to be noted that the replacement wells will be located at the same localities.

The area comprises an exclusively sedimentary suite, but some localities have fine grained aquifer material, impeding the process of recharge flow.

Some areas have medium-grained sandstones whereas others have coarse grained species of sandstones.

On the account stated above, some areas will appear to have better recharged aquifers than others and some other areas will have very fresh waters and yet others hard, saline or brackish water species. It is a paradox that some sites located very close to the river still bear saline and brackish waters. One would infer that laminar flow would offer a favorable recharge pathway in areas close to the river, thereby offering fresh water by default. As one traverses the river Tana Flow Profile from Balambala, moving towards Saka and Sankuri, the water is fresh and the aquifer materials is homogeneously medium to coarse-grained. The aquifers are thus better recharged and have high values of transmissivity. However, some unpredictable mineralization of gypsiferrous sediments hamper the uniformity, so that an area with coarse grains but located close to the river may still seem to have gypsum-rich anions and cations, rendering it saline and unsuitable for human consumption. Livestock have been found to be suited to using this water types, however. The challenge that comes up has been how to develop new wells using existing well data, to make informed decision so that if an area is already known to have saline water, an alternative area is proposed for development of the new undertaking.

In Garissa and Madogo areas, the sediments are predominantly medium to coarse grained as opposed to the exclusively coarse grains in Balambala. As one moves along the river flow course towards Korakora, warable, Jambele and Nanighi areas, the sediments of the aquifer material gradually and progressively diminish in grain size and also the clay enrichment is very high so that recharge flow dynamics is hampered. The Tana Alluvial Aquifer (TAA) has been over-abstracted and if the low rains persist, it is likely that the future five or so years will have a depleted aquifer. If the water will still be there, it will have an increased salinity in the absence of recharge flow.

The two main focus in the present study is to develop a simple model that will address the twin evils:

a) Low discharge values

b) High levels of salinity

Fuzzy Logic was opted for as it is an algorithm which leverages on previous experience and may work with limited data number, as is the case with the present study. Existing geophysical models from past work in the study area were also used in the present study.

SAMPLE GEOELECTRICAL MODEL IN BAWAMA SETTLEMENT-

The colored tomographic image of the two sites deemed favorable for drilling

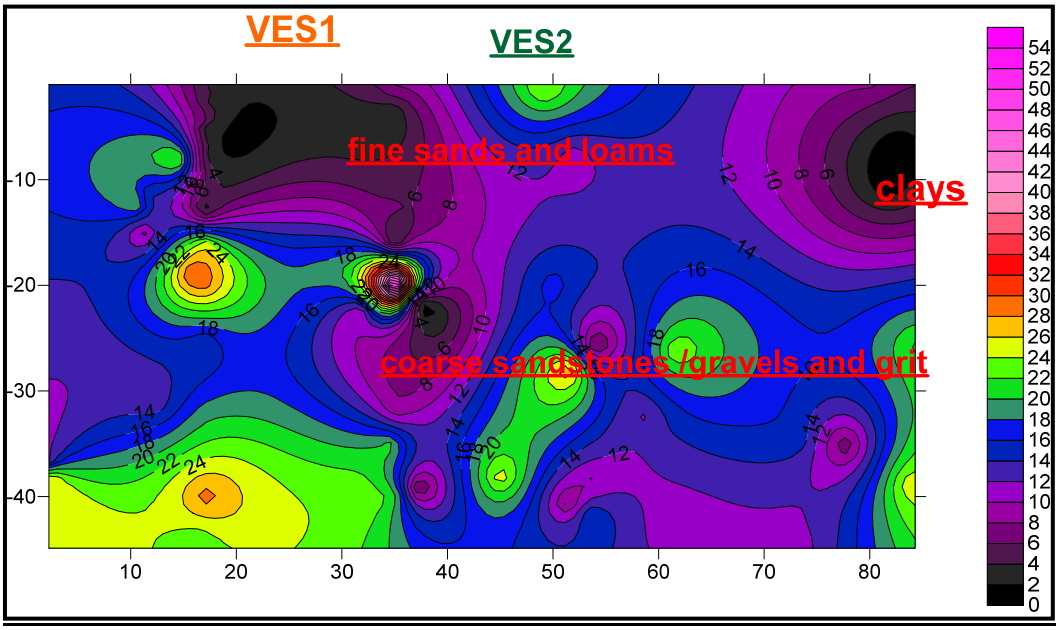


Figure 2: Geoelectrical Tomographic Imaging at the Bawama Settlement

Sample Resistivity Model 2 for Bawama

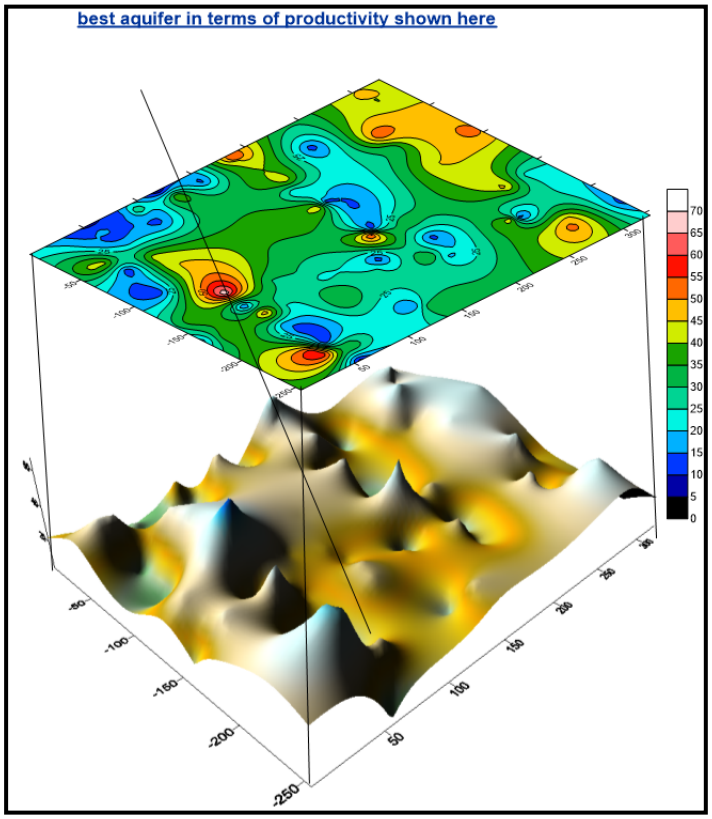


Figure 3: Resistivity Model in 3-Dimensions for Bawama area in The Tana Alluvial Aquifer

Bawama VES 02/2021

The curve of resistivity of sediments at depths showing water at the depths 32m to 45m bgl, and also at the depths of 80m to 135m bgl. The sediments are consistently saline

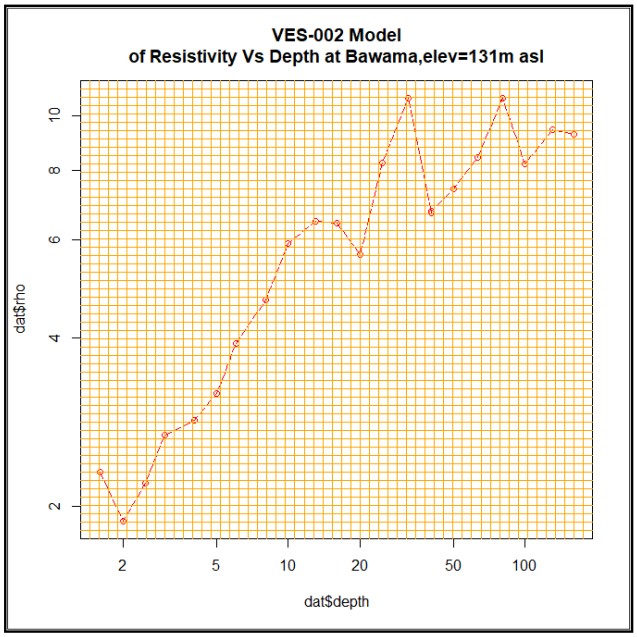


Figure 4: Resistivity Curve at Bawama, at point VES-002

TABLE 1: VES Data Bawama 001

Resistivity Curve No	**Schlumberger Probe Depth Interval(m)**	**Resistivity In OhmM**	**Expected Geological Sediment/Formation**
Sample VES in Bawama. R-001/2021 Site No 1	0-1	1.5	Top Alluvial Sediments
	1-10	5.1	Subsoils/Clayey Sediments
	10-13	5.7	Clayey Sediments
	13-16	5.7	Coarse sandstones
	16-20	4.8	sandy /Loamy Sediments
	20-32	10.1	Clays and fine sands/aquitards
	32-40	13	SANDSTONES
	40-50	13	Gravels/sandy aquifers
	50-80	15.3	Coarse sandstones
	80-100	15.3	sandy /Loamy aquifer
	100-130	13.5	Coarse to medium sst aquifer
	130-160	16	Clays Saturated sandstones/aquifer
	0ver 160	Infinity	Clays /medium sandstones

TABLE 2: VES Data Bawama 002

Resistivity Curve No	Schlumberger Probe Depth Interval(m)	Resistivity In OhmM	Expected Geological Sediment/Formation
Sample VES in Bawama. R-002/2021 Site No 2	0-1	2.5	Top Alluvial Sediments
	1-13	6.5	Subsoils/Clayey aquifers
	13-20	6	Clayey Sediments
	20-32	10.5	Coarse sandstones
	32-40	6.7	sandy /Loamy Sediments
	40-80	10.5	Clays and fine sands/aquifer
	80-100	8.5	Major sandy aquifer
	100-130	9.5	Coarse to medium sst aquifer
	130-160	9.5	Clays Saturated sandstones/aquifer
	Over 160	Infinity	Clays and sands

IV. Literature Review For Neuro-Fuzzy Assessment Of Tana Alluvial Aquifer

Fuzzy logic is an algorithm invented by Lotfi Zadeh in the 1960s (Mohebbi et al, 2021) to help out with real life simulations, with respect to the actual thinking of human brain. For example, the Boolean algebra (Sarda et al, 2020) describes an event as either good or bad, respectively coded as 1 or 0. If the event being described is a flood, it may be that the flood probability is 70 percent, so that the situation is described as 70 percent bad and 30 percent good. Fuzzy logic strives to model an event using descriptions other than bad and good, and rather opts to assign degrees of 'good' and multiple degrees of 'bad' to describe the event. Fuzzy logic has been immensely applied in hydrology (Rezaei et al, 2013). In artificial intelligence systems, fuzzy logic is used to imitate human reasoning and cognition. The real life environment does not subscribe to binary occurrences of 1 and 0, but is rather a gradual transmutation from one form, progressively grading into the other. The mathematics of neuro fuzzy inference operates on the notion that fuzzy logic should operate with 0 and 1 as extreme cases of truth, but with various intermediate degrees of truth (Behounek et al, 2006). This implies the use of graphical methods known as membership functions (Babanezhad et al, 2020).

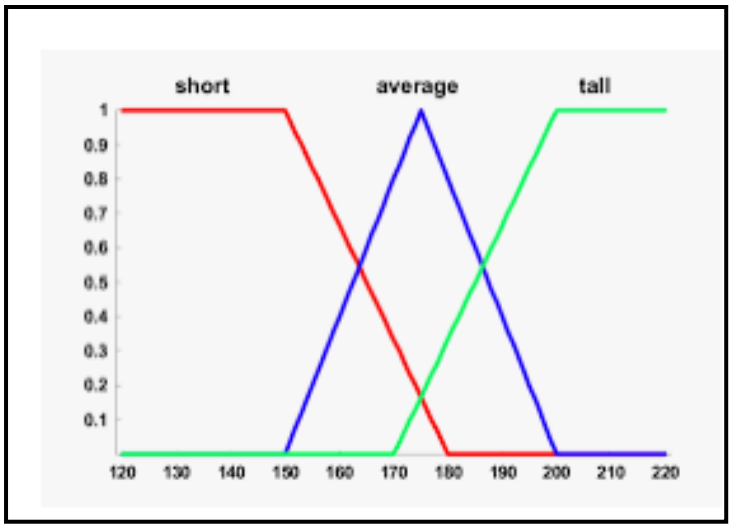


Figure 5: Illustrating the concept of membership functions. Here the person who is 'short' operates in a subset of 'tallness' and 'average' as well, via varying degrees of membership in the graph.

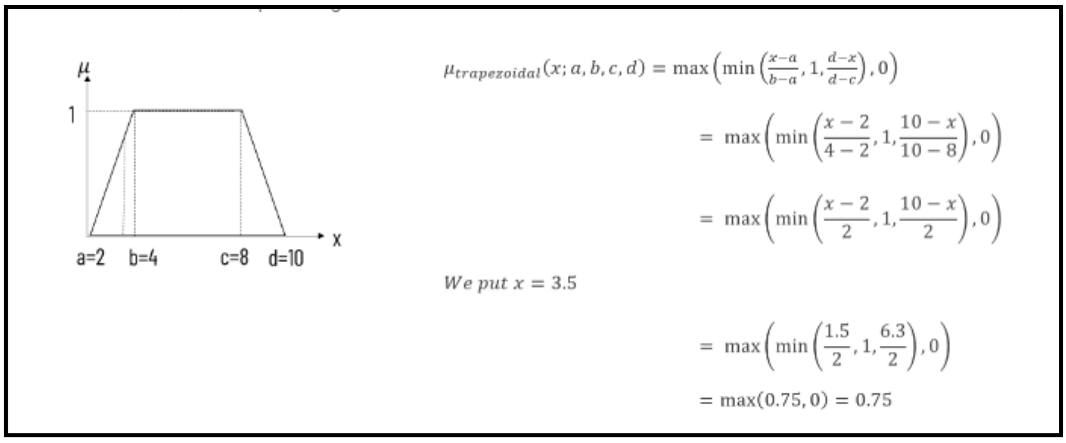
One may use the tallness or shortness to help one grasp the concept of fuzziness as used here. From the graph, the 'tall' person also has some degrees of 'shortness' as well as 'average' description. Note that the membership function value on the y-axis is the percentage representing the varying degrees of each belonging to any of the three categories. The short person has a membership function of 1.000 of being short, but has a value of 0.5 of membership function of being tall and also 0.5 being average. The tall person also has degrees of memberships to shortness and average heights as fractions of 1.000 in the graph. The above graph displays the trapezoidal membership values of the three categories of height. There are also triangular, sigmoidal and Gaussian membership functions, amongst others (Azam et al, 2020). The study used the trapezoidal memberships (Dombi et al, 2020) and the explanation rendered in the foregoing discussion illustrates the concept.

Example: Trapezoidal membership function

In the diagram hereunder, one sees a trapezoidal model graph with all the values of the various sides of this trapezium. The x dimensions are the four values, respectively spelt out as a=2, b=4, c=6, and d=10.

The formula for determining membership function value of a given new number is also stated. There are an outer and inner bracket harboring values.

As an example, the membership value of 3.5 has been computed manually and determined to be 0.75, taking the maximum of the minimum values in the inner bracketed values. The value of x in the equation stated is the new value, 3.5.



$$\mu_{trapezoidal}(x;a,b,c,d) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$$

$$= \max\left(\min\left(\frac{x-2}{4-2}, 1, \frac{10-x}{10-8}\right), 0\right)$$

$$= \max\left(\min\left(\frac{x-2}{2}, 1, \frac{10-x}{2}\right), 0\right)$$

$We\ put\ x = 3.5$

$$= \max\left(\min\left(\frac{1.5}{2}, 1, \frac{6.3}{2}\right), 0\right)$$

$$= \max(0.75, 0) = 0.75$$

Figure 6: Illustrating membership functions and computations appropriate to derive the values manually.

A. ILLUSTRATING THE WORKINGS OF A NEURO FUZZY MODEL WITH AN EXAMPLE

1. Fuzzification and Defuzzification:

The range between the values of a variable are usually broken down into three or more parts, depending on the number of classes of universe of discourse the modeler deems appropriate for the problem being solved (Bardossy et al, 1990). An illustration is paramount. Suppose we need to estimate discharge from the longitude and depths of wells drilled near a river. The tabulated info is as thus:

Item S/No	minimum	maximum	Estimated Range
longitude	0.1	500	500
depth	0.9	60	60
discharge	0.8	30	50

The following diagram shows the fuzzification of all the three variables in trapezoidal order:

```
7
8
9 Define fuzzy sets and linguistic variables longitudes
10
11  D_1 = FuzzySet(function=Triangular_MF(a=0, b=0, c=50), term="nondetectable")
12
13 D_2 = FuzzySet(function=Triangular_MF(a=50, b=100, c=150), term="vvvLow")
14
15 D_3 = FuzzySet(function=Triangular_MF(a=100, b=150, c=200), term="vvLow")
16
17 D_4= FuzzySet(function=Triangular_MF(a=150, b=200, c=250), term="vLow")
18
19 D_5 = FuzzySet(function=Triangular_MF(a=200, b=250, c=300), term="Low")
20
21 D_6 = FuzzySet(function=Triangular_MF(a=250, b=300, c=350), term="x1High")
22 D_7 = FuzzySet(function=Triangular_MF(a=300, b=350, c=400), term="x2High")
23 D_8 = FuzzySet(function=Triangular_MF(a=350, b=400, c=450), term="x3High")
24 D_9 = FuzzySet(function=Triangular_MF(a=400, b=450, c=500), term="x4High")
25 D_10 = FuzzySet(function=Triangular_MF(a=450, b=500, c=500), term="x5High")
26
27 LV13 = LinguisticVariable([D_1,D_2,D_3,D_4,D_5,D_6,D_7,D_8,D_9,D_10], concept="longitudeVV of v
28 FS.add_linguistic_variable("longtdx", LV13)
29 FS.produce_figure(outputfile='bh.pdf')
30
31
```

Figure 7: Illustrating the fuzzified values of longitudes

```
34
35
36  Define fuzzy sets and linguistic variables DEPTHS
37
38 S_1 = FuzzySet(function=Triangular_MF(a=0, b=0, c=6), term="nondetectable")
39
40 S_2 = FuzzySet(function=Triangular_MF(a=6, b=12, c=18), term="vvvLow")
41
42 S_3 = FuzzySet(function=Triangular_MF(a=12, b=18, c=24), term="vvLow")
43
44 S_4= FuzzySet(function=Triangular_MF(a=18, b=24, c=30), term="vLow")
45
46 S_5 = FuzzySet(function=Triangular_MF(a=24, b=30, c=36), term="Low")
47
48 S_6 = FuzzySet(function=Triangular_MF(a=30, b=36, c=42), term="x1High")
49 S_7 = FuzzySet(function=Triangular_MF(a=36, b=42, c=48), term="x2High")
50 S_8 = FuzzySet(function=Triangular_MF(a=42, b=48, c=54), term="x3High")
51 S_9 = FuzzySet(function=Triangular_MF(a=48, b=54, c=60), term="x4High")
52 S_10 = FuzzySet(function=Triangular_MF(a=54, b=60, c=60), term="x5High")
53
54 LV15 = LinguisticVariable([S_1,S_2,S_3,S_4,S_5,S_6,S_7,S_8,S_9,S_10], concept="depthXX of well"
55 FS.add_linguistic_variable("depthXX", LV15)
56 FS.produce_figure(outputfile='bh.pdf')
57
58
```

Figure 8: Illustrating the fuzzified values of depths

```
63
64 Define fuzzy sets and linguistic variables DISCHARGES
65 E_1 = FuzzySet(function=Triangular_MF(a=0, b=0, c=3), term="nondetectable")
66
67 E_2 = FuzzySet(function=Triangular_MF(a=3, b=6, c=9), term="vvvLow")
68
69 E_3 = FuzzySet(function=Triangular_MF(a=6, b=9, c=12), term="vvLow")
70
71 E_4= FuzzySet(function=Triangular_MF(a=9, b=12, c=15), term="vLow")
72
73 E_5 = FuzzySet(function=Triangular_MF(a=12, b=15, c=18), term="Low")
74
75 E_6 = FuzzySet(function=Triangular_MF(a=15, b=18, c=21), term="x1High")
76 E_7 = FuzzySet(function=Triangular_MF(a=18, b=21, c=24), term="x2High")
77 E_8 = FuzzySet(function=Triangular_MF(a=21, b=24, c=27), term="x3High")
78 E_9 = FuzzySet(function=Triangular_MF(a=24, b=27, c=30), term="x4High")
79 E_10 = FuzzySet(function=Triangular_MF(a=27, b=30, c=30), term="x5High")
80
81 LV16 = LinguisticVariable([E_1,E_2,E_3,E_4,E_5,E_6,E_7,E_8,E_9,E_10], concept="aquifer discharge of well",
82 FS.add_linguistic_variable("dischargeXX", LV16)
83 FS.produce_figure(outputfile='bh.pdf')
84
85
```

Figure 9: illustrating the fuzzified values of discharge

Suppose the new area has longitude and depth as 150 and 19 respectively. The task is now to compute the values of discharge using the principles of fuzzy logic (Tzimopoulos et al, 2016).

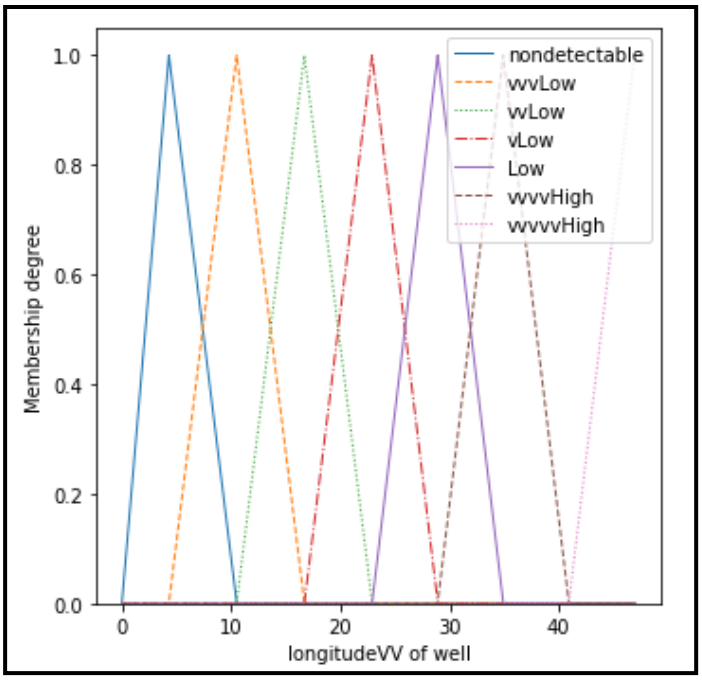


Figure 10: The MF of longitudes

Steps involved are stated hereunder.

1. *Derive Membership value of longitude*

The value of 150 represents 150 units of longitude whose total value is 500. The graph of longitudes show that 150 stands at an mf value of 0.4. Dividing 150 by 500 is 150/500 which gives a value of 0.3. What we wish to predict is discharge whose total range is 30. If we multiply 30 times 0.3 above, we obtain 9.0 as our value.

2. *Multiply MF by 9*

The value of 9 times 0.4, we get 3.6

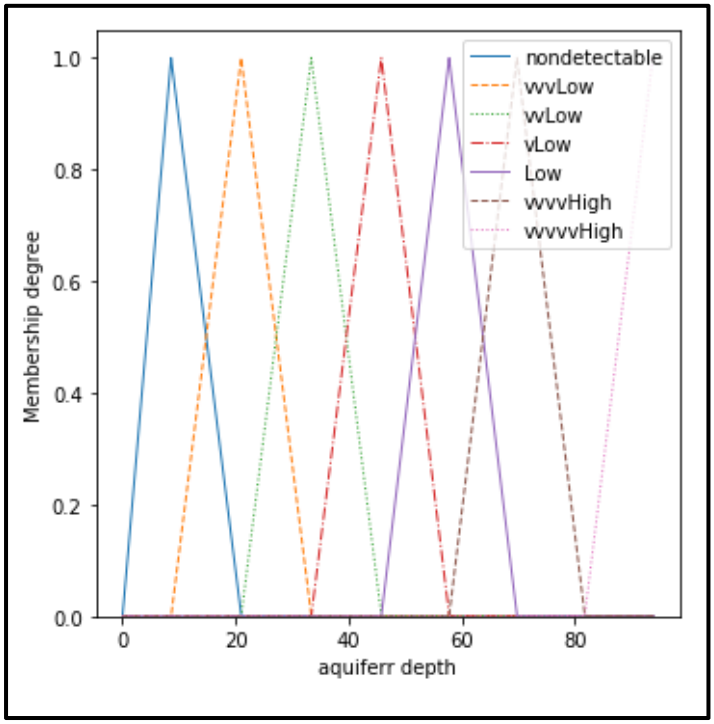


Figure 11: the MF graph for depths

Steps involved are hereunder:

a) Derive membership value of longitude

The value of 19 represents 60 units of depth whose total value is 60 in the Universe of Discourse (Chen et al, 2019). The graph of depths show that 19 stands at an mf value of 0.2.Dividing 19 by 60 is 19/60 which gives a value of 0.316. What we wish to predict is discharge whose total range is 30.if we multiply 30 times 0.316 above, we obtain 9.5 as our value.

b) Multiply MF by 9.5

The value of 9.5 times 0.2, we get 1.9

B. FUZZY RULES FOR PREDICTION

Consider the following two rules in the fuzzy rule base.

RULE1: If x is A and y is P then z is C1

RULE2: If x is B then y is Q then z is C2

In the example illustrated here, three rules (one may replace RULE1 with R1) are quoted hereunder:

R1 = "IF (longtdx IS nondetectable) AND (depthXX IS nondetectable) THEN (dischargeXX IS nondetectable)"

R2 = "IF (longtdx IS vvvLow) AND (depthXX IS vvvLow) THEN (dischargeXX IS vvvLow)"

R3 = "IF (longtdx IS vvLow) AND (depthXX IS vvLow) THEN (dischargeXX IS vvLow)"

These rules will form the basis of modeling with words rather than numbers, and at the end of it, during predictions, the rules will be applied to defuzzify the variables to obtain a crisp output using the fuzzy rule-based engine.

Defuzzification –The values we need are crisp or actual values. To get this, one needs to defuzzify the data ranges availed. This involves summing up the values obtained in 'a' and 'b' above then dividing by weighted sum, which is 0.4 added to 0.2, that is the sum of the two membership functions derived from physically observing the mf graphs generated in python.

The procedure of generating crisp values in this way is known as the average weighting used in the Sugeno algorithm or method (Moorthi et al, 2018). The combined result derived for discharge is as thus:

v=0.4*9.0+0.2*9.5/(0.4+0.2)

v=(3.6+1.9)/(0.6)

v=9.00

One may then run the python script and get values for comparison.

```
106 FS.add_rules([R1,R2,R3,R4,R5,R6,R7,R8,R9,R10])
107
108
109 # Set antecedents values
110
111 FS.set_variable("longtdx", 150)
112 FS.set_variable("depthXX", 19)
113 print(FS.Mamdani_inference(["dischargeXX"]))
114
115
```

The ouput is more or less simalr to the gpahically derived long procedure highlighted, thus.

```
    ...:
    ...: v/(0.4+0.2)
Out[43]: 9.166666666666666
```

Accuracy=9.00/9.16

Accuracy=98.3 percent

In retrospect, fuzzification converts the crisp input into fuzzy value (Starczewski et al, 2020).

This is the direct opposite of defuzzification, which is the process that generates the actual prediction. Defuzzification converts the fuzzy input of fuzzy inference engine into crisp value, so that a crisp value may be generated (Pourabdollah et al, 2020). Selection of defuzzification procedure depends on the properties of the application. Fuzzy logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" (1 or 0) Boolean logic on which the modern computer is based (Bělohlávek et al, 2017) .The idea of fuzzy logic has really simplified and advanced mathematical modeling in instances where data is scarce, in the field of hydrology. Zadeh was working on the problem of computer understanding of natural language (Gentili et al, 2017). Natural language -,like most other activities in life and indeed the universe ,is not easily translated into the absolute terms of 0 and 1. Whether everything is ultimately describable in binary terms is a philosophical question worth pursuing, but in practice, much data we might want to feed a computer is in some state in between and so, frequently, are the results of computing. It may help to see fuzzy logic as the way reasoning really works and binary, or Boolean, logic is simply a special case of it (Fisher et al, 2020).

C. LITERATURE REVIEW FOR DECISION TREE ASSESSMENT OF TANA ALLUVIAL AQUIFER

Decision tree finds useful applications in assessing groundwater hydrology (Nguyen et al, 2020). One need a brief background into how the method really works before applying it to solve problems in hydrology.

Machine Learning algorithms help simplify tasks in data analytics and this has found useful applications in research tasks within the field of hydrology (Lange et al, 2020). Primarily, the best predictors or classifiers are the tree-based algorithms, namely, the decision trees and random forest ensembles. The Decision Tree (DT) is designed to look like a flow chart structure where an internal organ may represent a feature being mapped/analyzed in the data frame used (Lan et al, 2020). The tree branch represents a decision rule which has been founded on the basis of gini index values (Tangirala et al, 2020. In this respect, the uppermost node at the top is known as the

Root node as per the research undertaken by Aldino et al (2020). This thus makes the DT model assume the looks of an inverted tree trunk, with its branches and leaves intact. The DT is simple to understand as its visualization charts mimic human brains intelligence thinking protocols that are in return quite easy to follow.

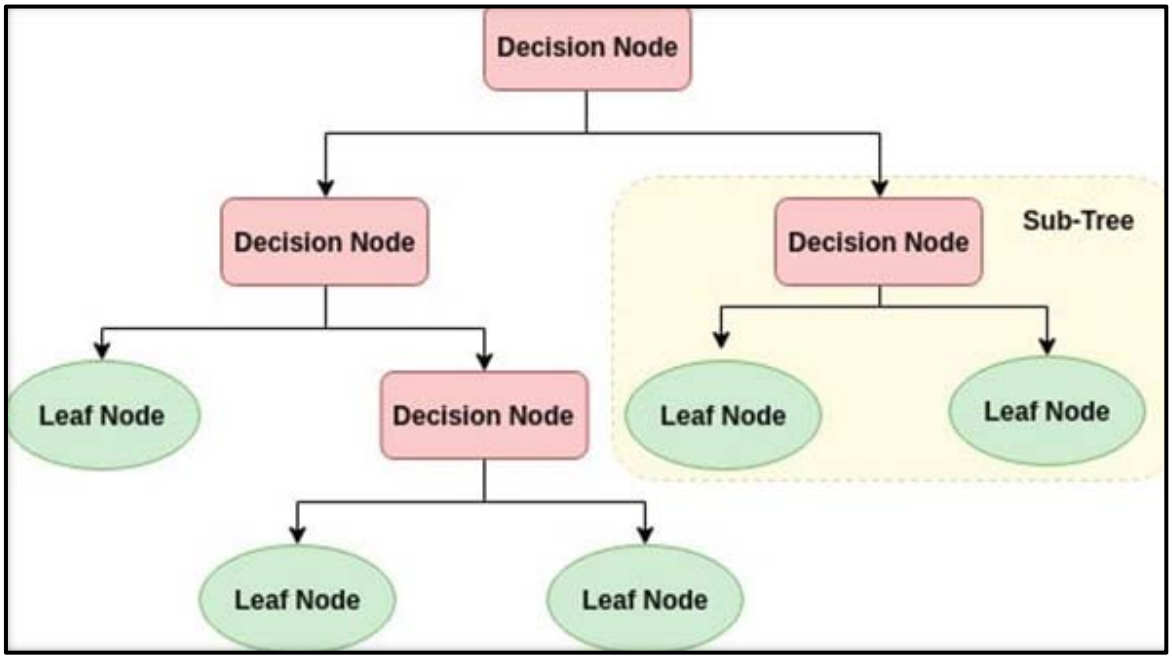


Figure 12: DT Visualization Model

As opposed to neural networks that are black-box ML models (Wang et al, 2020), the DT models may be comparatively termed as white box, as the model understands fully what proceed at each node during model development phase. No prior statistical-distribution may be assumed while using the DT models and this makes it a user-friendly algorithm as it works with any kind of data frames. For the non-statistician undertaking research, this is a welcome situation as the encumbrance of having to fit a specific distribution type is foregone. The artificial neural network back propagation (Purba et al, 2020) may take minutes before retuning an output after convergence. In the case of DT models, this is not the case as the results are rapidly generated in the output tables. This is irrespective of whether the data frames being modeled were high dimensional or low dimensional (Jiang et al, 2018).

1. *The workings of a Decision Tree Model*

The following steps sum up the main procedures involved in splitting tree and using it to predict desired model variables or attributes:

1. Employ the attribute selection measures to help split the trees. Many a times, one can also just observe the data set rows and pin point the variable that explains the class differences the most, within then data frame. It is this variable one may target for splitting.
2. Strive to make that attribute being split to become the decision node, so that it is used to help breaks the dataset into smaller subsets.

3. Begin the recursive process of tree building for each child generated by the chosen node, way until one of the following happens:

    a) A stage is attained when all the tuples belong to the same attribute value.

    b) A state is reached when there are no more remaining attributes to be split further in the tree/node models.

    c) A state is attained when any further splits or attempts at splitting any node makes no sense at all.

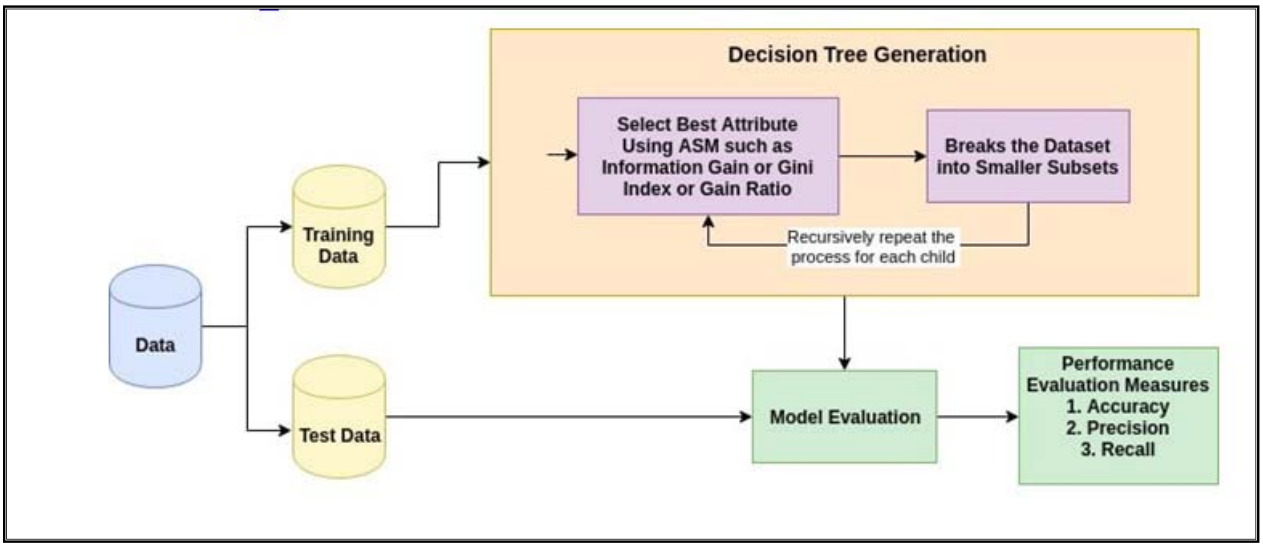


Figure 13: Decision Tree Generation

2. *The stage of selecting the variable to be used in nodal splits*

The process of picking the appropriate attribute is a sensitive heuristic, as the criteria adopted for nodal splits will be the template that helps us split the data in the best way possible with respect to the data being modeled (Zhu et al, 2020). It is the most important phase of DT model development since it helps us establish the breakpoints/terminal level at the chosen nodes being split, at whose levels/values no further splits are feasible (Cortes et al, 2020). The data frame may comprise class variables, also termed as categorical values and this may include a list of attributes such as saline, fresh or hard water species being abstracted from an aquifer. The other values that are discrete are termed continuous, namely, aquifer depths and discharges. To actualize the selection measures for nodal splitting to aid DT modeling the following three parameters may be employed:

a) information Gain,

b) Gain Ratio,

c) Gini Index.

3. *Information Gain*

This is an essential concept in information sciences and systems engineering. The concept of information gain (Wang et al, 2019) is essential in developing hydrological and environmental models using the DT protocols. It was developed by Shannon, and it primarily measures the degree of impurity of a system (Gonoordi et al,2019). Impurity as used in this explanation may be explained as thus. If we are in a localized aquifer, and assuming we are looking at wells spaced out at the intervals of every 100m. Suppose we have seven wells , such that wells numbers one to seven are of the following water quality types: saline, saline, saline, saline, saline, saline, and finally, fresh. In physics and mathematics, entropy referred as the randomness or the impurity in the system.

Suppose in zone B we now have another seven wells spaces out also at 100m each: Saline, fresh, fresh, saline, fresh, saline, and fresh. In this case, the second zone is more impure as it harbors almost an equal number of saline and fresh water wells. The first case has a uniform set of exclusively saline wells with the exception of the very last aquifer. The zone B is thus more impure than zone A. This concept of measuring impurity in a system or a group /set of variables was termed entropy by Shannon.

Information gain is the measure of decrease in entropy levels in a system (Nourani et al, 2019). In modeling using ML decision tree methods, Information Gain helps compute the arithmetic difference between entropy levels of a system before split, and average entropy after split of the dataset based on given attribute values, on the other hand. There is a DT algorithm known as the ID3, short for the Iterative Dichotomiser, and is a famous DT algorithm which uses Information Gain criterion to develop predictive analytical models.

It is illustrated as hereunder:

$$\text{Info(D)} = -\sum_{i=1}^{m} pi \log_2 pi$$

In the above case shown, the value 'pi' represents the probability that an arbitrary tuple in D belongs to class Ci.

$$\text{Info}_A(D) = \sum_{j=1}^{V} \frac{|Dj|}{|D|} \text{ X Info}(D_j)$$

$$\text{Gain(A)} = \text{Info(D)} - \text{Info}_A(D)$$

The variables analyzed are as thus defined:

(i) Info(D) represents the average amount of information needed to identify the class label of a tuple in D.

(ii) |Dj|/|D| acts as the weight of the jth partition.

(iii) InfoA(D) is the expected information required to classify a tuple from D based on the partitioning by A.

The attribute A with the highest information gain, Gain(A), is chosen as the splitting attribute at node N(). Assuming that attribute represented were the class value of an aquifer, information gain A would be used to split the predictor variables until such a time that the least value is gained on further split, then the appropriate computations are undertaken to determine the class of the row being mapped using Decision Tree.

4. Gain Ratio

The Gain ratio (Dou et al, 2019) covers up for the weaknesses inherent in the Information Gain based models (). As much as it has been illustrated in preceding discussion on just how useful the information gain concept is good, it has been found to be such a biased tool, with respect to mapping an attribute with many possible classes or outcomes. This means that the IG will by default prefer the attribute with a large number of distinct values. Take the example of an attribute with a unique identifier, namely, the customer_ID with a zero info (D) because of pure partition. This has problems during model development using the DT algorithm to the extent that it maximizes the information gain, thereby generating a useless partitioning profile for the data frame. To overcome this, an upgrading of ID3 was developed, and named C$.5. This C4.5 was an improvement of ID3, and employed then for use as an extension to information gain, which came to be known as the gain ratio. This tool, Gain Ratio, handles the issue of bias by normalizing the information gain using Split Info.

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

In the above equation, one notes that:

a) |Dj|/|D| acts as the weight of the jth partition.

b) v is the number of discrete values in attribute A.

Consequently, from the foregoing discussion, the gain ratio may be thus defined:

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo_A(D)}$$

The attribute dataframe variable/column bearing the highest gain ratio would be chosen as the splitting attribute, so that the appropriate DT algorithm is developed for modeling (Alsalman et al, 2019)

5. Gini Index

The final subset of the DT algorithm development is the Gini Index (Roy et al, 2019). It is to be noted that another decision tree algorithm, CART (Classification and Regression Tree), uses the Gini method to create split points (Zimmerman et al, 2021).

$$\text{Gini(D)}=1-\sum_{i=1}^{m}\text{Pi}^2$$

In the above equation, 'pi' is the probability that a tuple in D belongs to class Ci.

The gini index factors a binary split for every attribute, and one may compute then weighted sum of impurity of each partition. In the event that a binary split on attribute A partitions data D into D1 and D2, the Gini index of D may be attained as per the calculations hereunder:

$$\text{Gini}_A(D)=\frac{|D1|}{|D|}\text{Gini}(D_1)+\frac{|D2|}{|D|}\text{Gini}(D_2)$$

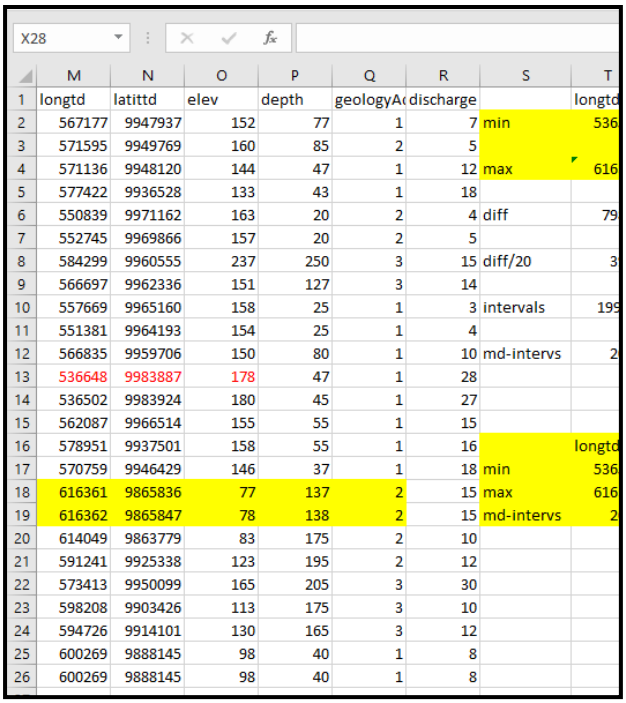
Suppose one wish to undertake models using the longitudes, latitudes, elevations, depths and discharges of a localized aquifer to maybe predict aquifer class. The class attribute may be mapped in such a way that the specific class subset which generates the minimum gini index would be selected as the splitting attribute' One may wish to predict discharge rather than class. This is a case of DT regression of a data frame (Ghosh et al, 2021). In that instance, since the discharge is a continuous valued attribute, the strategy is to select each pair of adjacent values as a possible split-point and point with smaller gini index chosen as the splitting point (Du et al, 2021).

$$\Delta Gini(A) = Gini(D) - Gini_A(D).$$

V. The Assessment Of Tana Alluvial Aquifer Discharge Using Neuro- Fuzzy Inference

A. Actual work done both in the field and data analysis

The data generated from wells data of boreholes sank along the Tana Alluvial aquifer in Madogo and Garissa areas , extending all the way to Garsweino along the River tana flow profile were assembled in excel and analysed. The data is scanty and may not merit modeling using conventional statistical methods, hence the decision to employ neuro fuzzy inference engines. The process of compiling this data was preceded by a three week field work where the wells were visited, and had their GPS points andavailable hydrologic data taken and conter-verified, via the Water Resources Authority offices in the Lower Tana Catchment.



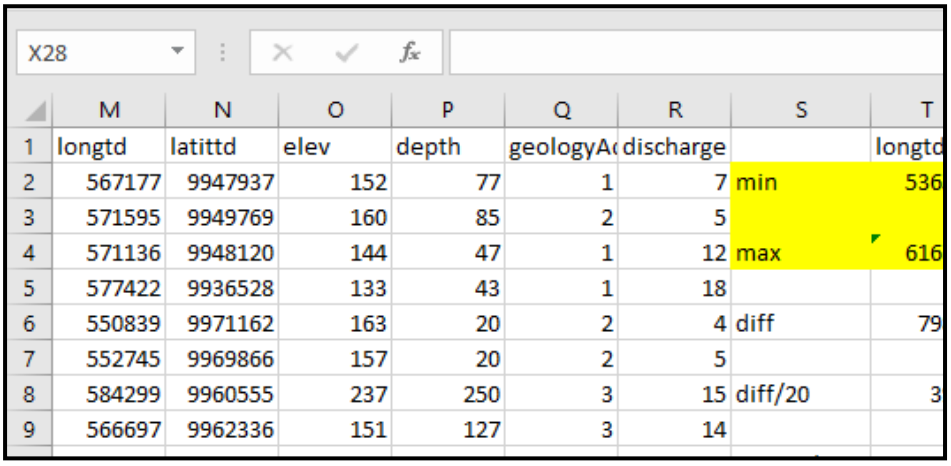
	M	N	O	P	Q	R	S	T
1	longtd	latittd	elev	depth	geologyA	discharge		longtd
2	567177	9947937	152	77	1	7	min	536
3	571595	9949769	160	85	2	5		
4	571136	9948120	144	47	1	12	max	616
5	577422	9936528	133	43	1	18		
6	550839	9971162	163	20	2	4	diff	79
7	552745	9969866	157	20	2	5		
8	584299	9960555	237	250	3	15	diff/20	3
9	566697	9962336	151	127	3	14		
10	557669	9965160	158	25	1	3	intervals	199
11	551381	9964193	154	25	1	4		
12	566835	9959706	150	80	1	10	md-intervs	2
13	536648	9983887	178	47	1	28		
14	536502	9983924	180	45	1	27		
15	562087	9966514	155	55	1	15		
16	578951	9937501	158	55	1	16		longtd
17	570759	9946429	146	37	1	18	min	536
18	616361	9865836	77	137	2	15	max	616
19	616362	9865847	78	138	2	15	md-intervs	2
20	614049	9863779	83	175	2	10		
21	591241	9925338	123	195	2	12		
22	573413	9950099	165	205	3	30		
23	598208	9903426	113	175	3	10		
24	594726	9914101	130	165	3	12		
25	600269	9888145	98	40	1	8		
26	600269	9888145	98	40	1	8		

The data then was subjected to the following:

a) Each column was analysed with a view of fuzzifying it. The range within the data of latitudes for example was 9863779 and 9983924. The same was done to longitudes, elevations, depths, and discharge values mapped in the area.

b) Fuzzification was then undertaken, by generating at least 20 or so differnet categories in the Universe of Discourse, which simply means discretizing the values into at least different classes or scales, but located well within that range defined. For example , the least scale of longitude was named as **'vvvvsmall'.** The maximum of the longitude values as thus defined was named as **'verylong5'. These terms are subjective and any modeller may use his/her own wordings.**

c) The data screenshot is displayed hereunder :



	M	N	O	P	Q	R	S	T
1	longtd	latittd	elev	depth	geologyA	discharge		longtd
2	567177	9947937	152	77	1	7	min	536
3	571595	9949769	160	85	2	5		
4	571136	9948120	144	47	1	12	max	616
5	577422	9936528	133	43	1	18		
6	550839	9971162	163	20	2	4	diff	79
7	552745	9969866	157	20	2	5		
8	584299	9960555	237	250	3	15	diff/20	3
9	566697	9962336	151	127	3	14		

d) The fuzzy rules were then generaed for the data, based on what was conceivable from visual observation of tha dataset.As an example to illustate the statement used here, **consider row number one.** The longitue is 567177. In the fuzzification profile developed, this value is categorised as 'x2average'.

```
#LONGITUDES
D1 = sf.TrapezoidFuzzySet(a=0, b=0, c=536502, d=538502, term="vvvvsmall")
D2 = sf.TrapezoidFuzzySet(a=536502, b=538502, c=540502, d=542502, term="vvvsmall")
D3 = sf.TrapezoidFuzzySet(a=540502, b=542502, c=544502, d=546502, term="vvsmall")
D4 = sf.TrapezoidFuzzySet(a=544502, b=546502, c=548502, d=550502, term="vsmall")

D5 = sf.TrapezoidFuzzySet(a=548502, b=550502, c=552502, d=554502, term="small")

D6 = sf.TrapezoidFuzzySet(a=552502, b=554502, c=556502, d=558502, term="average")

D7 = sf.TrapezoidFuzzySet(a=556502, b=558502, c=560502, d=562502, term="x0average")
D8 = sf.TrapezoidFuzzySet(a=560502, b=562502, c=564502, d=566502, term="x1average")
D9 = sf.TrapezoidFuzzySet(a=564502, b=566502, c=568502, d=570502, term="x2average")
D10 = sf.TrapezoidFuzzySet(a=568502, b=570502, c=572502, d=574502, term="x3average")

D11 = sf.TrapezoidFuzzySet(a=572502, b=574502, c=576502, d=578502, term="x4average")
D12 = sf.TrapezoidFuzzySet(a=576502, b=578502, c=580502, d=582502, term="x5average")
D13 = sf.TrapezoidFuzzySet(a=580502, b=582502, c=584502, d=586502, term="x6average")
D14 = sf.TrapezoidFuzzySet(a=584502, b=586502, c=588502, d=590502, term="x7average")
D15 = sf.TrapezoidFuzzySet(a=588502, b=590502, c=592502, d=594502, term="x8average")
D16 = sf.TrapezoidFuzzySet(a=592502, b=594502, c=596502, d=598502, term="x9average")
```

The fuzzy rule for the whole of that row is hereby stated:

R1 = "IF(longtd IS x2average) OR (lattd IS x8average) AND (elev IS x3average) AND (depth IS average) THEN (discharge IS vvsmall)"

This was done for all the rows stated in the excel sheet of the wells data used to map the Tana Alluvial Aquifer Hydrology. Twenty such rows were realized. The data was then entered into excels Anaconda GUI for modeling using the Simpfl* library of python. Trapezoidal membership functions were then used to compute the required discharge values in crisp form.

Trapezoidal membership function:

Trapezoidal membership function is defined by four parameters: a, b, c and d. Span b to c represents the highest membership value that element can take. And if x is between (a, b) or (c, d), then it will have membership value between 0 and 1.

Assume the trapezium hereunder represents the values being used to model an aquifer.

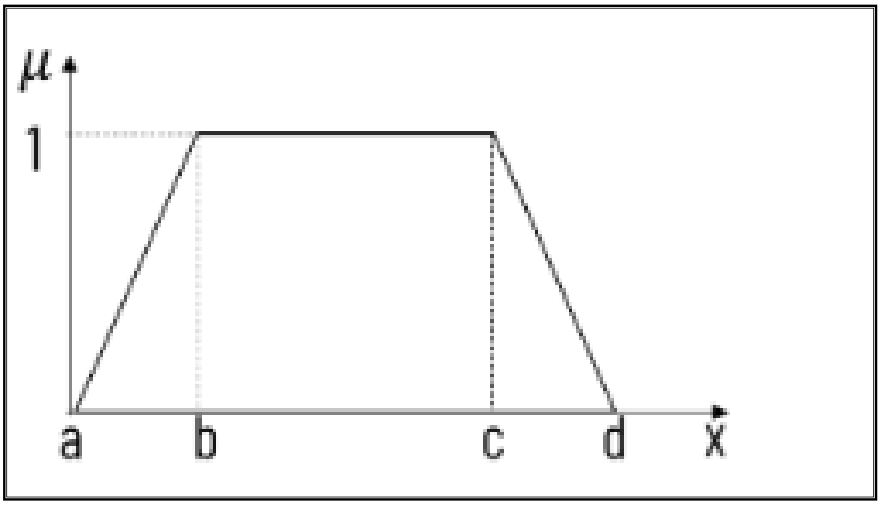


Figure 14: Trapezoidal membership function components defined.

The formula for computing membership functions of a trapezoidal function is as thus:

$$\mu_{trapezoidal}(x;a,b,c,d) = \begin{cases} 0, & x \le a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{cases}$$

$$= \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$$

Longitudes fuzzification screenshot looked thus:

```
#LONGITUDES
D1 = sf.TrapezoidFuzzySet(a=0, b=0, c=536502, d=538502, term="vvvvsmall")
D2 = sf.TrapezoidFuzzySet(a=536502, b=538502, c=540502, d=542502, term="vvvsmall")
D3 = sf.TrapezoidFuzzySet(a=540502, b=542502, c=544502, d=546502, term="vvsmall")
D4 = sf.TrapezoidFuzzySet(a=544502, b=546502, c=548502, d=550502, term="vsmall")

D5 = sf.TrapezoidFuzzySet(a=548502, b=550502, c=552502, d=554502, term="small")

D6 = sf.TrapezoidFuzzySet(a=552502, b=554502, c=556502, d=558502, term="average")

D7 = sf.TrapezoidFuzzySet(a=556502, b=558502, c=560502, d=562502, term="x0average")
D8 = sf.TrapezoidFuzzySet(a=560502, b=562502, c=564502, d=566502, term="x1average")
D9 = sf.TrapezoidFuzzySet(a=564502, b=566502, c=568502, d=570502, term="x2average")
D10 = sf.TrapezoidFuzzySet(a=568502, b=570502, c=572502, d=574502, term="x3average")

D11 = sf.TrapezoidFuzzySet(a=572502, b=574502, c=576502, d=578502, term="x4average")
D12 = sf.TrapezoidFuzzySet(a=576502, b=578502, c=580502, d=582502, term="x5average")
D13 = sf.TrapezoidFuzzySet(a=580502, b=582502, c=584502, d=586502, term="x6average")
D14 = sf.TrapezoidFuzzySet(a=584502, b=586502, c=588502, d=590502, term="x7average")
D15 = sf.TrapezoidFuzzySet(a=588502, b=590502, c=592502, d=594502, term="x8average")
D16 = sf.TrapezoidFuzzySet(a=592502, b=594502, c=596502, d=598502, term="x9average")
D17 = sf.TrapezoidFuzzySet(a=596502, b=598502, c=600502, d=602502, term="verylong1")
D18 = sf.TrapezoidFuzzySet(a=600502, b=602502, c=604502, d=606502, term="verylong2")
D19 = sf.TrapezoidFuzzySet(a=604502, b=606502, c=608502, d=610502, term="verylong3")
D20 = sf.TrapezoidFuzzySet(a=608502, b=610502, c=612502, d=614502, term="verylong4")
D21 = sf.TrapezoidFuzzySet(a=612502, b=614502, c=616361, d=616361, term="verylong5")
FS.add_linguistic_variable("longtd", LinguisticVariable([D1, D2,D3,D4,D5,D6,D7,D8,
D9,D10,D11,D12,D13,D14,D15,D16,D17,D18,D19,D20,D21],
concept="longitude of the well mapped ",universe_of_discourse=[0,616361]))
```

Figure 15: sample fuzzification of longitude values of the Tana Alluvial aquifer developed for modelling using sugeno algorithms

The latitudes fuzzufied profile looked thus:

```
#LATTITUDE
L1 = sf.TrapezoidFuzzySet(a=0, b=0, c=9863779, d=9866779, term="vvvvsmall")
L2 = sf.TrapezoidFuzzySet(a=9863779, b=9866779, c=9869779, d=9872779, term="vvvsmall")
L3 = sf.TrapezoidFuzzySet(a=9869779, b=9872779, c=9875779, d=9878779, term="vvsmall")
L4 = sf.TrapezoidFuzzySet(a=9875779, b=9878779, c=9881779, d=9884779, term="vsmall")

L5 = sf.TrapezoidFuzzySet(a=9881779, b=9884779, c=9887779, d=9890779, term="small")

L6 = sf.TrapezoidFuzzySet(a=9887779, b=9890779, c=9893779, d=9896779, term="average")

L7 = sf.TrapezoidFuzzySet(a=9893779, b=9896779, c=9899779, d=9902779, term="x0average")
L8 = sf.TrapezoidFuzzySet(a=9899779, b=9902779, c=9905779, d=9908779, term="x1average")
L9 = sf.TrapezoidFuzzySet(a=9905779, b=9908779, c=9911779, d=9914779, term="x2average")

L10 = sf.TrapezoidFuzzySet(a=9911779, b=9914779, c=9917779, d=9920779, term="x3average")
L11 = sf.TrapezoidFuzzySet(a=9917779, b=9920779, c=9923779, d=9926779, term="x4average")
L12 = sf.TrapezoidFuzzySet(a=9923779, b=9926779, c=9929779, d=9932779, term="x5average")
L13 = sf.TrapezoidFuzzySet(a=9929779, b=9932779, c=9935779, d=9938779, term="x6average")
L14 = sf.TrapezoidFuzzySet(a=9935779, b=9938779, c=9941779, d=9944779, term="x7average")

L15 = sf.TrapezoidFuzzySet(a=9941779, b=9944779, c=9947779, d=9950779, term="x8average")
L16 = sf.TrapezoidFuzzySet(a=9947779, b=9950779, c=9953779, d=9956779, term="x9average")
L17 = sf.TrapezoidFuzzySet(a=9953779, b=9956779, c=9959779,d=9962779, term="x10average")
L18 = sf.TrapezoidFuzzySet(a=9959779,b=9962779, c=9965779, d=9968779, term="x11average")
L19 = sf.TrapezoidFuzzySet(a=9965779, b=9968779, c=9971779, d=9972779, term="x12average")
L20 = sf.TrapezoidFuzzySet(a=9971779, b=9972779, c=9975779, d=9978779, term="x13average")
L21 = sf.TrapezoidFuzzySet(a=9975779, b=9978779, c=9983924, d=9983924, term="x14average")

FS.add_linguistic_variable("lattd", LinguisticVariable([L1, L2,L3,L4,L5,L6,L7,L8,
L9,L10,L11,L12,L13,L14,L15,L16,L17,L18,L19,L20,L21],
concept="latitude of well mappd ",universe_of_discourse=[0,9983924]))
```

Figure 16: sample fuzzification of latitude values of the Tana Alluvial aquifer developed for modelling using Sugeno algorithms

The discharge of the area was fuzzified and the profile looked thus:

```
# discharge of aquifer
Q1 = sf.TrapezoidFuzzySet(a=0, b=0, c=3, d=3.7, term="vvvvsmall")
Q2 = sf.TrapezoidFuzzySet(a=3.7, b=4.4, c=5.1, d=5.8, term="vvvsmall")
Q3 = sf.TrapezoidFuzzySet(a=5.1, b=5.8, c=6.5, d=7.2, term="vvsmall")
Q4 = sf.TrapezoidFuzzySet(a=6.5, b=7.2, c=7.9, d=8.6, term="vsmall")
Q5 = sf.TrapezoidFuzzySet(a=7.9, b=8.6, c=9.3, d=10, term="small")

Q6 = sf.TrapezoidFuzzySet(a=9.3, b=10, c=10.7, d=11.4, term="average")
Q7 = sf.TrapezoidFuzzySet(a=10.7, b=11.4, c=12.1, d=12.8, term="x0average")
Q8 = sf.TrapezoidFuzzySet(a=12.1, b=12.9, c=13.6, d=14.3, term="x1average")
Q9 = sf.TrapezoidFuzzySet(a=13.6, b=14.3, c=15, d=15.7, term="x2average")
Q10 = sf.TrapezoidFuzzySet(a=15, b=15.7, c=16.4, d=17.1, term="x3average")

Q11 = sf.TrapezoidFuzzySet(a=16.4, b=17.1, c=17.8, d=18.5, term="x4average")
Q12 = sf.TrapezoidFuzzySet(a=17.8, b=18.5, c=19.2, d=19.9, term="x5average")
Q13 = sf.TrapezoidFuzzySet(a=19.2, b=19.9, c=20.6, d=21.3, term="x6average")
Q14 = sf.TrapezoidFuzzySet(a=20.6, b=21.3, c=22, d=22.7, term="x7average")
Q15 = sf.TrapezoidFuzzySet(a=22, b=22.7, c=23.4, d=24.1, term="x8average")
Q16 = sf.TrapezoidFuzzySet(a=23.4, b=24.1, c=24.8, d=25.5, term="x9average")
Q17 = sf.TrapezoidFuzzySet(a=24.8, b=25.5, c=26.2, d=26.9, term="x10average")
Q18 = sf.TrapezoidFuzzySet(a=26.2, b=26.9, c=27.6, d=28.3, term="x11average")
Q19 = sf.TrapezoidFuzzySet(a=27.6, b=28.3, c=30, d=30, term="verylong")

FS.add_linguistic_variable("discharge", LinguisticVariable([Q1, Q2,Q3,Q4,Q5,Q6,Q7,Q8,
Q9,Q10,Q11,Q12,Q13,Q14,Q15,Q16,Q17,Q18,Q19],
```

Figure 17: sample fuzzification of discharge values of the Tana Alluvial aquifer developed for Modelling using sugeno algorithms

VI. DATA ANALYSIS

Consider tha data whose coordinates have been entered in the python Anaconda Console for prediction. The area needs to have a replacement well, but the original well has 12 .0 cubic meters per hour.

```
287
288  FS.set_variable("longtd", 594726)
289
290  FS.set_variable("lattd", 9914101)
291
292  FS.set_variable("elev", 130)
293
294  FS.set_variable("depth", 165)
295
296  # Perform Sugeno inference and print output
297
298  predictedValue1=(FS.Sugeno_inference(["discharge"]))
299
300  predictedValue1
301
302
```

Figure 18: The Sugeno Inference print output

The prediction is now shown hereunder:

```
   ...:
   ...: FS.set_variable("longtd", 594726)
   ...:
   ...: FS.set_variable("lattd", 9914101)
   ...:
   ...: FS.set_variable("elev", 130)
   ...:
   ...: FS.set_variable("depth", 165)
   ...:
   ...: # Perform Sugeno inference and print output
   ...:
   ...: predictedValue1=(FS.Sugeno_inference(["discharge"]))
   ...:
   ...: predictedValue1
Out[89]: {'discharge': 12.8}
```

Figure 19: The Predicted Discharge Value

One can see clearly that the answer /output predicted is 12.8. we may compute the accuracy of the algorithm as thus:

Accuracy=100x(12.0/12.8)

Accuracy= 94.0%

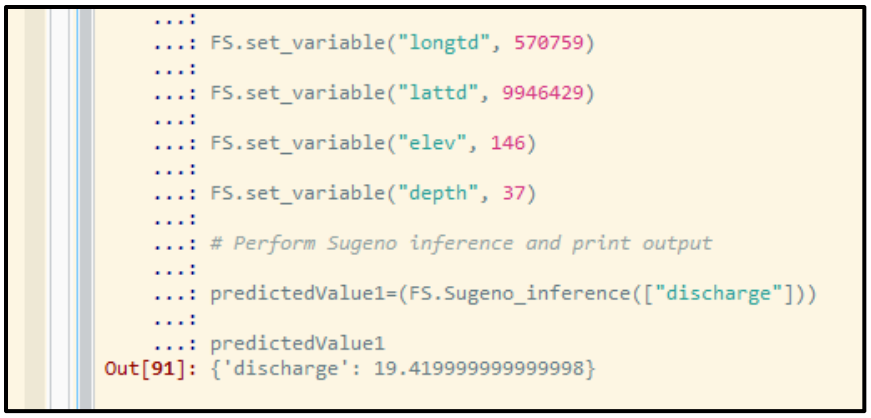


Figure 20: Consider a second case at Young Muslim secondary school, with the screen shot below, in a row whose expected discharge is 18.0 cubic meters per hour, and which the model predicts as 19.4. This is registers an accuracy of 93.0 percent, using the Sugeno engine.

A table of Original Discharges Vs Predicted Discharge values have been prepared for the Tana Alluvial aquifer wells. Replacement wells are being proposed very close to the wells appearing in this table, and whose longitudes and latitudes would approximately map onto those of the existing wells.

TABLE 3: TABLE OF ORIGINAL DISCHARGE VS PREDICTED DISCHARGE

Item	**Name**	**Original discharge value**	**Predicted discharge value**	**Accuracy=Smaller value/larger value**
594726 9914101	Abaqdera	12	12.8	94.0 %
570759 9946429	Young Muslim secondary	18	19.4	93.0%
614049 9863779	Hawajod	14	14.4	97.0%
536648 9983887	Saka centre	28	28.3	99.0%
598208 9903426	Nanighi	10	9.22	92.2%
600269 9888145	jambele	8	8.6	93.0%
580795 9959435	Umma University, Modika	16	16.6	96.0%
591067 9911741	Bawama	12	12.8	94.0%
569918 9948939	Mororo	20	18.44	92.0%
570250 9953979	Maalims Hardware ADC	20	19.5	98.9%

The ten examples are deemed suffiecnt in illustrating the power of fuzzy logic in aquifer discharge predictions within the Lower tana catchment.

```
    ...:
    ...: FS.set_variable("longtd", 614049)
    ...:
    ...: FS.set_variable("lattd", 9863779)
    ...:
    ...: FS.set_variable("elev", 83)
    ...:
    ...: FS.set_variable("depth", 175)
    ...:
    ...: # Perform Sugeno inference and print output
    ...:
    ...: predictedValue1=(FS.Sugeno_inference(["discharge"]))
    ...:
    ...: predictedValue1
Out[95]: {'discharge': 14.408949999999999}

In [96]:
    ...: 14/14.4
Out[96]: 0.9722222222222222
```

Figure 21: shows the prediction for table row number three at Hawajod posting accuracy value of 97% . Note that the true discharge is 14.0 whereas the computed / predicted discharge is 14.4 using neuro fuzzy inference using the Sugeno engine.

A. Decision Tree Modeling of Water Quality

Various categories of water quality were found in the existing wells and the classes included:

a) Freshwater

b) Hardwater

c) Saline water

d) Saline to brackish water.

Decision Tree models were then built to help preeict the water quality of the study area mapped using the python libraries .

Screenshot of the data used in mapping the water quality using Decision Tree models

	A	B	C	D	E	F	G	H
1	longtd	latittd	elev	depth	discharge	quality		
2	567187	9947947	161.0214	86.03062	14.07984	fresh		
3	571605	9949780	167.0514	96.06999	12.0428	saline		
4	571147	9948131	151.0097	54.05053	19.05123	fresh		
5	571147	9948131	151.0097	54.05053	19.05123	fresh		
6	577433	9936536	143.041	51.01463	28.04502	saline		
7	550849.1	9971172	171.0557	30.004	14.07512	fresh		
8	552755	9969873	167.0791	28.04583	13.065	fresh		
9	584309.1	9960565	244.0385	260.0309	22.04652	saline to brackish		
10	566708.1	9962346	160.0004	134.0739	21.02566	saline to brackish		
11	557678	9965170	167.0039	34.0774	12.00211	fresh		
12	551388	9964203	161.0632	35.06911	14.0085	fresh		
13	566842	9959715	160.0192	87.02855	20.00593	fresh		
14	536655.1	9983895	188.0361	57.06488	38.03129	fresh		
15	536512	9983932	188.058	55.06104	34.01243	fresh		
16	562096.1	9966525	163.0148	64.05429	25.01558	fresh		
17	578958	9937509	168.0657	65.00785	26.03284	fresh		

Data was partitioned for training and testing so that 75 percent of the dataset was sued in training and the rest used to test model accuracy, with the result that the DT model was 100 % accurate , as oposed to the kNN algorithm, which gave an accuracy of 94.1 % with that same dataset.

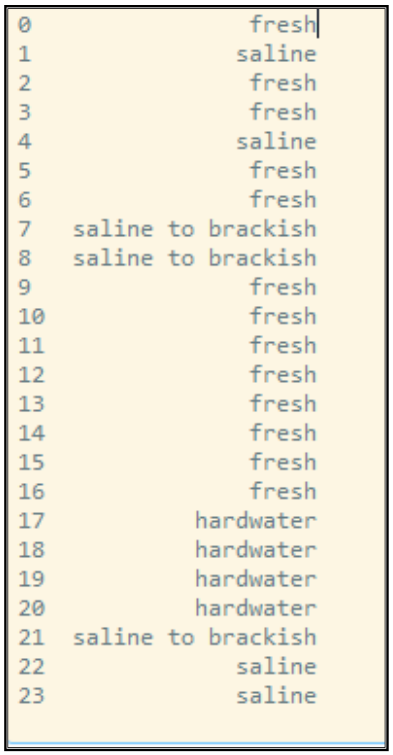
```
111
112    from sklearn.tree import DecisionTreeClassifier
113    clf = DecisionTreeClassifier(random_state=0)
114    modelDT=clf.fit(X, y)
115    modelDT
```

Figure 22: The algorithm used with the anaconda GUI in python. See the original test data and class of water quality hereunder:

	A	B	C	D	E	F
1	longtd	latittd	elev	depth	discharge	quality
2	567187	9947947	161.0214	86.03062	14.07984	fresh
3	571605	9949780	167.0514	96.06999	12.0428	saline
4	571147	9948131	151.0097	54.05053	19.05123	fresh
5	571147	9948131	151.0097	54.05053	19.05123	fresh
6	577433	9936536	143.041	51.01463	28.04502	saline
7	550849.1	9971172	171.0557	30.004	14.07512	fresh
8	552755	9969873	167.0791	28.04583	13.065	fresh
9	584309.1	9960565	244.0385	260.0309	22.04652	saline to brackish
10	566708.1	9962346	160.0004	134.0739	21.02566	saline to brackish
11	557678	9965170	167.0039	34.0774	12.00211	fresh
12	551388	9964203	161.0632	35.06911	14.0085	fresh
13	566842	9959715	160.0192	87.02855	20.00593	fresh
14	536655.1	9983895	188.0361	57.06488	38.03129	fresh
15	536512	9983932	188.058	55.06104	34.01243	fresh
16	562096.1	9966525	163.0148	64.05429	25.01558	fresh
17	578958	9937509	168.0657	65.00785	26.02284	fresh
18	570766	9946437	155.0487	46.02133	26.04506	fresh
19	616370	9865843	86.0141	146.0332	24.06782	hardwater
20	616370	9865843	86.0141	146.0332	24.06782	hardwater
21	614057.1	9863789	94.05058	186.0576	21.04192	hardwater
22	591251	9925348	134.025	206.0735	22.05299	hardwater
23	573422	9950108	172.042	213.0488	37.04768	saline to brackish
24	598216	9903436	123.0455	182.0625	20.07535	saline
25	594734	9914112	140.0162	175.0715	20.00026	saline
26						
27						

Figure 23: Data screenshot of test data of the Tana Alluvial Aquifer.

Compare this with the predicted class in python using the DT algorithm. The predicted classes match 100 percent in similarity to morignal test data used.



0	fresh
1	saline
2	fresh
3	fresh
4	saline
5	fresh
6	fresh
7	saline to brackish
8	saline to brackish
9	fresh
10	fresh
11	fresh
12	fresh
13	fresh
14	fresh
15	fresh
16	fresh
17	hardwater
18	hardwater
19	hardwater
20	hardwater
21	saline to brackish
22	saline
23	saline

Figure 24: The predicted class using the Decision Tree algorithm in python to map the Tana Alluvial aquifer water Quality Class.

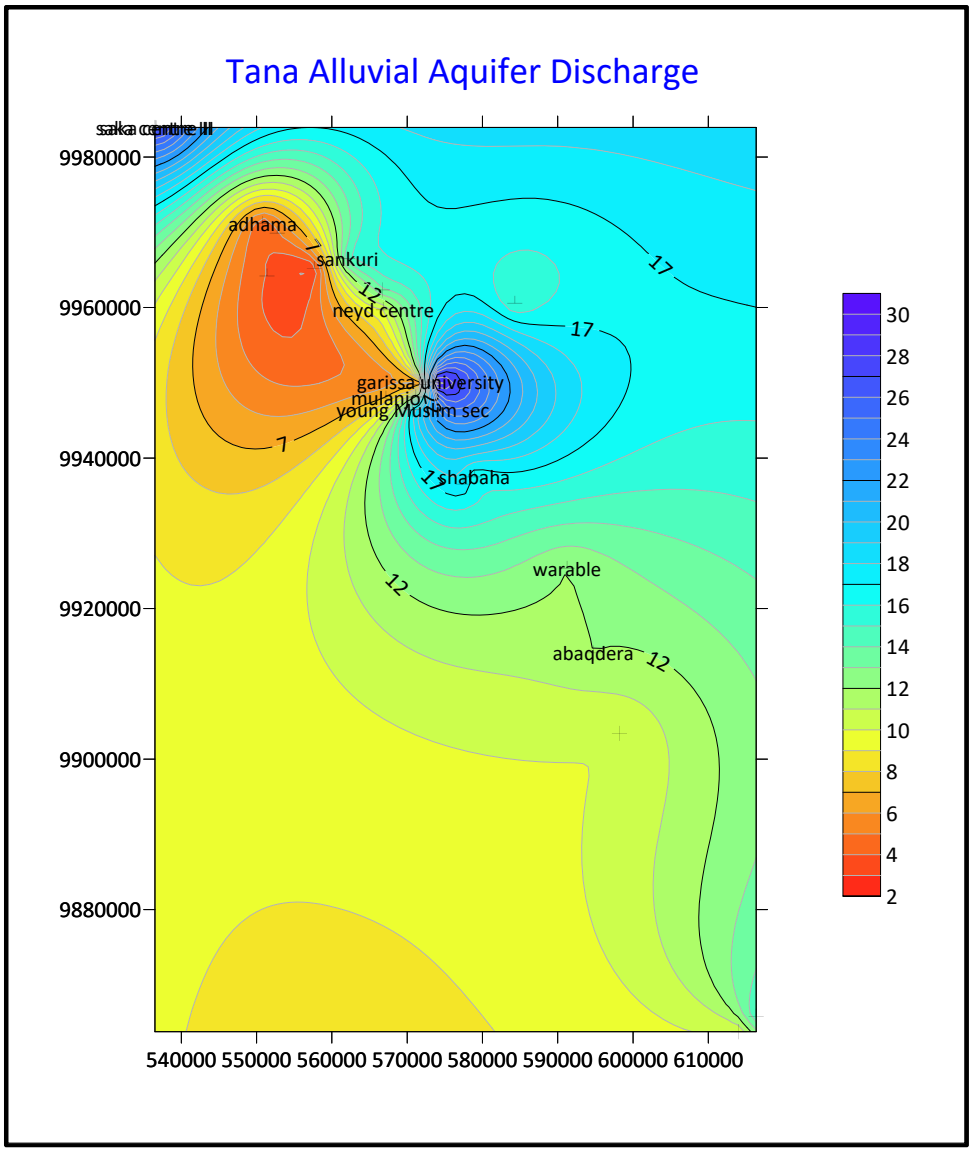


Figure 25: Discharge along the Tana Alluvial Aquifer

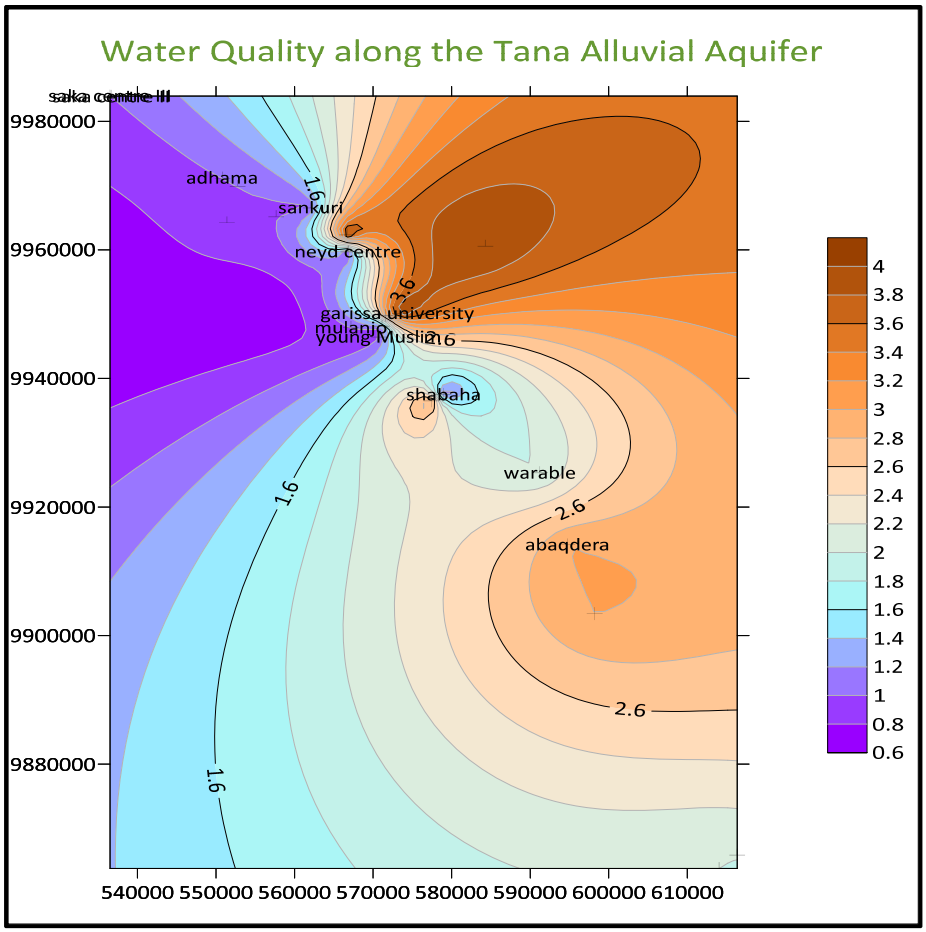


Figure 26: Water Quality along the Tana Alluvial Aquifer

VII. Recommendations And Conclusions

The importance of this study has been three-fold:

i) The study involving neuro-fuzzy assesment has established the efficacy of the use of neuro-fuzzy inference in the mapping of the uncertainity relating to aquifer discharge within the lithologic subsets/geozones, located within the Tana Alluvial Aquifer. In the event of a donor organization wishing to fund a water supply project for the local community or for local schools, this information shall be of immense help. The donor and the community will be able to tell, well in advance , the kind of discharge they would be anticipating, amidst the uncertainity, and this will aid both budget and planning.

ii) The study employing the use of Decision Tree has been proven a useful tool for mapping aquifer water quality class in the study area.This way, any spot earmarked for groundwater development may be known before-hand, as to the kind of water quality that would be coming out of it. This is a useful piece of information as relates to the planning of the Water Resources Planning & Development, regarding the use the water will be put to.

iii) Areas that were previously having fewer wells but presently having too many wells may be identified and targetted for de-crowding of the well fields, by using the new predictions to sink newer wells elsewhere, so as to guard against depletion of the TAA suites.

REFERENCES

[1] Aldino, A. A., & Sulistiani, H. (2020). Decision Tree C4. 5 Algorithm For Tuition Aid Grant Program Classification (Case Study: Department Of Information System, Universitas Teknokrat Indonesia). *Jurnal Ilmiah Edutic: Pendidikan dan Informatika*, *7*(1), 40-50.

[2] Alsalman, Y. S., Halemah, N. K. A., AlNagi, E. S., & Salameh, W. (2019, June). Using decision tree and artificial neural network to predict students academic performance. In *2019 10th International Conference on Information and Communication Systems (ICICS)* (pp. 104-109). IEEE.

[3] Azam, M. H., Hasan, M. H., Hassan, S., & Abdulkadir, S. J. (2020, October). Fuzzy type-1 triangular membership function approximation using fuzzy C-means. In *2020 International Conference on Computational Intelligence (ICCI)* (pp. 115-120). IEEE.

[4] Babanezhad, M., Masoumian, A., Nakhjiri, A. T., Marjani, A., & Shirazian, S. (2020). Influence of number of membership functions on prediction of membrane systems using adaptive network based fuzzy inference system (ANFIS). *Scientific Reports*, *10*(1), 1-20.

[5] Bardossy, A., Bogardi, I., & Duckstein, L. (1990). Fuzzy regression in hydrology. *Water Resources Research*, *26*(7), 1497-1508.

[6] Běhounek, L., & Cintula, P. (2006). From fuzzy logic to fuzzy mathematics: A methodological manifesto. *Fuzzy Sets and Systems*, *157*(5), 642-646.

[7] Bělohlávek, R., Dauben, J. W., & Klir, G. J. (2017). *Fuzzy logic and mathematics: a historical perspective*. Oxford University Press.

[8] Chen, C. S., Jhong, Y. D., Wu, W. Z., & Chen, S. T. (2019). Fuzzy time series for real-time flood forecasting. *Stochastic Environmental Research and Risk Assessment*, *33*(3), 645-656.

[9] Dombi, J., & Jónás, T. (2020). Ranking trapezoidal fuzzy numbers using a parametric relation pair. *Fuzzy sets and systems*, *399*, 20-43.

[10] Dou, J., Yunus, A. P., Bui, D. T., Merghadi, A., Sahana, M., Zhu, Z., ... & Pham, B. T. (2019). Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan. *Science of the total environment*, *662*, 332-346.

[11] Du, X., Xu, H., & Zhu, F. (2021). A data mining method for structure design with uncertainty in design variables. *Computers & Structures*, *244*, 106457.

[12] Gentili, P. L., Giubila, M. S., & Heron, B. M. (2017). Processing binary and fuzzy logic by chaotic time series generated by a hydrodynamic photochemical oscillator. *ChemPhysChem*, *18*(13), 1831-1841.

[13] Ghosh, A., & Maiti, R. (2021). Soil erosion susceptibility assessment using logistic regression, decision tree and random forest: study on the Mayurakshi river basin of Eastern India. *Environmental Earth Sciences*, *80*(8), 1-16.

[14] Gonoodi, K., Tayefi, M., Saberi-Karimian, M., Darroudi, S., Farahmand, S. K., Abasalti, Z., ... & Mobarhan, M. G. (2019). An assessment of the risk factors for vitamin D deficiency using a decision tree model. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, *13*(3), 1773-1777.

[15] Jiang, J., Cui, B., Zhang, C., & Fu, F. (2018, May). Dimboost: Boosting gradient boosting decision tree to higher dimensions. In *Proceedings of the 2018 International Conference on Management of Data* (pp. 1363-1376).

[16] Lan, T., Hu, H., Jiang, C., Yang, G., & Zhao, Z. (2020). A comparative study of decision tree, random forest, and convolutional neural network for spread-F identification. *Advances in Space Research*, *65*(8), 2052-2061.

[17] Lange, H., & Sippel, S. (2020). Machine learning applications in hydrology. In *Forest-water interactions* (pp. 233-257). Springer, Cham.

[18] Mohebbi Tafreshi, G., Nakhaei, M., & Lak, R. (2021). Land subsidence risk assessment using GIS fuzzy logic spatial modeling in Varamin aquifer, Iran. *GeoJournal*, *86*(3), 1203-1223.

[19] Moorthi, P. V. P., Singh, A. P., & Agnivesh, P. (2018). Regulation of water resources systems using fuzzy logic: a case study of Amaravathi dam. *Applied Water Science*, *8*(5), 1-11.

[20] Nguyen, P. T., Ha, D. H., Nguyen, H. D., Van Phong, T., Trinh, P. T., Al-Ansari, N., ... & Prakash, I. (2020). Improvement of credal decision trees using ensemble frameworks for groundwater potential modeling. *Sustainability*, *12*(7), 2622.

[21] Nourani, V., Razzaghzadeh, Z., Baghanam, A. H., & Molajou, A. (2019). ANN-based statistical downscaling of climatic parameters using decision tree predictor screening method. *Theoretical and Applied Climatology*, *137*(3), 1729-1746.

[22] Pourabdollah, A., Mendel, J. M., & John, R. I. (2020). Alpha-cut representation used for defuzzification in rule-based systems. *Fuzzy Sets and Systems*, *399*, 110-132.

[23] Purba, R. A., Samsir, S., Siddik, M., Sondang, S., & Nasir, M. F. (2020, April). The optimalization of backpropagation neural networks to simplify decision making. In *IOP Conference Series: Materials Science and Engineering* (Vol. 830, No. 2, p. 022091). IOP Publishing.

[24] Rezaei, F., Safavi, H. R., & Ahmadi, A. (2013). Groundwater vulnerability assessment using fuzzy logic: a case study in the Zayandehrood aquifers, Iran. *Environmental management*, *51*(1), 267-277.

[25] Roy, S., Mondal, S., Ekbal, A., & Desarkar, M. S. (2019). Dispersion ratio based decision tree model for classification. *Expert Systems with Applications*, *116*, 1-9.

[26] Sarda, K., Yerudkar, A., & Vecchio, C. D. (2020). Disturbance decoupling control design for Boolean control networks: a Boolean algebra approach. *IET Control Theory & Applications*, *14*(16), 2339-2347.

[27] Tangirala, S. (2020). Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm. *International Journal of Advanced Computer Science and Applications*, *11*(2), 612-619.

[28] Tzimopoulos, C., Papadopoulos, K., & Papadopoulosc, B. (2016). Fuzzy regression with applications in hydrology. *optimization*, *5*(8).

[29] Wang, D., Li, Y., Wang, L., & Gong, B. (2020). Neural networks are more productive teachers than human raters: Active mixup for data-efficient knowledge distillation from a blackbox model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1498-1507).

[30] Wang, Y., & Kong, T. (2019). Air quality predictive modeling based on an improved decision tree in a weather-smart grid. *IEEE Access*, *7*, 172892-172901.