

Efficient Water Quality Analysis & Prediction using Machine Learning

A PROJECT REPORT

Submitted by

- | | |
|----------------------------------|---------------------|
| 1. Afra Iman. A | 410617106002 |
| 2. Fathima Koya Nilora. S | 410617106005 |
| 3. Hameetha Shajini. A | 410619106006 |
| 4. Farhana. K | 410617106004 |

In partial fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATIONS ENGINEERING



**DHAANISH AHMED COLLEGE OF ENGINEERING,
PADAPPAL, CHENNAI – 601301**



**ANNA UNIVERSITY: CHENNAI 600 025
2022-2023**



ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report “ Efficient Water Quality Analysis & Prediction using Machine Learning ” is the bonafide work of **Afra Iman .A (410619106002)** , who carried out the project work under my supervision.

SIGNATURE

Mr. A.Rajasekar M.E,(Ph.D).,
Head Of Department,
Project Mentor,
Electronics and communications
Engineering,
Dhaanish Ahmed College of
Engineering,
Padappai
Chennai – 601 301

SIGNATURE

Mr.M.Gandhi M.E.,
Assisant Professor,
Project Evaluator
Electronics and communications
Engineering,
Dhaanish Ahmed College of
Engineering,
Padappai
Chennai – 601 301

Project Viva-Voce held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

First and foremost, we thank the Almighty for helping us in all situations for bringing out this project successfully.

We express our sincere heartfelt gratitude to **ALHAJ K. MOOSA**, Founder and Chairman, and **Mr. M. KADAR SHAH, B.A., M.B.A.**, Secretary, **Dhaanish Ahmed College of Engineering, Chennai**.

We record our immense pleasure in expressing sincere gratitude to our Principal **Dr. Uma Gowri.G** and our Director **Dr.Paramasivan, Ph.d**, **Dhaanish Ahmed College of Engineering**, for granting permission to undertake the project in our college.

We express our sincere thanks to our Head of the Department **Mr.C.Elayaraja, M.E.,(Ph.D).**, **Dhaanish Ahmed College of Engineering**, and our project guide **Mr. Rajasekhar, M.E.**, Assistant professor, **Dhaanish Ahmed College of Engineering**, for their constant encouragement and direction for this project.

We wish to express our thanks to all the **Faculty Member and Non-teaching staff** of the Department of Electronics and Communications Engineering for their valuable support.

We also thank our Parents and Friends for their support throughout the project.

ABSTRACT

Water is considered as a vital resource that affects various aspects of human health and lives. The quality of water is a major concern for people living in urban areas. The quality of water serves as a powerful environmental determinant and a foundation for the prevention and control of waterborne diseases. However predicting the urban water quality is a challenging task since the water quality varies in urban spaces non-linearly and depends on multiple factors, such as meteorology, water usage patterns, and land uses, so this project aims at building a Machine Learning (ML) model to Predict Water Quality by considering all water quality standard indicators. This study investigates the performance of Machine learning techniques , group method of data handling (GMDH) for predicting water quality components located in Goa, India. To develop the accurate predictions many models were implemented and reviewing the results of models the Random Forest Regression have suitable performance for predicting water quality components. Comparison of outcomes of GMDH model with other applied models shows that although this model has acceptable performance for predicting the components of water quality, its accuracy is slightly less than Random forest regressor. The evaluation of the accuracy of the applied models according to the error indexes declared that SVM was the most less accurate model. Examining the results of the models showed that all of them had some over-estimation properties. By evaluating the results of the models based on the Water Quality index, it was found that the highest pH value was related to the conductivity of the water.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	ii
	TABLE OF CONTENTS	iii
1	INTRODUCTION	1
	1.1 Overview	
2	LITERATURE SURVEY	4
3	SYSTEM ANALYSIS	6
	3.1 Empathy Map	
	3.2 Brainstorm and ideation process	
	3.3. Problem Statement	
	3.4 Proposed Solution	
4	EXPERIMENTAL DESCRIPTIONS	10
	4.1 Technical Block Diagram	
5	PROJECT DESIGN PHASE	11
	5.1 Data flow diagram	

6	SOFTWARE DESCRIPTION	12
	6.1 Machine Learning	
	6.2 Regression Models in ML	
	6.3 Data Visualization in ML	
7	PROJECT PLANNING & SPRINT DELIVERIES	25
	7.1 Sprint schedule and estimation	
8	PROJECT DEVELOPMENT PHASE	27
	8.1 Source code	
9	WEBPAGE DEVELOPMENT	33
	9.1 Source code	
10	OUTPUT AND RESULTS	35
	10.1 User data Visualization	
	10.2 Model Final Prediction & Metrics	
	10.3 Webpage design Output	
11	CONCLUSION	38

1. INTRODUCTION

Water quality has a direct impact on public health and the environment. Water is used for various practices, such as drinking, agriculture, and industry. Recently, development of water sports and entertainment has greatly helped to attract tourists (Jennings 2007). Among various sources of water supply, due to easy access, rivers have been used more frequently for the development of human societies. Using other water resources such as groundwater and seawater sometimes assisted with problems. For example, using groundwater without suitable recharge will lead to land subsidence (Motagh *et al.* 2017) and using seawater is usually associated with pollution transmission (El-Kowrany *et al.* 2016). Therefore, the use of rivers has attracted attention. Several investigations related to rivers around the world have been conducted and a field of engineering named river engineering has been proposed. In river engineering, studies on morphological changes, sediment transport, water quality, and pollution transmission mechanisms are very important (Julien 2002; Dey 2014). Flow structure, sediment transport and morphology of rivers are investigated in the hydraulics of rivers in river engineering (Wu 2007). The study of water quality of rivers is a common theme in earth sciences. To evaluate the quality of rivers two approaches are considered, including measuring the water quality components and defining the mechanism of pollution transmission (Kashefipour 2002; Kashefipour & Falconer 2002; Naseri Maleki & Kashefipour 2012; Qishlaqi *et al.* 2016). Among water quality components, measuring the dissolved oxygen (DO), chemical oxygen demand (COD), biochemical oxygen demand (BOD), electrical conductivity (EC), pH, temperature, K, Na, Mg, etc. have been proposed (Şener *et al.* 2017). To this end, governments have constructed

hydrometry stations along rivers that cross from urban areas, agro-industrial projects, industrial estates, and rivers that join dams' reservoirs ([Herschy 1993](#); [Kejiang 1993](#)). In hydrometry stations, the water quality components are measured and the stage-discharge relation is defined. Obtained values from hydrometry stations contain basic information for feasibility studies and development of water conservation projects. Evaluation of water quality is a basic stage for development of agriculture projects in terms of determination of cropping pattern, type of irrigation system, and systems of water purification for industries ([Chen *et al.* 2017](#)). To investigate the mechanism of pollution transmission, in addition to field and laboratory experiments, advanced numerical methods such as computational hydraulic, image processing and GIS methods have been utilized ([Parsaie & Haghiabi 2015, 2017a, 2017b](#)). By reviewing the time history of water quality components, investigators have attempted to estimate future values. Nowadays, by advancing soft computing techniques in most areas of water and environmental engineering, researchers have attempted to accurately analyse time series of water quality components and their internal relation ([May *et al.* 2008](#); [Palani *et al.* 2008](#); [Haghiabi 2016a, 2016b](#); [Jaddi & Abdullah 2017](#)). In this regard, [Emamgholizadeh *et al.* \(2013\)](#) used multilayer perceptron (MLP), radial basis network (RBF) and an adaptive neuro-fuzzy inference system (ANFIS) for water quality components of Karoon River. They stated that all applied models have suitable performance for prediction of water quality components; however, the MLP model was slightly more accurate. [Shokoohi *et al.* \(2017\)](#) managed the water quality of a water supply system. They considered this an optimization problem and used modern optimization methods to solve it. [Zhang *et al.* \(2010\)](#) introduced a new approach for water allocation. They

considered water quality as one of the main factors in their approach. [Nikoo & Mahjouri \(2013\)](#) developed a Probabilistic Support Vector Machines (PSVMs) model associated with GIS technique for planning the classification and distribution of surface and groundwater water in Iran. They stated that the use of these two methods would provide accurate information for feasibility studies of water conservation projects. [Heddami \(2016a; 2016b; 2016c; 2016d; 2016e\)](#) utilized artificial neural networks for predicting the water quality components in several case studies. He stated that artificial intelligence techniques have suitable performance for modeling and predicting the internal relation between the water quality components and modeling their time series. Reviewing the literature shows that water quality assessment and prediction is an important factor for developing water conservation projects and, to this end, artificial intelligence techniques have been proposed. Hence, in this study the water quality components of Goa River were predicted using a random forest regression and group method of data handling.

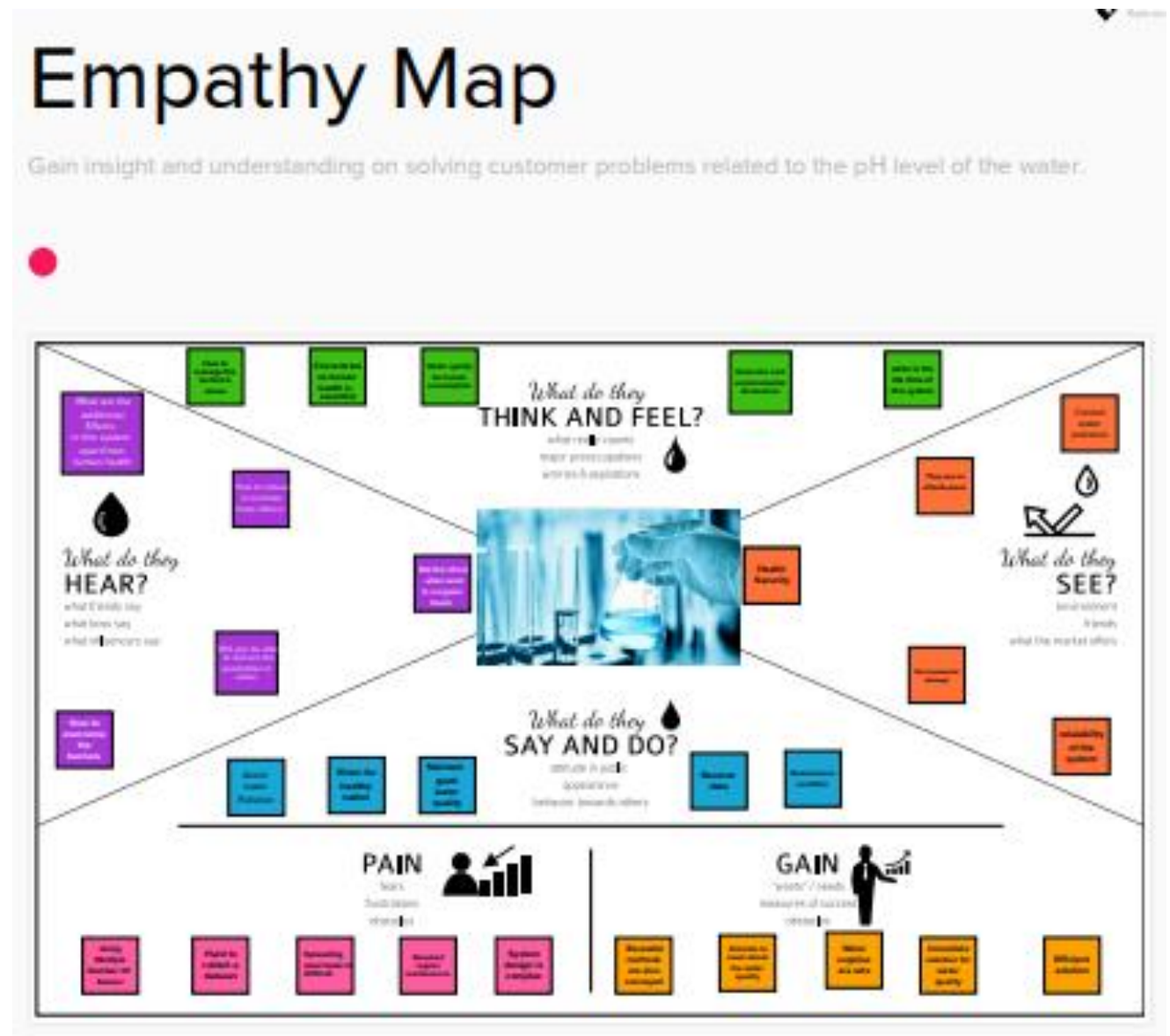
2. LITERATURE SURVEY

S.NO	TITLE	AUTHORS	DESCRIPTION
1.	Efficient Water Quality Prediction Using Supervised Machine Learning	Umair Ahmed , Rafia Mumtaz , Hirra Anwar , Asad A. Shah , Rabia Irfan and José García-Nieto	This research explored an alternative method of machine learning to predict water quality using minimal and easily available water quality parameters. The data used to conduct the study were acquired from PCRWR and contained 663 samples from 12 different sources of Rawal Lake, Pakistan. A set of representative supervised machine learning algorithms were employed to estimate WQI. This showed that polynomial regression with a degree of 2, and gradient boosting, with a learning rate of 0.1, outperformed other regression algorithms by predicting WQI most efficiently, while MLP with a configuration of (3, 7) outperformed other classification algorithms by classifying WQC most recently
2.	A review of the application	Mengyuan Zhu,	More advanced sensors, including soft sensors, should be

	of machine learning in water quality evaluation	Jiawei Wang, Xiao Yang, Yu Zhang, Linyu Zhang, Hongqiang Ren, Bing Wu, Lin Ye	developed and applied in water quality monitoring to collect sufficiently accurate data to facilitate the application of machine learning approaches. The feasibility and reliability of the algorithms should be improved, and more universal algorithms and models should be developed according to the water treatment and management requirements. Interdisciplinary talent with
3.	International Research Journal of Modernization in Engineering Technology and Science	Sai Sreeja Kurra, Sambangi Geethika Naidu, Sravani Chowdala, Sree Chithra Yellanki, Dr. B. Esther Sunanda.	To analyze overall water quality in terms of potability, ten water quality factors were used for each data set. pH, Hardness, Solids, Chloramines, Sulfate, Conductivity, Organic carbon, Trihalomethanes, Turbidity, and Potability were among the metrics studied. The choice of parameters was influenced by the fact that they are all commonly monitored critical parameters with well-defined water quality standards. The predictive modeling described in this paper, on the other hand, is adaptable enough to function with any number of parameters.

3. SYSTEM ANALYSIS

3.1 Empathy Map



3.2 Brainstorm and ideation process

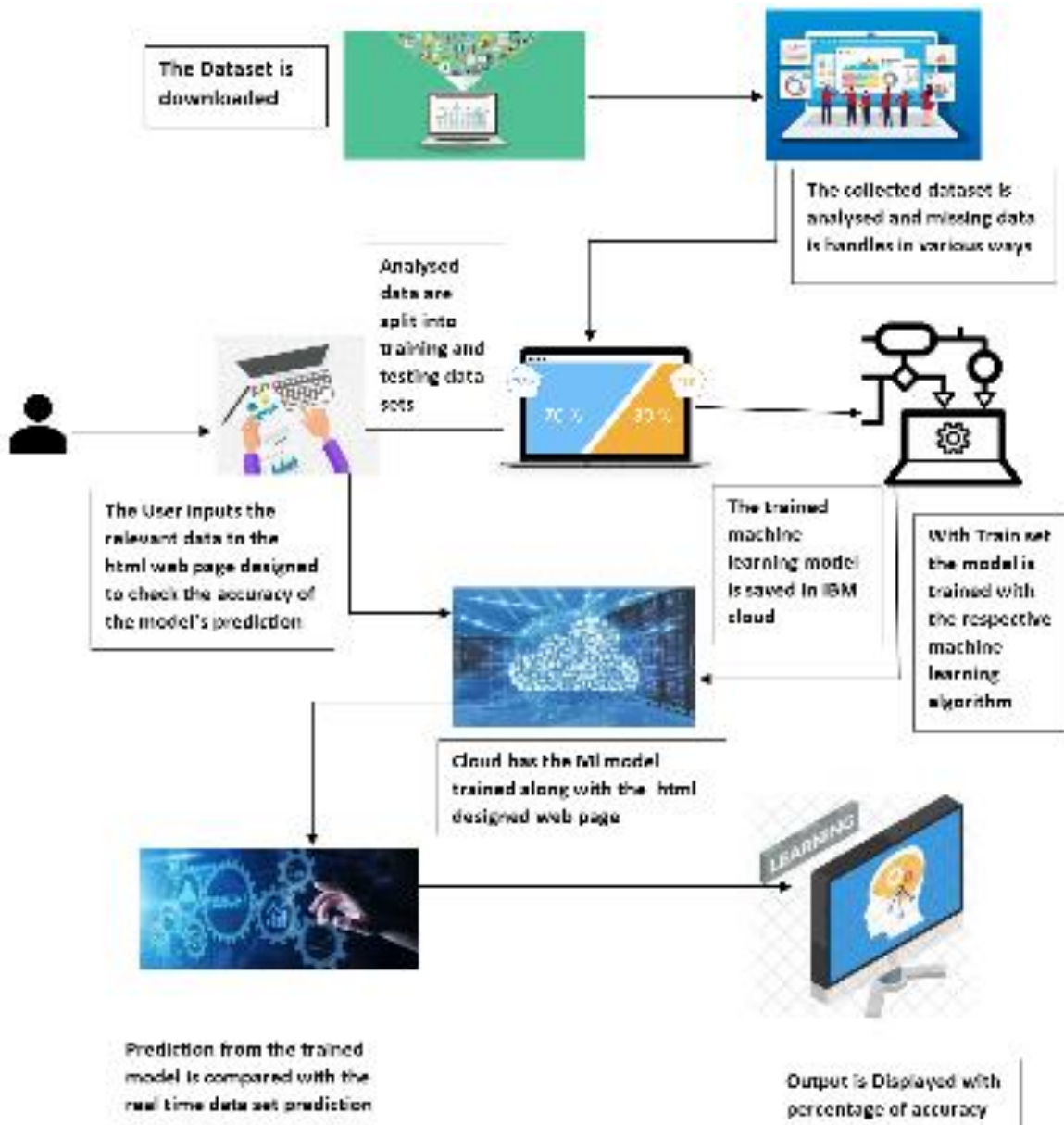
3.4 Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> Water is a necessity for life, and as such that necessity needs to be protected. Water testing will uncover any harmful substances in your water supply and may give insight into any health issues that you are experiencing. The quality of water is a major concern for people living in urban areas. Contaminated water and poor sanitation are linked to transmission of diseases such as cholera, diarrhoea, dysentery, hepatitis A, typhoid and polio. So this project aims at building a Machine Learning (ML) model to Predict Water Quality by considering all water quality standard indicators.
2.	Idea / Solution description	<ul style="list-style-type: none"> By using Machine Learning algorithm a model is created to collect data about water and analyse the quality of the water and predict the quality percentage of it and based on which we can utilize it.
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> In the past they found water quality with the help of WQI and WQC. Now the solution is with the help of advanced artificial intelligence and it includes seven parameters. User Friendly and Eco friendly.
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> Water quality has been threatened by various pollutants. Therefore, modelling and predicting water quality have become very important in controlling

		<p>water pollution. In this work, advanced artificial intelligence (AI) algorithms are developed to predict water quality index (WQI) and water quality classification (WQC). This is the impact of this statement.</p>
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> ● Affording a model would not cost much, thus it allows one to get hold of it easily. Hence it provides a better profit if marketed.
6.	Scalability of the Solution	<ul style="list-style-type: none"> ● Improves financial management. Secured and safe to use. Insights about money management. ● Scalability of this solution can handle any amount of data and perform many computations in a cost effective and timesaving way to instantly serve millions of users residing at global locations. ● Helps in getting all required aspects of water.

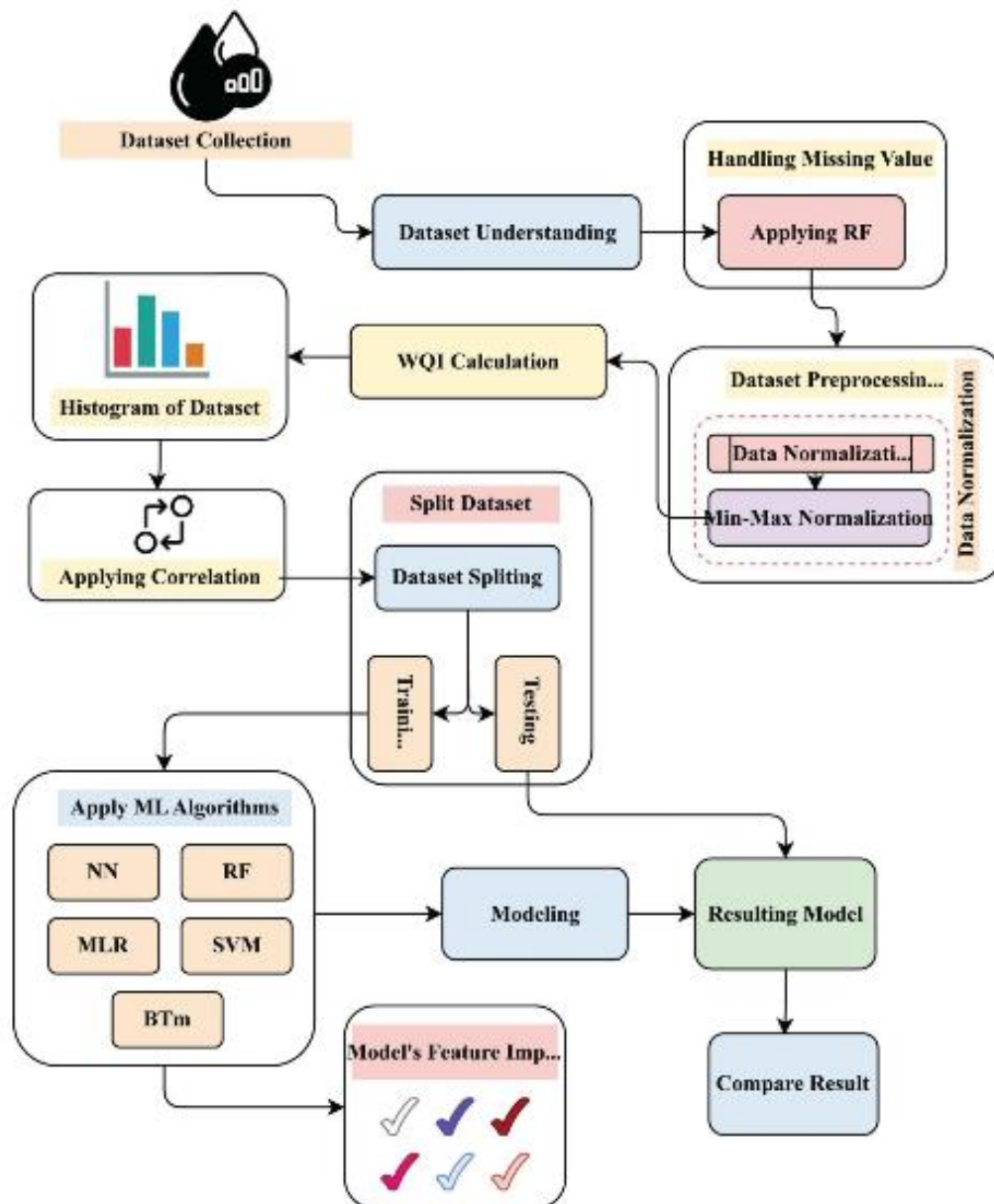
4.EXPERIMENTAL DESCRIPTIONS

4.1 Technical Block Diagram



5.PROJECT DESIGN PHASE

5.1 Data flow diagram



6.SOFTWARE DESCRIPTION

6.1 Machine Learning

Machine Learning is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: ***The ability to learn.*** Machine learning is actively being used today, perhaps in many more places than one would expect.

Machine Learning algorithm

Machine learning algorithm into three main parts.

1. **A Decision Process:** In general, machine learning algorithms are used to make a prediction or classification. Based on some input data, which can be labelled or unlabeled, your algorithm will produce an estimate about a pattern in the data.
2. **An Error Function:** An error function serves to evaluate the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model.
3. **An Model Optimization Process:** If the model can fit better to the data points in the training set, then weights are adjusted to reduce the discrepancy between the known example and the model estimate. The algorithm will repeat this evaluate and optimize process, updating weights autonomously until a threshold of accuracy has been met.

Methods in Machine Learning

Machine learning classifiers fall into three primary categories.

1.Supervised machine learning

Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately. This occurs as part of the cross validation process to ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve for a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, support vector machine (SVM), and more.

2.Unsupervised machine learning

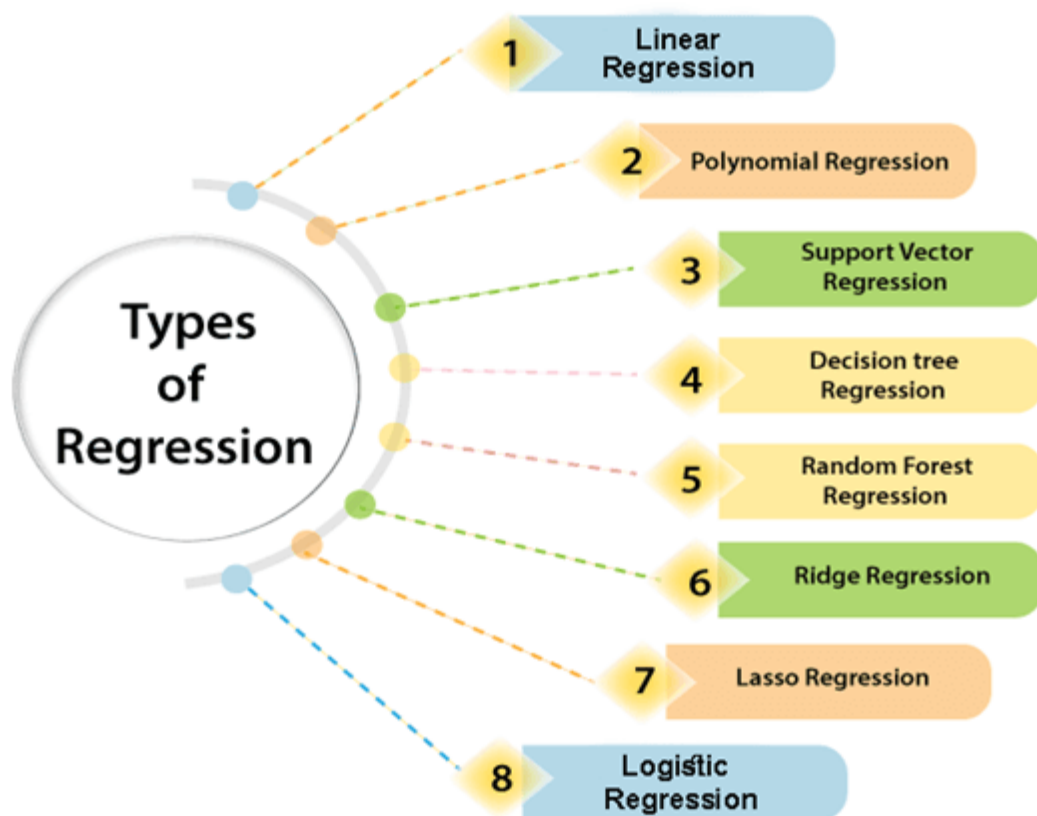
Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction; principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, probabilistic clustering methods, and more.

3.Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data (or not being able to afford to label enough data) to train a supervised learning algorithm.

6.2 Regression Models in ML

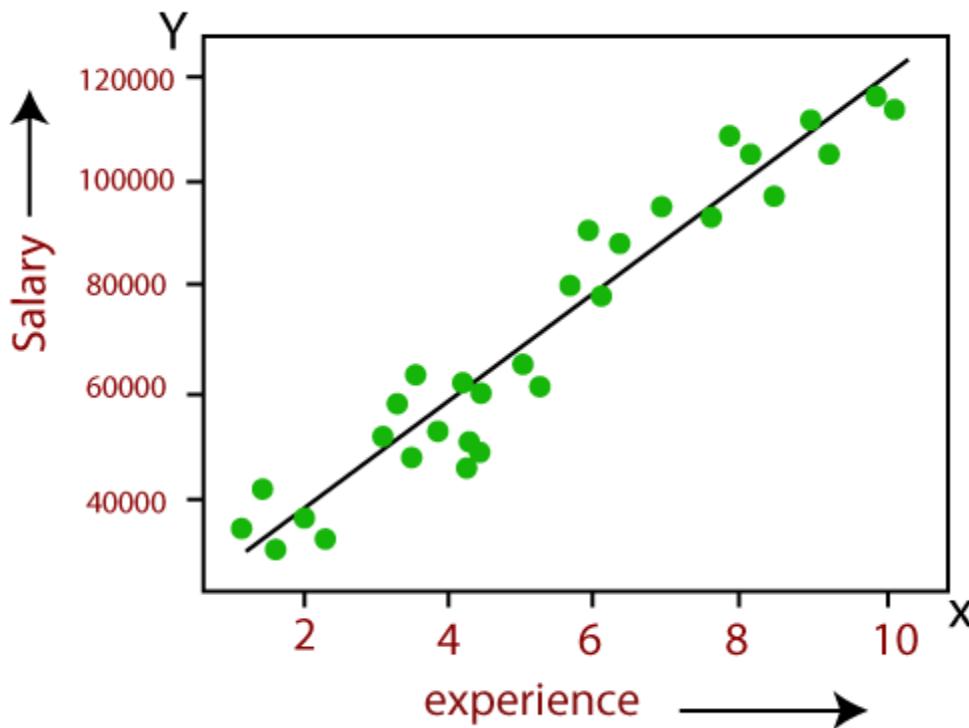
There are various types of regressions which are used in data science and machine learning. Each type has its own importance on different scenarios, but at the core, all the regression methods analyze the effect of the independent variable on dependent variables. Here we are discussing some important types of regression which are given below:



Linear Regression:

- Linear regression is a statistical regression method which is used for predictive analysis.
-

- It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables.
- It is used for solving the regression problem in machine learning.
- Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.
- If there is only one input variable (x), then such linear regression is called **simple linear regression**. And if there is more than one input variable, then such linear regression is called **multiple linear regression**.
- The relationship between variables in the linear regression model can be explained using the below image. Here we are predicting the salary of an employee on the basis of **the year of experience**.



- Below is the mathematical equation for Linear regression:

1. $Y = aX + b$

Here, Y = dependent variables (target variables),
 X = Independent variables (predictor variables),
 a and b are the linear coefficients

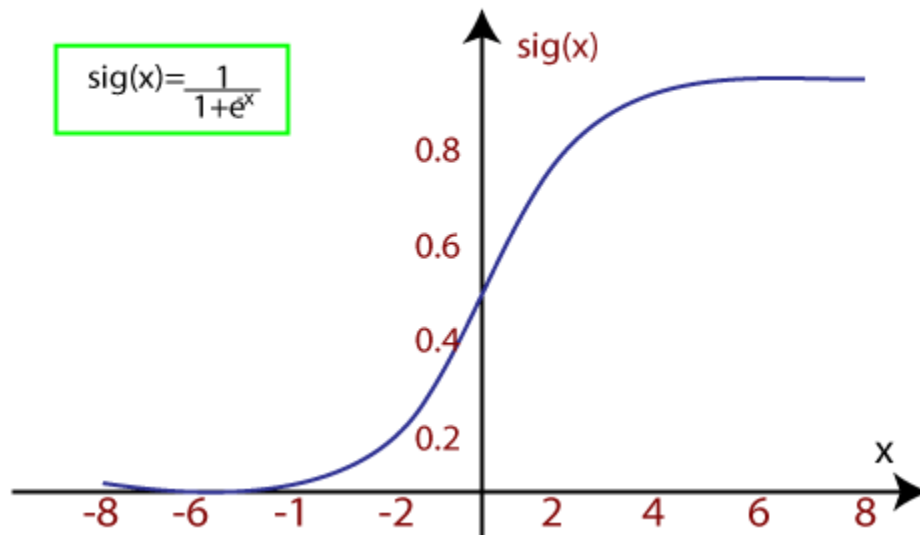
Logistic Regression:

- Logistic regression is another supervised learning algorithm which is used to solve the classification problems. In **classification problems**, we have dependent variables in a binary or discrete format such as 0 or 1.
- Logistic regression algorithm works with the categorical variable such as 0 or 1, Yes or No, True or False, Spam or not spam, etc.
- It is a predictive analysis algorithm which works on the concept of probability.
- Logistic regression is a type of regression, but it is different from the linear regression algorithm in the term how they are used.
- Logistic regression uses **sigmoid function** or logistic function which is a complex cost function. This sigmoid function is used to model the data in logistic regression. The function can be represented as:

$$f(x) = \frac{1}{1 + e^{-x}}$$

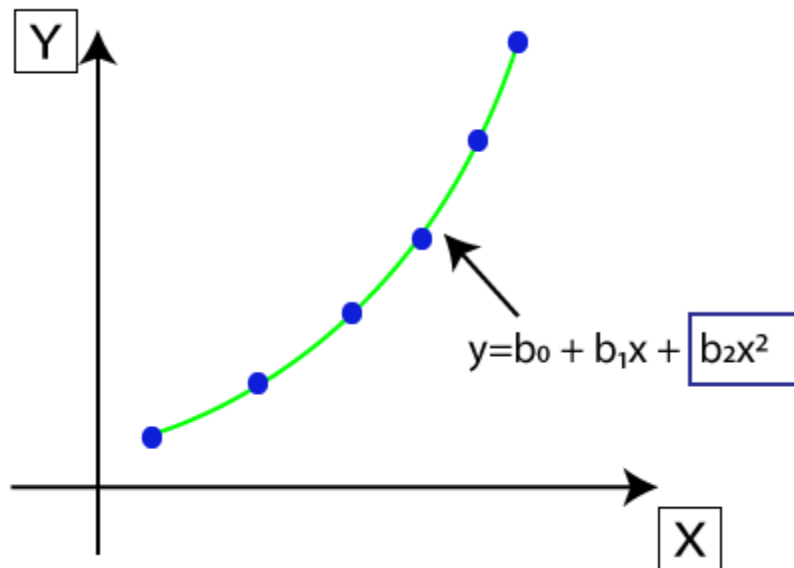
- $f(x)$ = Output between the 0 and 1 value.
- x = input to the function
- e = base of natural logarithm.

When we provide the input values (data) to the function, it gives the S-curve as follows



Polynomial Regression:

- Polynomial Regression is a type of regression which models the **non-linear dataset** using a linear model.
- It is similar to multiple linear regression, but it fits a non-linear curve between the value of x and corresponding conditional values of y.
- Suppose there is a dataset which consists of datapoints which are present in a non-linear fashion, so for such case, linear regression will not best fit to those datapoints. To cover such datapoints, we need Polynomial regression.
- **In Polynomial regression, the original features are transformed into polynomial features of given degree and then modeled using a linear model.** Which means the datapoints are best fitted using a polynomial line.



SVM Regression:

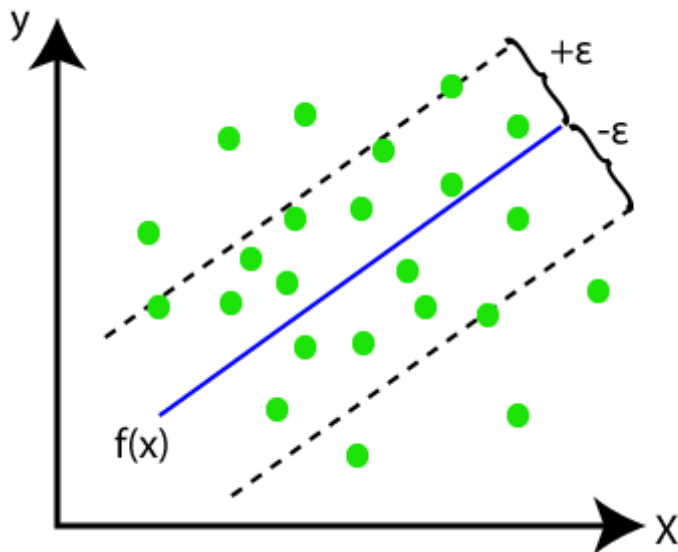
Support Vector Machine is a supervised learning algorithm which can be used for regression as well as classification problems. So if we use it for regression problems, then it is termed as Support Vector Regression.

Support Vector Regression is a regression algorithm which works for continuous variables. Below are some keywords which are used in **Support Vector Regression**:

- **Kernel:** It is a function used to map a lower-dimensional data into higher dimensional data.
- **Hyperplane:** In general SVM, it is a separation line between two classes, but in SVR, it is a line which helps to predict the continuous variables and cover most of the datapoints.
- **Boundary line:** Boundary lines are the two lines apart from hyperplane, which creates a margin for datapoints.
-

- **Support vectors:** Support vectors are the datapoints which are nearest to the hyperplane and opposite class.

In SVR, we always try to determine a hyperplane with a maximum margin, so that maximum number of datapoints are covered in that margin. *The main goal of SVR is to consider the maximum datapoints within the boundary lines and the hyperplane (best-fit line) must contain a maximum number of datapoints.* Consider the below image:

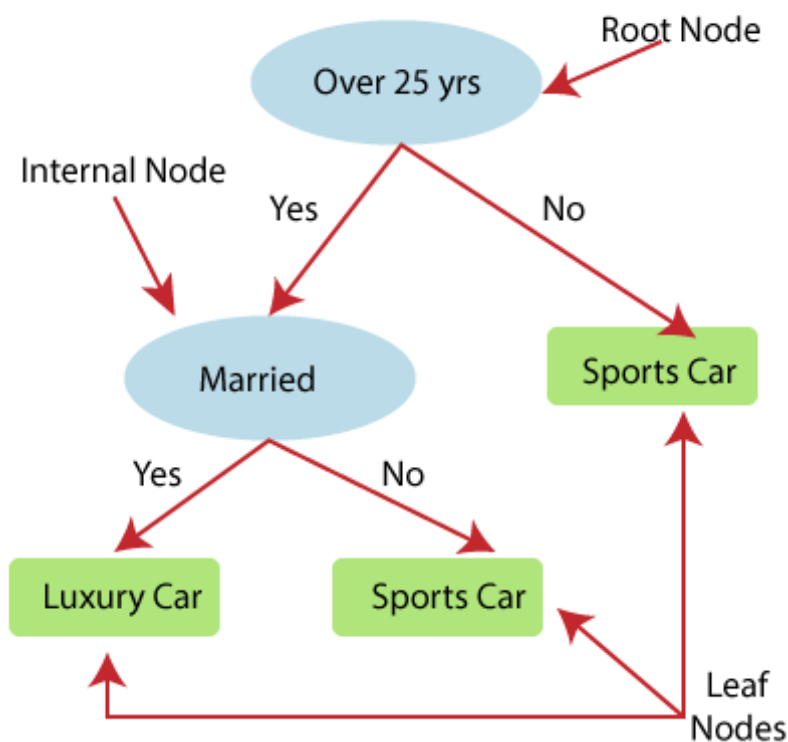


Here, the blue line is called hyperplane, and the other two lines are known as boundary lines.

Decision Tree Regression:

- Decision Tree is a supervised learning algorithm which can be used for solving both classification and regression problems.
- It can solve problems for both categorical and numerical data
- Decision Tree regression builds a tree-like structure in which each internal node represents the "test" for an attribute, each branch represent the result of the test, and each leaf node represents the final decision or result.

- A decision tree is constructed starting from the root node/parent node (dataset), which splits into left and right child nodes (subsets of dataset). These child nodes are further divided into their children node, and themselves become the parent node of those nodes. Consider the below image:

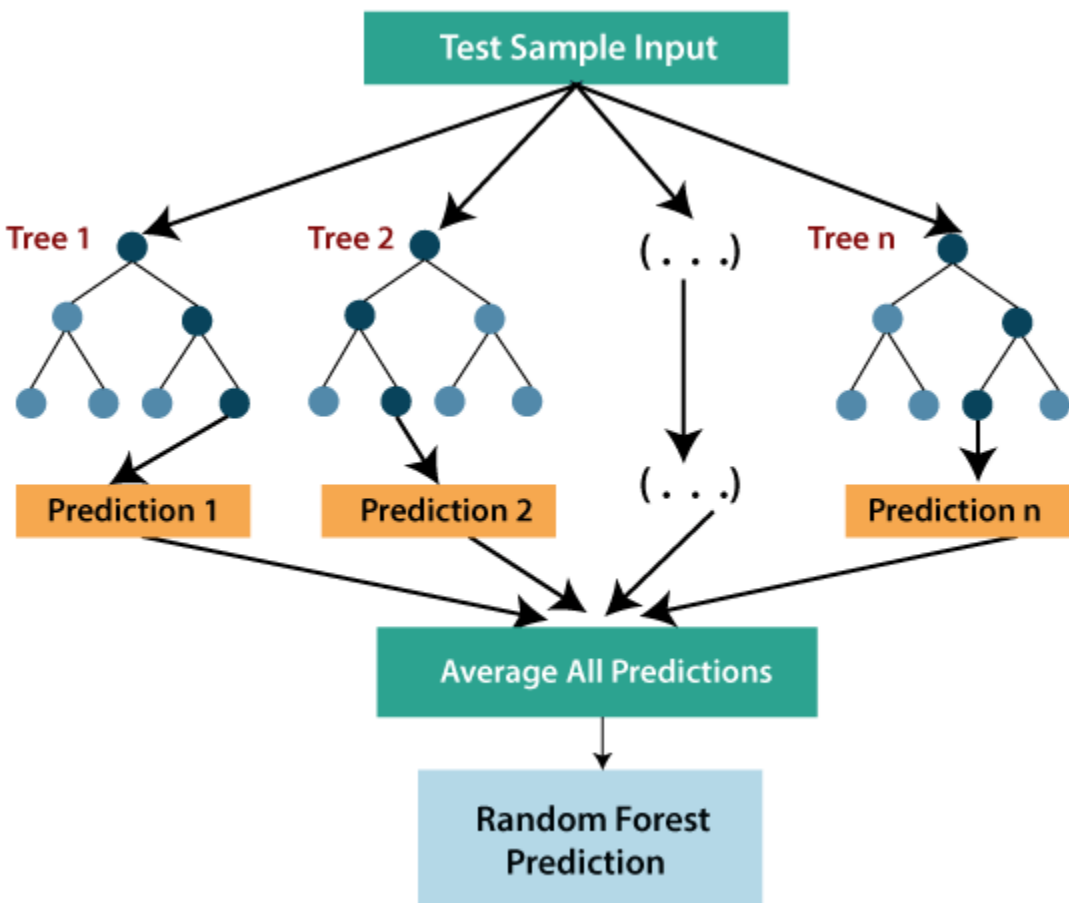


Above image showing the example of Decision Tree regression, here, the model is trying to predict the choice of a person between Sports cars or Luxury car.

- Random forest is one of the most powerful supervised learning algorithms which is capable of performing regression as well as classification tasks.
- The Random Forest regression is an ensemble learning method which combines multiple decision trees and predicts the final output based on the average of each tree output. The combined decision trees are called as base models, and it can be represented more formally as:

$$g(x) = f_0(x) + f_1(x) + f_2(x) + \dots$$

- Random forest uses **Bagging or Bootstrap Aggregation** technique of ensemble learning in which aggregated decision tree runs in parallel and do not interact with each other.
- With the help of Random Forest regression, we can prevent Overfitting in the model by creating random subsets of the dataset.



Ridge Regression:

- Ridge regression is one of the most robust versions of linear regression in which a small amount of bias is introduced so that we can get better long term predictions.
- The amount of bias added to the model is known as **Ridge Regression penalty**. We can compute this penalty term by multiplying with the lambda to the squared weight of each individual features.
- The equation for ridge regression will be:

$$L(x, y) = \text{Min}(\sum_{i=1}^n (y_i - w_i x_i)^2 + \lambda \sum_{i=1}^n (w_i)^2)$$

- A general linear or polynomial regression will fail if there is high collinearity between the independent variables, so to solve such problems, Ridge regression can be used.
- Ridge regression is a regularization technique, which is used to reduce the complexity of the model. It is also called as **L2 regularization**.
- It helps to solve the problems if we have more parameters than samples.

Lasso Regression:

- Lasso regression is another regularization technique to reduce the complexity of the model.
- It is similar to the Ridge Regression except that penalty term contains only the absolute weights instead of a square of weights.
- Since it takes absolute values, hence, it can shrink the slope to 0, whereas Ridge Regression can only shrink it near to 0.
- It is also called as **L1 regularization**. The equation for Lasso regression will be:

$$L(x, y) = \text{Min}(\sum_{i=1}^n (y_i - w_i x_i)^2 + \lambda \sum_{i=1}^n |w_i|)$$

6.3 Data Visualization in Machine Learning

As we all knew that there is a huge buzz going over the term **data**, like Big data, Data science, Data Analysts, Data Warehouse, Data mining etc. which emphasize that, In the current era data plays a major role in influencing day to day activities of the mankind. Everyday we are generating more than 2.5 quintillion(10^{18}) bytes of data([Link](#)) ranging from our Text messages, Images, emails, till data from autonomous cars, IOT devices etc. With such huge amount of data being available on hand, leveraging useful information from this data can help each and every organization very much, for getting a clear insight on several areas like, what can bring a boost for their organization's revenue, which field needs more focus, how to seek more customer's attention etc. Machine learning(ML), Data science are some of the interrelated areas of Artificial Intelligence(AI) where this task of learning from data is done in a huge extent on these recent days.

Why Visualization?

Do you think giving you the data of lets say 1 Million points in a table/Database file and asking you to provide your inferences by just seeing the data on that table is feasible? Unless you're a super human its not possible. This is when we make use of Data visualization, wherein all the data will be transformed into some form of plots and analyzed further from that. As being a human, we are more used to

grasp a lot of info from diagrammatic representation than the counterpart.

As a human, we can just visualize anything in either in 2-d or 3-d. But trust me almost of the data that you obtain in real world won't be this way. As a Machine learning engineer, working with more than 1000-dimensional data is very common. So what can we do in such cases where data is more than 3D

? *There are some **Dimensionality Reduction(DR)** techniques*

like [PCA](#), [TSNE](#), [LDA](#) etc which helps you to convert data from a higher dimension to a 2D or 3D data in order to visualize them. There may be some loss of information with each DR techniques, but only they can help us visualize very high dimensional data on a 2d plot. TSNE is one of the state of the art DR technique employed for visualization of high dimensional data.

From perspective of building models , By visualizing the data we can find the hidden patterns, Explore if there are any clusters within data and we can find if they are linearly separable/too much overlapped etc. From this initial analysis we can easily rule out the models that won't be suitable for such a data and we will implement only the models that are suitable, without wasting our valuable time and the computational resources.

7.PROJECT PLANNING & SPRINT DELIVERIES

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint1	Data Collection	USN-1,2	Collecting/downloading dataset for pre-processing .	10	High	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A
Sprint1		USN-1,2	Data pre-processing-formats the data and handles the missing data in the dataset..	10	Medium	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A
Sprint2	Model Building	USN-1,2	Calculate the Water Quality Index (WQI) using specified formula for every parameter.	10	High	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A
Sprint2		USN-1,2	Splitting the data into training and testing data set from the entire dataset.	10	High	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A
Sprint3	Training and Testing	USN-1,2	Training the model using Random Forest Regression algorithm and testing the performance of the model (accuracy rate)	20	High	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A
Sprint4	Implementation of Web page	USN-1,2	Implementing the web page for collecting the data from user	10	High	Afra Iman A Fathima koya Nilora S

						Farhana K Hameetha Shajini A
Sprint4		USN-1,2	Deploying the model using IBM Cloud and IBM Watson Studio	10	Medium	Afra Iman A Fathima koya Nilora S Farhana K Hameetha Shajini A

8.PROJECT DEVELOPMENT PHASE

Importing libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
```

Reading Dataset

```
data = pd.read_csv('water_dataX.csv',encoding='ISO-8859-1',low_memory=False)
```

Analyze the data

```
data.head()
data.describe()
data.info()
data.shape
```

Handling Missing Values

```
data.isnull().any()
data.isnull().sum()
data.dtypes
data["Temp"]=pd.to_numeric(data["Temp"],errors='coerce')
data["D.O. (mg/l)"]=pd.to_numeric(data["D.O. (mg/l)"],errors='coerce')
```

```

data['PH']=pd.to_numeric(data['PH'],errors='coerce')
data['B.O.D. (mg/l)']=pd.to_numeric(data['B.O.D. (mg/l)'],errors='coerce')
data['CONDUCTIVITY (μmhos/cm)']=pd.to_numeric(data['CONDUCTIVITY
(μmhos/cm)'],errors='coerce')
data['NITRATENAN N+ NITRITENANN
(mg/l)']=pd.to_numeric(data['NITRATENAN N+ NITRITENANN
(mg/l)'],errors='coerce')
data['TOTAL COLIFORM (MPN/100ml)Mean']=pd.to_numeric(data['TOTAL
COLIFORM (MPN/100ml)Mean'],errors='coerce')
data.isnull().sum()
data['Temp'].fillna(data['Temp'].mean(),inplace=True)
data['D.O. (mg/l)'].fillna(data['D.O. (mg/l)'].mean(),inplace=True)
data['PH'].fillna(data['PH'].mean(),inplace=True)
data['CONDUCTIVITY (μmhos/cm)'].fillna(data['CONDUCTIVITY
(μmhos/cm)'].mean(),inplace=True)
data['B.O.D. (mg/l)'].fillna(data['B.O.D. (mg/l)'].mean(),inplace=True)
data['NITRATENAN N+ NITRITENANN (mg/l)'].fillna(data['NITRATENAN N+
NITRITENANN (mg/l)'].mean(),inplace=True)
data['TOTAL COLIFORM (MPN/100ml)Mean'].fillna(data['TOTAL COLIFORM
(MPN/100ml)Mean'].mean(),inplace=True)
data.drop(["FECAL COLIFORM (MPN/100ml)"],axis=1,inplace=True)
data=data.rename(columns = {'D.O. (mg/l)': 'do'})
data=data.rename(columns = {'CONDUCTIVITY (μmhos/cm)': 'co'})
data=data.rename(columns = {'B.O.D. (mg/l)': 'bod'})
data=data.rename(columns = {'NITRATENAN N+ NITRITENANN (mg/l)': 'na'})
data=data.rename(columns = {'TOTAL COLIFORM (MPN/100ml)Mean': 'tc'})
data=data.rename(columns = {'STATION CODE': 'station'})

```

```
data=data.rename(columns = {'LOCATIONS': 'location'})
data=data.rename(columns = {'STATE': 'state'})
data=data.rename(columns = {'PH': 'ph'})
```

Water Quality Index (WQI) Calculation

#calculation of pH

```
data['npH']=data.ph.apply(lambda x: (100 if(8.5>=x>=7)
                                     else(80 if(8.6>=x>=8.5) or (6.9>=x>=6.8)
                                     else (60 if(8.8>=x>=8.6) or (6.8>=x>=6.7)
                                     else(40 if(9>=x>=8.8) or (6.7>=x>=6.5)
                                     else 0))))))
```

#calculation of dissolved oxygen

```
data['ndo']=data.do.apply(lambda x: (100 if(x>=6)
                                     else(80 if(6>=x>=5.1)
                                     else (60 if(5>=x>=4.1)
                                     else(40 if(4>=x>=3)
                                     else 0))))))
```

#calculation of total coliform

```
data['nco']=data.tc.apply(lambda x: (100 if(5>=x>=0)
                                     else(80 if(50>=x>=5)
                                     else (60 if(500>=x>=50)
                                     else(40 if(10000>=x>=500)
                                     else 0))))))
```

#calculation of B.D.O

```
data['nbdo']=data.bod.apply(lambda x:(100 if(3>=x>=0)
```

```

else(80 if(6>=x>=3)
else (60 if(80>=x>=6)
else(40 if(125>=x>=80)
else 0))))))
#calculation of electric conductivity
data['nec']=data.co.apply(lambda x:(100 if(75>=x>=0)
else(80 if(150>=x>=75)
else (60 if(225>=x>=150)
else(40 if(300>=x>=225)
else 0))))))
#calculation of nitrate
data['nna']=data.na.apply(lambda x:(100 if(20>=x>=0)
else(80 if(50>=x>=20)
else (60 if(100>=x>=50)
else(40 if(200>=x>=100)
else 0))))))
#Calculation of Water Quality Index WQI
data['wph']=data.npH*0.165
data['wdo']=data.ndo*0.281
data['wbdo']=data.nbdo*0.234
data['wec']=data.nec*0.009
data['wna']=data.nna*0.028
data['wco']=data.nco*0.281
data['wqi']=data.wph+data.wdo+data.wbdo+data.wec+data.wna+data.wco
data

```

```
#Calculation of overall WQI for each year
average = data.groupby('year')['wqi'].mean()
average.head()
```

Splitting Dependent and Independent Columns

```
data.head()
data.drop(['location','station','state'],axis =1,inplace=True)
x=data.iloc[:,1:7].values
x.shape
y=data.iloc[:,-1:].values
y.shape
print(x)
print(y)
```

Splitting the Data Into Train and Test

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state=10)
```

Random_Forest_Regression

```
#Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
from sklearn.ensemble import RandomForestRegressor
```

```
regressor = RandomForestRegressor(n_estimators = 10, random_state = 0)
regressor.fit(x_train, y_train)
y_pred = regressor.predict(x_test)
```

Model Evaluation

```
from sklearn import metrics
print('MAE:',metrics.mean_absolute_error(y_test,y_pred))
print('MSE:',metrics.mean_squared_error(y_test,y_pred))
print('RMSE:',np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
```

```
#accuracy of the model
metrics.r2_score(y_test, y_pred)
```

Save The Model

```
import pickle
pickle.dump(regressor,open('wqi.pkl', 'wb'))
model = pickle.load(open('wqi.pkl','rb'))
```

User Data Link

<https://drive.google.com/drive/folders/1v9iJ825c62QU6gJPVs1Zjoh2phmoS-JY>

9.WEBPAGE DEVELOPMENT

```
import numpy as np
from flask import Flask,render_template,request
import pickle
app= Flask(__name__)
model=pickle.load(open(r'F:\Project_demo\wqi.pkl','rb'))
@app.route('/')
def home() :
    return render_template("web.html")
@app.route('/login',methods = ['POST'])
def login() :
    year = request.form["year"]
    do = request.form["do"]
    ph = request.form["ph"]
    co = request.form["co"]
    bod = request.form["bod"]
    tc = request.form["tc"]
    na = request.form["na"]
    total =
[[float(do),float(ph),float(co),float(bod),float(na),float(tc)]]
```

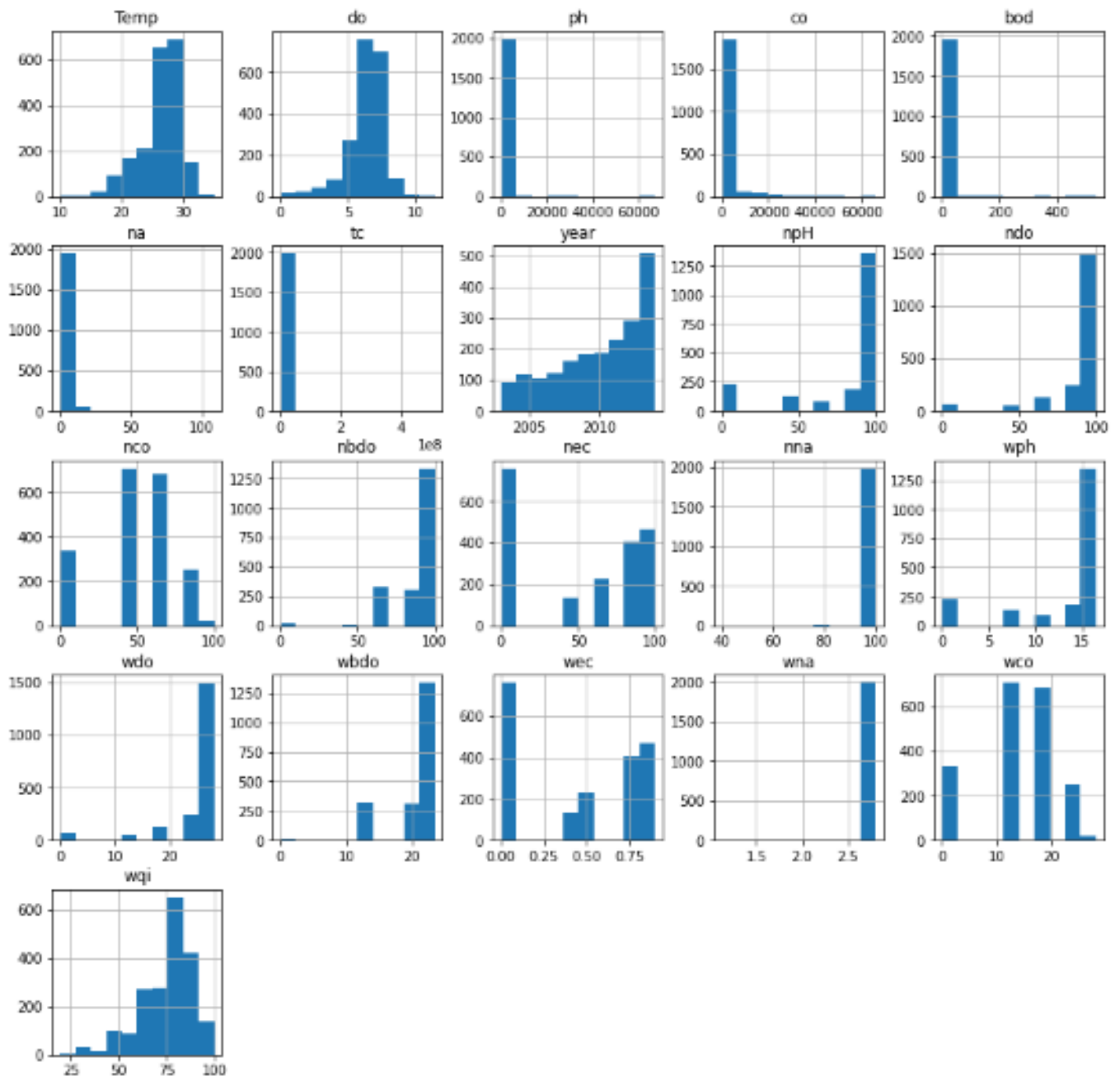
```

y_pred = model.predict(total)
y_pred = y_pred[[0]]
if(y_pred >= 95 and y_pred<=100):
    return render_template("web.html",showcase = 'Excellent,
The Predicted Value Is'+ str(y_pred))
elif(y_pred >= 89 and y_pred<=94):
    return render_template("web.html",showcase = 'Very Good,
The Predicted Value Is'+ str(y_pred))
elif(y_pred >= 80 and y_pred<=88):
    return render_template("web.html",showcase = 'Good, The
Predicted Value Is'+ str(y_pred))
elif(y_pred >= 65 and y_pred<=79):
    return render_template("web.html",showcase = 'Fair, The
Predicted Value Is'+ str(y_pred))
elif(y_pred >= 45 and y_pred<=64):
    return render_template("web.html",showcase = 'Marginal,
The Predicted Value Is'+ str(y_pred))
else:
    return render_template("web.html",showcase = 'Poor, The
Predicted Value Is'+ str(y_pred))
if __name__ == '__main__':
    app.run(debug = True,port=5000)

```


10.OUTPUT AND RESULTS

10.1 User Data Visualization



10.2 Model Final Prediction & Metrics

User Data

	Temp	do	ph	co	bod	na	tc	year	npH	ndo	...	nbdo	nec	nna	wph	wdo	wbdo	wec	wna	wco	wqi
0	30.6	6.7	7.5	203.0	6.940049	0.1	27.0	2014	100	100	...	60	60	100	16.5	28.10	14.04	0.54	2.8	22.48	84.46
1	29.8	5.7	7.2	189.0	2.000000	0.2	8391.0	2014	100	80	...	100	60	100	16.5	22.48	23.40	0.54	2.8	11.24	76.96
2	29.5	6.3	6.9	179.0	1.700000	0.1	5330.0	2014	80	100	...	100	60	100	13.2	28.10	23.40	0.54	2.8	11.24	79.28
3	29.7	5.8	6.9	64.0	3.800000	0.5	8443.0	2014	80	80	...	80	100	100	13.2	22.48	18.72	0.90	2.8	11.24	69.34
4	29.5	5.8	7.3	83.0	1.900000	0.4	5500.0	2014	100	80	...	100	80	100	16.5	22.48	23.40	0.72	2.8	11.24	77.14

Accuracy Score

```
from sklearn import metrics
print('MAE:',metrics.mean_absolute_error(y_test,y_pred))
print('MSE:',metrics.mean_squared_error(y_test,y_pred))
print('RMSE:',np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
```

```
MAE: 1.0140200501253205
MSE: 5.786707157894741
RMSE: 2.405557556554143
```

```
#accuracy of the model
metrics.r2_score(y_test, y_pred)
```

```
0.9684566685516488
```

10.3 Webpage design Output



The webpage features a dark blue background with a faint world map and a large, glowing blue wireframe structure on the left. The title "Urban Water Quality Prediction" is displayed in white text next to a water drop icon. Below the title, there are seven input fields for user data, followed by a "Predict" button. The output of the prediction is shown in a white box at the bottom right.

Urban Water Quality Prediction

Enter Year

Enter D.O

Enter PH

Enter Conductivity

Enter B.O.D

Enter Nitratene

Enter Total Coliform

Predict

Marginal, The Predicted Value Is[47.31]

11.CONCLUSION

A Water Quality Index (WQI) is a means by which water quality data is summarized for reporting to the public in a consistent manner. It is similar to the UV index or an air quality index, and it tells us, in simple terms, what the quality of drinking water is from a drinking water supply. The WQI measures the scope, frequency, and amplitude of water quality exceedances and then combines the three measures into one score. This calculation produces a score between 0 and 100. The higher the score the better the quality of water. The scores are then ranked into one of the five categories described below:

- Excellent: (WQI Value 95-100) – Water quality is protected with a virtual absence of impairment; conditions are very close to pristine levels. These index values can only be obtained if all measurements meet recommended guidelines virtually all of the time.
- Very Good: (WQI Value 89-94) – Water quality is protected with a slight presence of impairment; conditions are close to pristine levels.
- Good: (WQI Value 80-88) – Water quality is protected with only a minor degree of impairment; conditions rarely depart from desirable levels.
- Fair: (WQI Value 65-79) – Water quality is usually protected but occasionally impaired; conditions sometimes depart from desirable levels.
- Marginal: (WQI Value 45-64) – Water quality is frequently impaired; conditions often depart from desirable levels.
- Poor: (WQI Value 0-44) – Water quality is almost always impaired; conditions usually depart from desirable levels.

WQI scores are computed for each public water supply system that has been sampled in a sampling season. The same variables are used in the computation of the WQI for all public water supply systems and only the six most recent samples are used. However if a public water supply system is on a Boil Water Order, or it has a current contaminant exceedance, or has a THMs average above the drinking water quality guideline a WQI score is not computed.

The WQI is a summary tool and the Department does not intend to use the WQI to replace detailed analysis of drinking water quality data. The Department continues

to closely monitor and analyze drinking water quality to protect drinking water safety on a proactive basis. River water quality assessment is one of the most important tasks to enhance water resources management plans. A water quality index (WQI) considers several water quality variables simultaneously.

Traditionally WQI calculations consume time and are often fraught with errors during derivations of sub-indices. In this study, 4 standalone (random forest (RF), M5P, random tree (RT), and reduced error pruning tree (REPT)) and 12 hybrid data-mining algorithms (combinations of standalones with bagging (BA), CV parameter selection (CVPS) and randomizable filtered classification (RFC)) were used to create Iran WQI ($IRWQI_{sc}$) predictions. Six years (2012 to 2018) of monthly data from two water quality monitoring stations within the Talar catchment were compiled. Using Pearson correlation coefficients, 10 different input combinations were constructed. The data were divided into two groups (ratio 70:30) for model building (training dataset) and model validation (testing dataset) using a 10-fold cross-validation technique. The models were evaluated using several statistical and visual evaluation metrics. Result show that fecal coliform (FC) and total solids (TS) had the greatest and least effect on the prediction of $IRWQI_{sc}$. The best input combinations varied among the algorithms; generally variables with very low correlations displayed weaker performance. Hybrid algorithms improved the prediction power of several of the standalone models, but not all. Hybrid BA-RT outperformed the other models ($R^2 = 0.941$, RMSE = 2.71, MAE = 1.87, NSE = 0.941, PBIAS = 0.500). PBIAS indicated that all algorithms, with the exceptions of RT, BA-RT and CVPS-REPT, overestimated WQI values.