**Artificial and Computational Intelligence**

**Assignment-2 – Water Quality Classification**

**Prepared and executed by**

|  |  |
| --- | --- |
| 2020FC04101 | AMAL SHAJI |
| 2020FC04102 | GNANESWAR S V VIJJAPU |
| 2020FC04103 | UMANG MISHRA |
| 2020FC04063 | DHAVALKUMAR KOTECHA |
| 2020FC04531 | JANAKIRAMAN H |

**TABLE OF CONTENTS**

[1 Scope, Purpose, Audience 3](#_Toc82954253)

[2 Problem Statement 3](#_Toc82954254)

[3 Approaches and Design 3](#_Toc82954255)

[3.1 Data Preprocessing 3](#_Toc82954256)

[3.2 Discretization of numerical data 4](#_Toc82954258)

[3.3 Parameter learning and Network structure 7](#_Toc82954261)

[3.4 Model fitting and Generating Inferences 9](#_Toc82954262)

[4 Q & A -- Python 11](#_Toc82954263)

[5 Q & A -- Prolog 14](#_Toc82954264)

[6 Testing Summary 18](#_Toc82954265)

[7 Assumptions & Considerations 18](#_Toc82954266)

# 

# Scope, Purpose, Audience

The purpose of this document is to provide the information needed to understand the approach, design solution and algorithm that were considered while solving the Assignment 2 –Water Quality Classification for Artificial and Computational Intelligence. The primary audience of this document is the members of BITS Pilani WILP who evaluate and assess this problem statement.

# Problem Statement

Water quality data is provided along with the file and contains the following attributes

Attributes Information:

1. ph: pH of 1. water (0 to 14).

2. Hardness: Capacity of water to precipitate soap in mg/L.

3. Solids: Total dissolved solids in ppm.

4. Chloramines: Amount of Chloramines in ppm.

5. Sulfate: Amount of Sulfates dissolved in mg/L.

6. Conductivity: Electrical conductivity of water in μS/cm.

7. Organic\_carbon: Amount of organic carbon in ppm.

8. Trihalomethanes: Amount of Trihalomethanes in μg/L.

9. Turbidity: Measure of light emitting property of water in NTU.

10. Potability: Indicates if water is safe for human consumption. Potable -1 and Not potable -0

# Approaches and Design

## Data Preprocessing

The first step that was included as part of pre-processing the water quality classification data was importing the necessary modules for the notebook. Below is the list of python libraries that are required:

1. Pandas
2. Numpy
3. Pylab
4. Time
5. warnings
6. Pgmpy – to create graphical models
7. feature\_engine.discretisation – to create feature engineering discretization transformers.
8. NetworkX - for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.

Using pandas read\_csv, the dataframe is created with the shape – (3276,10).

## Applying Linear Interpolation for Missing Values:

* Linear Interpolation simply means to estimate a missing value by connecting dots in a straight line in increasing order. In short, it estimates the unknown value in the same increasing order from previous values.
* The linear method ignores the index and treats missing values as equally spaced and finds the best point to fit the missing value after previous points.

|  |  |
| --- | --- |
| **Given Dataset – Missing Values** | **Linear Interpolation – Forwarding Direction** |
|  |  |

As we have one record which is missing value for ph, we further decided to drop this from the dataset for future processing.

## Discretization of numerical data

## Bayesian Network

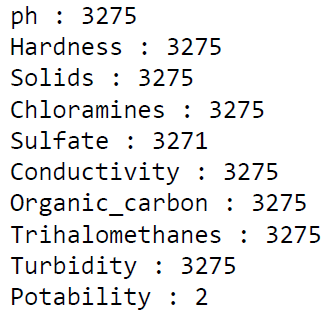
A Bayesian network (BN) is a probabilistic graphical model for representing knowledge about an uncertain domain where each node corresponds to a random variable and each edge represents the conditional probability for the corresponding random variables

**Reason for Discretization/Binning**

We are using PGMPY library for the creation of Bayesian networks. Bayesian Networks does not support continuous data hence we need to make it as categorical. Discretization is the method of doing the above process.

As a general standard we will check which numerical features need to be binned, the feature which has more than 32 unique values should be binned.

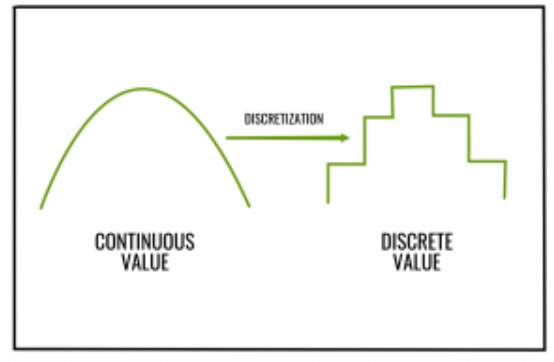
**Uniqueness:**



As we can see that, other than Potability every column is having more than 32 unique values hence we need to discretize these columns.

## Discretization

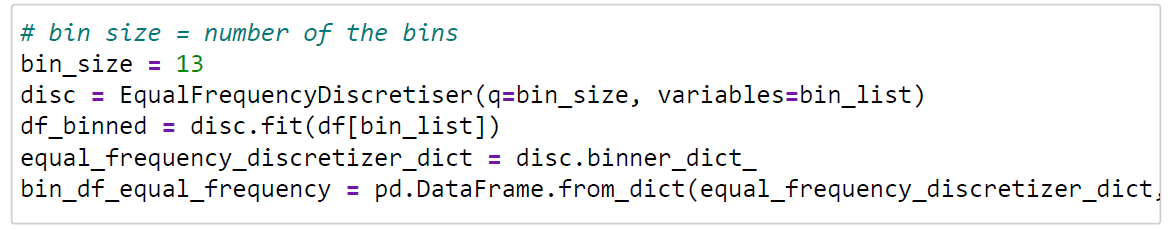
Discretization is the process of transforming numeric variables into nominal variables called bin. The created variables are nominal but are ordered (which is a concept that you will not find in true nominal variable) and algorithms can exploit this ordering information.



**EqualFrequencyDiscretiser Method:**

In equal-frequency binning we divide the range [A, B] of the variable into intervals that contain (approximately) equal number of points; equal frequency may not be possible due to repeated values.

We have prepared the list of features (bin\_list) which we will bin using EqualFrequencyDiscretiser.EqualFrequencyDiscretiser: It divides continuous numerical variables into intervals that containapproximately the same proportion of observations from feature\_engine library.



**We have chosen bin size \*\*13\*\* considering the computational power of PC. It may affect the accuracy, but always the higher the better**.

As discretization is being done, in order to increase the meaning and easy querying of dataset we need to rename each value in each column corresponding to the range of bin it represents as below.

|  |  |
| --- | --- |
| **After Discretization** | **After Bin Ranging** |
|  |  |

## Parameter learning and Network structure

Different types of parameter learning and tree generation techniques:

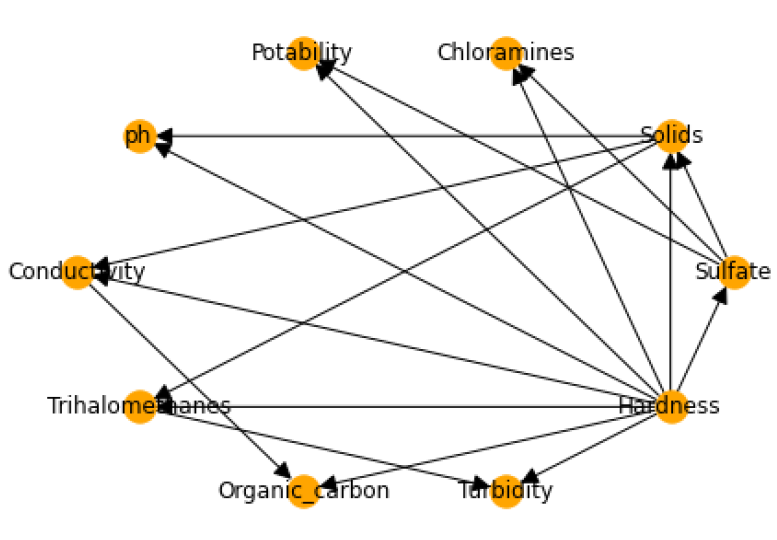
Hill climb search

Tree search

Mmhc Estimator

Exhaustive search

We tried all of the algorithms and found that Tree Search algorithm with the estimator type tan isgiving the maximum accuracy.Hence we are fixing that algorithm for our network structure generation

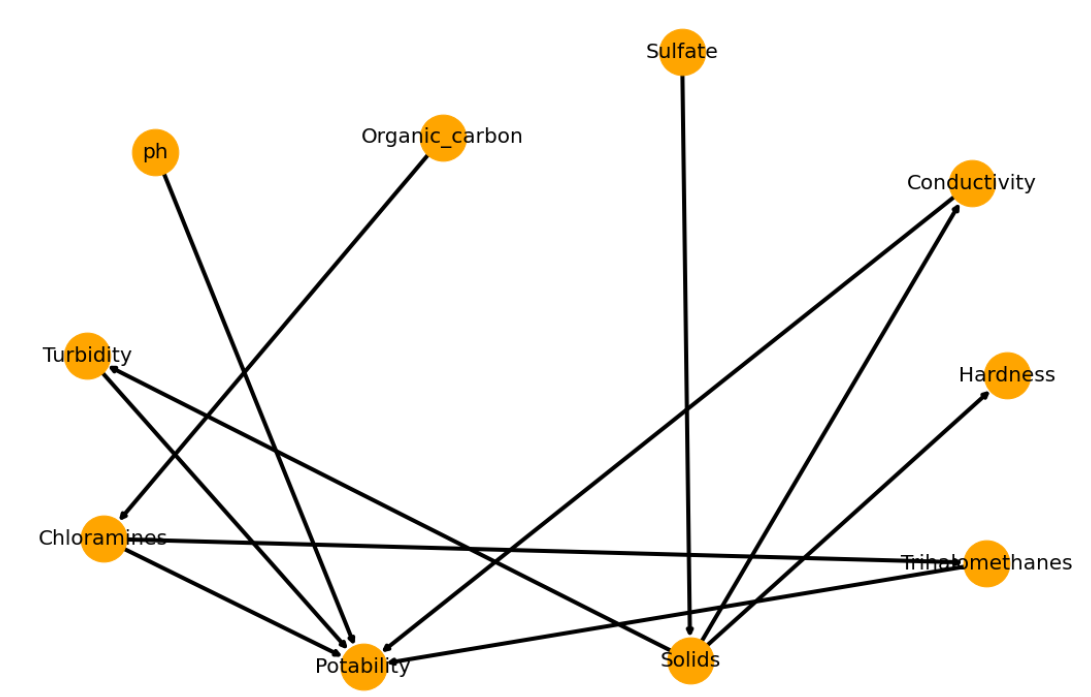


**Observation:**

This network structure was giving very low accuracy on our test set. Hence, we had to construct the Bayesian network ourselves.

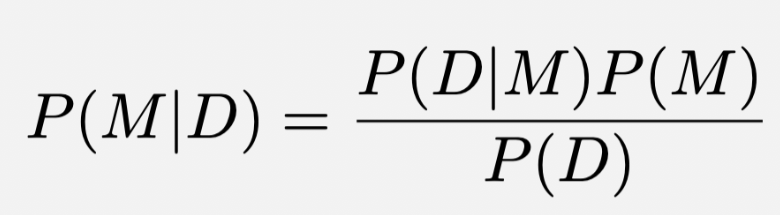
After doing a case study on the parameters affecting the water quality (detailed explanation is mentioned in the notebook), we came at the below connections between the factors.

1. Ph → potability
2. Solids → hardness
3. Solids → potability
4. Chloramines → potability
5. Sulphate→ Solids
6. Solids → Conductivity
7. Organic\_carbon → Potability
8. Organic\_carbon → Chloramines
9. Derived Tree Structure after doing the case study –



## Model fitting and Generating Inferences

Bayesian model selection can be applied to situations where we have multiple competing models and need to select the best model. According to the Bayes’s theorem, any model’s posterior probability can be written as,



Here, P(M|D) is the posterior probability of model M given the data D, P(D|M) is the evidence for the model M, P(M) is the prior knowledge about the model M, and P(D) is normalization factor.

**Variable Elimination**

Variable elimination is a standard algorithm for computing probability of evidence with respect to a given a Bayesian network. First, the algorithm acts on a set of factors. Each factor involves a set of variables and maps instantiations of those variables to real–numbers. The initial set of factors are the network’s conditional probability distributions

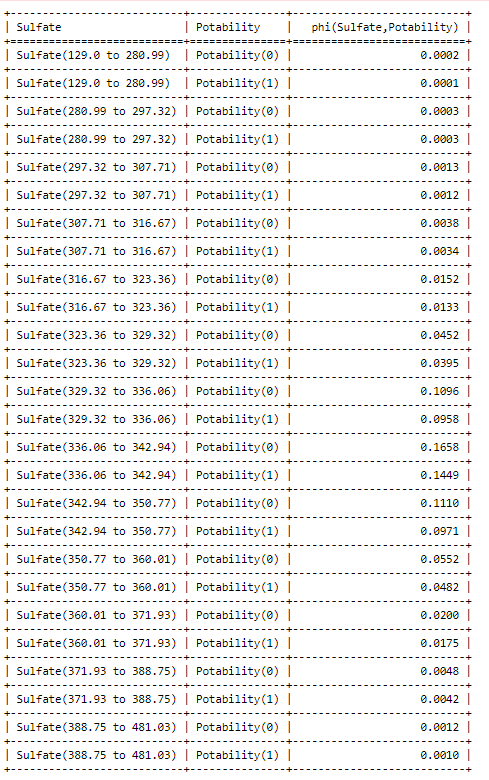
Elimination is driven by an ordering on the variables called an elimination order. During the algorithm, two factor operations are performed many times: factors are multiplied and a variable is summed out of a factor. These factor operations reduce to performing many multiplication and addition operations on real–numbers.

**Joint Distribution**

Here we are considering two factors and applying joint distribution on Sulphate and Potability using variable elimination.



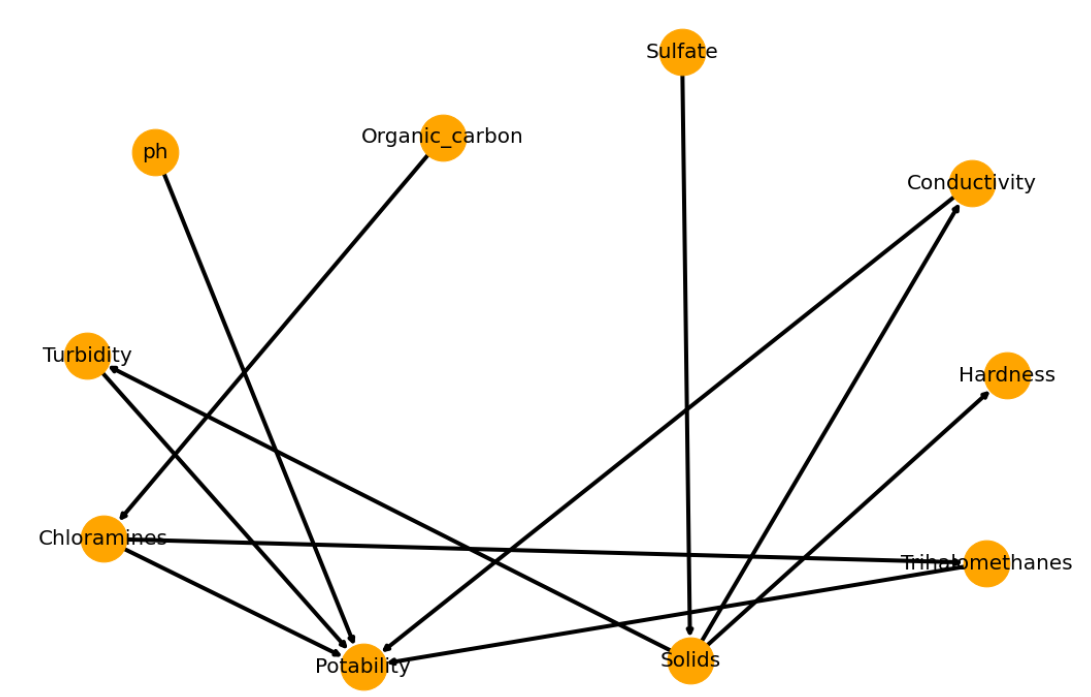




# Q & A -- Python

* 1. *Construct a Bayesian Belief Network for the given data*

As per section 3.3, we came at the below connections between the factors for Bayesian Network.



* 1. *Predict the water quality for the following data*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *ph* | *Hardness* | *Solids* | *Chloramines* | *Sulfate* | *Conductivity* | *Turbidity* |
| *3.72* | *204.89* | *20791.32* | *7.3* | *368.5* | *564.30* | *2.96* |

**Step 1:** We considered each variable is a function which takes an input value and return the interval according to the bin range.

Below table is the representation of the selection of bin for the given variable.

|  |  |
| --- | --- |
| **Given Ph variable** | **Mapped bin range** |
|  |  |

**Step 2:** Using Variable Elimination method, we query Potability versus the other evidence attributes to predict the water quality for the given data and retrieve the potability using map\_query from the Variable Elimination. As per the results in notebook, we observe the portability is 0 and hence water is non drinkable for these given attributes.

* 1. *Infer the probability for the data with the following properties:*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ph | Hardness | Solids | Chloramines | Sulfate | Conductivity | Organic carbon | Trihalomethanes | Turbidity | Potability |
| 10. | 248. | 28749 | 7.5 | 393 | 283 | 13.78 | 84.6 | 2.67 | 1 |

**Step 1:** We considered each variable is a function which takes an input value and return the interval according to the bin range.

Below table is the representation of the selection of bin for the given variable.

|  |  |
| --- | --- |
| **Given Ph variable** | **Mapped bin range** |
|  |  |

As we are asked to find the probability of occurring the above condition which can be done using Chain Rule.

**Step 2:** The chain rule, also called as general product rule, calculates any component of the joint distribution of a set of random variables using only conditional probabilities. This probability theory is used as a foundation for backpropagation and in creating Bayesian networks.

We need to find each of these values using infer method of variable elimination and derive terms which explained in detail in the notebook.

This simple chain of probability and random variables is expressed as:

P(A,B) = P(B | A) P(A)

Thus, according the chain rule –

P(ph,har,sol,chl,sul,con,tur,org,tri,pot)= P(pot|tri,org,tur,con,sul,chl,sol,har,ph)\* P(tri|org,tur,con,sul,chl,sol,har,ph)\* P(org|tur,con,sul,chl,sol,har,ph) \* P(tur|con,sul,chl,sol,har,ph) \* P(con|sul,chl,sol,har,ph) \* P(sul|chl,sol,har,ph) \* P(chl|sol,har,ph) \* P(sol|har,ph) \* P(har|ph) \* P(ph)

We need to find each of these values using infer method of variable elimination and derive terms which explained in detail in the notebook.

* P(pot|tri,org,tur,con,sul,chl,sol,har,ph)🡺 term1 = 0.5
* P(tri|org,tur,con,sul,chl,sol,har,ph)🡺 term2 = 0.01287
* P(org|tur,con,sul,chl,sol,har,ph) 🡺 term3 = 0.1724
* P(tur|con,sul,chl,sol,har,ph)🡺 term4 = 0.0
* P(con|sul,chl,sol,har,ph) 🡺 term5 = 0.0
* P(sul|chl,sol,har,ph) 🡺 term6 = 0.0
* P(chl|sol,har,ph) 🡺 term7 = 0.24281
* P(sol|har,ph) 🡺 term8 = 0.0
* P(har|ph) 🡺 term9 = 0.0003125
* P(ph)🡺 term10 = 0.00125
* req\_prob = (term1 \* term2 \* term3 \* term4 \* term5 \* term6 \* term7 \* term8 \* term9 \* term10)
* req\_prob= 0.0

Since the derived probability is 0, the chances and the likelihood of having such combination of records is very less and so small as it to be not worth considering.

* 1. *Find the probability of the quality of water being good and the attributes take the following values: low ph, high in hardness, with high presence of solids, and other chemicals.*

Here, for answering the question, we will only consider the following attributes/features:

ph

Hardness

Solids

Chloramines

Sulfate

Trihalomethanes

Conductivity, Organic\_carbon, and Turbidity are not checmicals, hence ignoring them.

Here considering LOW value and HIGH value is a decision-making-challenge, as we don't have the threshold values available.

So, we will be using the quartiles to make fine distinctions for LOW and HIGH values.

LOW value <= (range/4), (lets denote it with q1)

HIGH value >= ((range/4)\*3), (lets denote it with q3)

1. For ph the q1 is 3.5, so anything which is less than or equal to q1, will be considered as low value. And this low value for ph comes within the '0.0 to 4.94' interval, hence taking that interval only.
2. F or Solids the q3 is 45679.69, so anything which is more than or equal to q3, will be considered as high value. And this high value for Solids comes within the intervals, '35297.61 to 61227.2', hence taking this intervals only.
3. For Chloramines the q3 is 9.585, so anything which is more than or equal to q3, will be considered as high value. And this high value for Chloramines comes within the intervals, '8.66 to 9.38', '9.38 to 13.13', hence taking these intervals only.
4. For Sulfate the q3 is 264.02, so anything which is more than or equal to q3, will be considered as high value. And this high value for Sulfate comes within the intervals, '129.0 to 280.99', '280.99 to 297.32', '297.32 to 307.71', '307.71 to 316.67', '316.67 to 323.36', '323.36 to 329.32', '329.32 to 336.06', '336.06 to 342.94', '342.94 to 350.77', '350.77 to 360.01', '360.01 to 371.93', '371.93 to 388.75', '388.75 to 481.03', hence taking all these intervals only.
5. For Trihalomethanes the q3 is 92.445, so anything which is more than or equal to q3, will be considered as high value. And this high value for Trihalomethanes comes within the intervals, '88.69 to 124.0', hence taking this intervals only.

**Conclusion:**

For low value of ph, the likelihood of water being drinkable is 0.4999

For high value of Hardness, the likelihood of water being drinkable is 0.4665

For high value of Solids, the likelihood of water being drinkable is 0.4521

For high value of Chloramines, the likelihood of water being drinkable is 0.4967

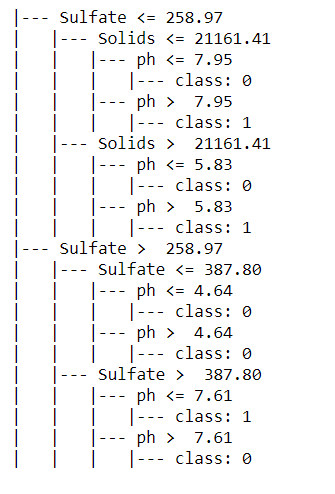
For high value of Sulfate, the likelihood of water being drinkable is 0.4710

For high value of Trihalomethanes, the likelihood of water being drinkable is 0.5010

# Q & A -- Prolog

* 1. *Use Any of the decision tree algorithms to build a decision tree for the given data*

Using DecisionTreeClassifier below is the tree representation that are being derived.



* 1. *Create rules from the decision tree*

Rules derived from the above decision tree algorithm

1. if (Sulfate > 258.97) and (Sulfate <= 387.796) and (ph > 4.636) then class: Non drinkable (proba: 62.19%) | based on 2,809 samples
2. if (Sulfate > 258.97) and (Sulfate > 387.796) and (ph <= 7.61) then class: Drinkable (proba: 66.04%) | based on 159 samples
3. if (Sulfate > 258.97) and (Sulfate <= 387.796) and (ph <= 4.636) then class: Non drinkable (proba: 79.45%) | based on 146 samples
4. if (Sulfate > 258.97) and (Sulfate > 387.796) and (ph > 7.61) then class: Non drinkable (proba: 78.08%) | based on 73 samples
5. if (Sulfate <= 258.97) and (Solids > 21161.408) and (ph > 5.825) then class: Drinkable (proba: 94.34%) | based on 53 samples
6. if (Sulfate <= 258.97) and (Solids <= 21161.408) and (ph <= 7.947) then class: Non drinkable (proba: 73.91%) | based on 23 samples
7. if (Sulfate <= 258.97) and (Solids > 21161.408) and (ph <= 5.825) then class: Non drinkable (proba: 50.0%) | based on 8 samples
8. if (Sulfate <= 258.97) and (Solids <= 21161.408) and (ph > 7.947) then class: Drinkable (proba: 100.0%) | based on 5 samples
   1. *Code the rules into a Prolog Knowledge base*

Below are the rules loaded into the Prolog Knowledge Base.

start :-

write('Write the value of PH: '),

read(Ph),nl,

write('Write the value of Solids: '),

read(Sol),nl,

write('Write the value of Sulphate: '),

read(Sul),nl,

input(Ph,Sol,Sul).

input(stop) :- !.

input(Ph,Sol,Sul):-

Ph > 4.636,

Sul =< 387.796,

Sul > 258.97,

write("The Results are"),nl,nl,

write(" Non Drinkalble"),nl,

write(" proba: 62.19%"),nl.

input(Ph,Sol,Sul):-

Ph =< 7.61,

Sul > 387.796,

write("The Results are"),nl,nl,

write("Drinkalble"),nl,

write(" proba: 66.04%"),nl.

input(Ph,Sol,Sul):-

Ph =< 4.636,

Sul =< 387.796,

Sul > 258.97,

write("The Results are"),nl,nl,

write("Non Drinkalble"),nl,

write(" proba: 79.45%"),nl.

Input(Ph,Sol,Sul):-

Ph > 7.61,

Sul > 387.796,

write("The Results are"),nl,nl,

write("non Drinkalble"),nl,

write(" proba: 78.08%"),nl.

Input(Ph,Sol,Sul):-

Ph > 5.825,

Sul =< 258.97,

Sol > 21161.408,

write("The Results are"),nl,nl,

write("Drinkalble"),nl,

write(" proba: 94.34%"),nl.

input(Ph,Sol,Sul):-

Ph =< 7.947,

Sul =< 258.97,

Sol =< 21161.408,

write("The Results are"),nl,nl,

write("Non Drinkalble"),nl,

write(" proba: 73.91%"),nl.

input(Ph,Sol,Sul):-

Ph =< 5.825,

Sul =< 258.97,

Sol > 21161.408,

write("The Results are"),nl,nl,

write("Non Drinkalble"),nl,

write(" proba: 50.0%"),nl.

input(Ph,Sol,Sul):-

Ph > 7.947,

Sul =< 258.97,

Sol =< 21161.408,

write("The Results are"),nl,nl,

write("Drinkalble"),nl,

write(" proba: 100.0%"),nl.

* 1. *Get water properties as input from the user and classify whether water is potable or not*

|  |  |
| --- | --- |
| **Input Values with Drinkable** | **Input Values with Non-Drinkable** |
|  |  |

# Testing Summary



# Assumptions & Considerations

|  |  |
| --- | --- |
| **S.No** | **Detailed Description** |
| 1 | While doing the equal frequency discretiser, we tested the model ranging the bin size from 3 till 20 as larger bin size gives more accuracy. However, when we choose larger bin size the model is taking time for doing the evaluation due to computational power of local PC. Thus, we have chosen an optimal value – 13 for better performance. |