** **

**FINAL YEAR PROJECT**

**Water Quality Monitoring for Waterborne Diseases Prediction using Machine Learning**

**By**

**MUZAMIL MOHAMMED AHMED ABDULATIEF**

**TP048867**

**UC3F2103CS(DA)**

A report submitted in partial fulfilment of the requirements of Asia Pacific University of Technology and Innovation for the degree of

B.Sc. (Hons) Computer Science with a Specialism in Data Analytics

Supervised by Mr. RAHEEM MAFAS

2nd Marker: Dr. RAJA RAJESWARI

Oct-2021

**Acknowledgements**

I would like to give recognition to the teachers, professors and supervisors at Asia Pacific University. Firstly, I would like to express my sincere gratitude towards Mr. Dhason Padmakumar for his guidance in terms of what needs to be done for the Investigation Report as well as the Final Year Project in general as without that guidance, we would not know what we should do. I would also like to thank him for providing us with templates, reminding us about schedules and due dates for each week of my semester.

Next, I would like to express my gratitude towards my academic supervisor, Mr. Raheem Mafas, for his continuous support of my Investigation Report as well as my research. Moreover, I would like to show appreciation to him for his continuous enthusiasm and insight, as without him, I would not be able to overcome a lot of the challenges and obstacles that I ran into while performing this project. Lastly, I would like to thank him for being readily available for meetings, small discussions and chats as those kept me going and pushing to finish this project.

Furthermore, I would like to thank my friends, specifically Richard Wijaya, Johnson Wong, Kazi Nishat, Haris Emir, Yohann Adrian Meyepa, Yaya, Soh Zong Xian, Thomas Clark and Stamford See Shanghan for the stimulating discussions which kept me up at night trying to fix the issues they addressed, for the helpful advice and the encouragement that they gave me.

Lastly, I would like to thank my mother and siblings for their constant reassurance, backing and care during the many times I doubted myself and my abilities to complete my studies.

Table of Contents

[**CHAPTER 1: INTRODUCTION TO THE STUDY 1**](#_Toc85016389)

[**1.1 Background to the project 1**](#_Toc85016390)

[**1.2 Problem statements 2**](#_Toc85016391)

[**1.3 Rationale 3**](#_Toc85016392)

[**1.4 Potential benefits 3**](#_Toc85016393)

[**1.4.1 Tangible benefits 3**](#_Toc85016394)

[**1.4.2 Intangible benefits 4**](#_Toc85016395)

[**1.5 Target users 4**](#_Toc85016396)

[**1.6 Scope and objectives 4**](#_Toc85016397)

[**1.6.1 Scope 4**](#_Toc85016398)

[**1.6.2 Aim 4**](#_Toc85016399)

[**1.6.3 Objectives 4**](#_Toc85016400)

[**1.6.3 Deliverables 5**](#_Toc85016401)

[**1.6.4 Nature of Challenge 5**](#_Toc85016402)

[**1.7 Overview of this report 5**](#_Toc85016403)

[**1.8 Project Plan 7**](#_Toc85016404)

[**CHAPTER 2: LITERATURE REVIEW 9**](#_Toc85016405)

[**2.1 Introduction 9**](#_Toc85016406)

[**2.2 Domain research 10**](#_Toc85016407)

[**2.2.1 Water Quality Monitoring 10**](#_Toc85016408)

[**2.2.2 Waterborne Diseases 12**](#_Toc85016409)

[**2.2.3 Supervised Machine Learning Algorithms 14**](#_Toc85016410)

[**2.2.4 Support Vector Machines 15**](#_Toc85016411)

[**2.2.5 Long Short-Term Memory 17**](#_Toc85016412)

[**2.3 Similar Systems 18**](#_Toc85016413)

[**2.4 Summary 22**](#_Toc85016414)

[**CHAPTER 3: TECHNICAL RESEARCH 23**](#_Toc85016415)

[**3.1 Programming language chosen 23**](#_Toc85016416)

[**3.1.1 Python 25**](#_Toc85016417)

[**3.2 IDE (Interactive Development Environment) chosen 26**](#_Toc85016418)

[**3.3 Libraries/Tools chosen 27**](#_Toc85016419)

[**3.4 Operating System chosen 28**](#_Toc85016420)

[**3.4.1 Hardware Specification 28**](#_Toc85016421)

[**3.4.2 Software Specification 28**](#_Toc85016422)

[**3.5 Summary 29**](#_Toc85016423)

[**CHAPTER 4: METHODOLOGY 30**](#_Toc85016424)

[**4.1 Introduction 30**](#_Toc85016425)

[**4.2 CRISP-DM Methodology 34**](#_Toc85016426)

[**4.2.1 Process Flow Overview 38**](#_Toc85016427)

[**4.3 Summary 39**](#_Toc85016428)

[**CHAPTER FIVE: DATA ANALYSIS 40**](#_Toc85016429)

[**5.1 Introduction 40**](#_Toc85016430)

[**5.2 Data Exploration 41**](#_Toc85016431)

[**5.3 Data Cleaning 42**](#_Toc85016432)

[**5.3.1 Data Conversion 42**](#_Toc85016433)

[**5.3.2 Dealing with NULL Values 43**](#_Toc85016434)

[**5.3.3 Removing Outliers 51**](#_Toc85016435)

[**5.3.4 WQI Calculation 54**](#_Toc85016436)

[**5.3.5 Data Transformation 58**](#_Toc85016437)

[**5.4 Data Modelling 61**](#_Toc85016438)

[**5.4.1 Data Splitting 61**](#_Toc85016439)

[**5.4.2 Basic Classifier Training 62**](#_Toc85016440)

[**5.4.3 Optimised Models 64**](#_Toc85016441)

[**5.5 Model Evaluation 65**](#_Toc85016442)

[**5.5.1 SVM Models 65**](#_Toc85016443)

[**5.5.2 LSTM Model 66**](#_Toc85016444)

[**5.6 Model Deployment 67**](#_Toc85016445)

[**5.7 Summary 70**](#_Toc85016446)

[**CHAPTER 6: Results and Discussion 71**](#_Toc85016447)

[**6.1 Introduction 71**](#_Toc85016448)

[**6.2 Results and Discussion 71**](#_Toc85016449)

[**CHAPTER 7: CONCLUSIONS AND REFLECTIONS 72**](#_Toc85016450)

[**References 73**](#_Toc85016451)

[**Appendices 83**](#_Toc85016452)

[**Poster 83**](#_Toc85016453)

[**Confidentiality Document 84**](#_Toc85016454)

[**Library Cataloguing Details 85**](#_Toc85016455)

[**Project Meeting Log Sheets 86**](#_Toc85016456)

[**Project Proposal Form 87**](#_Toc85016457)

[**Project Specification Form 90**](#_Toc85016458)

[**Ethics Form 96**](#_Toc85016459)

[**Gantt Chart 100**](#_Toc85016460)

# CHAPTER 1: INTRODUCTION TO THE STUDY

## 1.1 Background to the project

Water is a very important element to all living things, including human beings. This is because water does a lot of things for the body, some of which include controlling body temperature, helping with the absorption of nutrients, enhancing oxygen circulation within the body and fighting of illnesses (Silver, 2020). These are all true only to those that consume clean water. The opposite of that is contaminated water which can occur in one of two ways, either through microbes which are bacteria and viruses or through chemical pollution. This study will focus on contaminated water caused by microbes. So, what happens to people that consume contaminated water? Contaminated water can carry diseases that can affect a person almost immediately and if they do not affect them immediately, they may still get long-term effects such as liver or kidney damage (United States Environmental Protection Agency, 2021).

Reid (2021) reports that currently, 2 billion people do not have access to proper sanitation (Reid, 2021). According to the World Health Organisation (2019), half of the world’s population will be living in water-stressed areas by 2025, which is in 4 years from now (World Health Organisation, 2019). Additionally, over 800 million people globally lack access to clean water and more than 800 children below the age of 5 pass away from diarrhoea related to poor sanitation and dirty water (Reid, 2021). Through the power of prediction, scientists and doctors alike would be able to determine what disease a person would contract if they were to use this water for drinking or just day-to-day activities such as showering and cleaning. Furthermore, they would also be able to determine which water sources can be considered clean and which can be considered contaminated.

Human beings have long been interested and captivated by what might happen tomorrow, the day after that, or at any point in time in the future. Due to the huge boom of technology in the 21st century, this has now become possible to a degree using prediction models and these predictions models work with one thing at the centre, data. The reason I say to a degree is because the real world is constantly changing and therefore conditions are never constantly the same so predictions cannot be accurately made 100% of the time. But being able to improve this accuracy gradually over time is something that people have been trying to do and that is why there are so many prediction models currently out there. Being able to apply these prediction models in different areas of expertise has greatly helped all kinds of occupations. For example, using prediction, people are able to find out what the expected weather would be for the rest of the week. By being able to apply this type of power, the medical field will help bring about new advances. Predictive models have also helped improve customer service by being able to predict, for example, what they would like to purchase next based on previous purchases and recommending that product to them. Predictive models and machine learning are coherent as predictive models generally include some form of machine learning algorithm imbued in them. According to IBM Cloud Education (2020), machine learning is a subdivision of artificial intelligence (AI) and computer science which mimics the way humans learn using data and algorithms with the aim of improving accuracy (IBM Cloud Education, 2020). Through this, predictive models can then be trained to act on new values in order to obtain the desirable results. Hence, this study aims to utilise machine learning in order to be able to predict the presence of waterborne diseases by monitoring a water source.

## 1.2 Problem statements

As mentioned previously, this study will focus on contaminated water through microbes. This type of contaminated water can cause many different kinds of diseases, some of which are life threatening. Some examples of waterborne diseases include cholera, Hepatitis A and E. coli (Lenntech, 2021). In most cases, unsafe water sources can be found in developing countries such as countries in Africa. According to Butler (2015), a lack of education leads to the healthcare shortage in many developing countries. In such cases, members of the community lack the education necessary to avoid preventable illnesses, and medical professionals lack the skills necessary to treat patients after they have become sick (Butler, 2018). Even when people do have the necessary skills to treat illnesses, they often travel to countries whose working conditions and salaries are better than the country they are in, and this is done in the pursuit of a better life. Due to this, vaccination for diseases is scarce. Moreover, waterborne diseases cause a weakened immune system and so patients are more prone to other diseases. In addition to this, most developing countries lack sufficient nutritional food, which causes malnutrition and a weakened immune system and body. A high occurrence of poor diet and infectious diseases creates a vicious cycle in these countries. Therefore, the issue that I am trying to solve through the conducting of this research is predicting the presence of waterborne diseases in a water source in order to decrease the chances of someone catching these diseases.

## 1.3 Rationale

The main reasoning behind performing this project specifically is because I can also relate to the problem in question. Coming from a third-world country like Sudan, the water crisis that is in it right now has become a serious issue for many years now. The water crisis in Sudan is caused by many reasons. For starters, their main source of water is the River Nile, which it shares with 11 other countries. Most of Sudan’s other water sources are polluted with chemicals, pesticides and waste (Mark, 2019). This, combined with the government’s inability to tackle the issue, causes diseases and infections spread quicky through the water system. This also results in women and children having to travel almost everyday to collect water from another source (Barton, 2021). By understanding the problem statement stated above, through data and a functional predictive model that has a high accuracy rate, the Sudan and other countries in the world would be able to live on longer without having to worry as much about waterborne diseases. Through data that is constantly being updated, namely live data, people would be able to tell almost immediately if a water source is contaminated and contains diseases or not. This type of data is unfortunately unavailable for this project and so a dataset is going to be used. This dataset contains data relating to multiple water sources in India. The result of this project is going to be a system that will be able to predict the presence of waterborne diseases in a water source. By changing the dataset to live data, the system would be able to produce accurate results to make more informed decisions about the usage of the water source in question.

## 1.4 Potential benefits

### **1.4.1 Tangible benefits**

* Reduces the number of people getting sick due to water-borne diseases
* Reduces the amount of money spent on healthcare due to water-borne diseases
* Increases the number of clean water sources found
* Giving recommendations for the relevant authorities to act on contaminated sources and not allow people to access them

### 1.4.2 Intangible benefits

* Increases awareness of the prevalence of water in the world
* Increases awareness of the prevalence of water-borne diseases

## 1.5 Target users

The target users of this project are going to be the general public due to the fact that this project is being performed or done using a pre-recorded dataset. If the project was done using sensors, the target audience would change and become governments and companies that could make use of the model to make more informed decisions to be able to help citizens of a country by specifying which water sources they can use and which ones are contaminated and cannot be used.

## 1.6 Scope and objectives

### 1.6.1 Scope

This project will be focusing on a dataset since no sensors are currently available for this project but with some adjustments to the system, it would be able to, in real-time, be able to predict this

### 1.6.2 Aim

The project aims to build a model that is going to be able to classify whether a water source is contaminated or not.

### 1.6.3 Objectives

The objectives of this research are as follows:

* To perform data pre-processing on the obtained dataset to get it into an appropriate format for data analytics
* To build a more suitable machine learning predictive model
* To evaluate the model performance using suitable evaluation metrics

1.6.3 Deliverables

* A machine learning classification model that will be able to classify the safety of a water source
* Users will be able to view the results and determine whether the water source is safe for use and consumption or not

### 1.6.4 Nature of Challenge

In order to build a prediction model to detect water-borne diseases, there are many obstacles that could get in the way. Firstly, there are many machine learning algorithms currently in use for prediction models, so a thorough research of these algorithms needs to be carried out to find a suitable one for this project. Secondly, the world of medicine is very broad and always evolving. This means that new diseases may discovered at any point in time making it difficult to detect them and find the variables that cause these diseases. Lastly, there are also many different variants of diseases which means that the variables and their values are going to be different for each disease.

## 1.7 **Overview of this report**

This investigation report consists of 5 main chapters and the breakdown for those is as follows. Chapter 1 of the report is the introduction, which consists of the project background, the problem statement, the rationale behind conducting this study, the potential tangible and intangible benefits of performing the project, who the target users of this study are, the scope and objectives which comprises of the aim, objectives, deliverables and the challenges that might be faced. Chapter 2 is the literature review which will compare the study being performed with other studies performed in the same area. This will examine the domain research, existing similar systems and a summary. Chapter 3 concerns the technical research that will be carried out. In this, the programming language will be selected after a few are compared for suitability. Next, the Interactive Development Environment (IDE) will be selected here along with the libraries and tools to be used. Following that, the operating system to be used will be selected and the web browser for the deployment will also be selected after a comparison is made. Chapter 4 concerns the methodology to be carried out to complete this project. In order to select the best methodology to be applied, a comparison will be conducted and based on that, one methodology will be selected. A detailed explanation of the methodology stages and how I apply them into the project then follows that. Next, the processs behind cleaning the data and preparing it for the model building will take place in chapter 4. Once the models are built they will then be evaluated in chapter 6 and the best model will be deployed. Lastly, chapter 7 is the conclusion, where a summary of what was achieved at the end of the project will be done and a reflection will be provided to reflect on where additional work needs to be done.

## 1.8 Project Plan

*Table 1: Project Plan*

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Name** | **Duration** | **Start Date** | **End Date** |
| **CHAPTER 1: INTRODUCTION TO THE STUDY** | | | |
| **Background to the project** | 1 hour | 26 May 2021 | 27 May 2021 |
| **Problem Statement** | 4 hours | 26 May 2021 | 27 May 2021 |
| **Rationale** | 1 hour | 26 May 2021 | 27 May 2021 |
| **Potential Benefits** | 30 minutes | 26 May 2021 | 27 May 2021 |
| **Target Users** | 30 minutes | 26 May 2021 | 27 May 2021 |
| **Scope and Objectives** | 1 hour | 26 May 2021 | 27 May 2021 |
| **Overview of this report** | 30 minutes | 26 May 2021 | 27 May 2021 |
| **CHAPTER 2: LITERATURE REVIEW** | | | |
| **Introduction** | 3 hours | 1 June 2021 | 2 June 2021 |
| **Domain Research** | 6 hours | 1 June 2021 | 8 June 2021 |
| **Similar Systems** | 6 hours | 4 June 2021 | 8 June 2021 |
| **Summary** | 30 minutes | 5 June 2021 | 8 June 2021 |
| **CHAPTER 3: TECHNICAL RESEARCH** | | | |
| **Programming Language chosen** | 2 hours | 27 May 2021 | 28 May 2021 |
| **IDE chosen** | 1 hour | 27 May 2021 | 28 May 2021 |
| **Libraries/Tools chosen** | 2 hours | 27 May 2021 | 28 May 2021 |
| **Operating System chosen** | 1 hour | 27 May 2021 | 28 May 2021 |
| **Summary** | 30 minutes | 27 May 2021 | 28 May 2021 |
| **CHAPTER 4: METHODOLOGY** | | | |
| **Introduction** | 2 hours | 21 May 2021 | 22 May 2021 |
| **Methods** | 2 hours | 21 May 2021 | 8 June 2021 |
| **Summary** | 30 minutes | 21 May 2021 | 22 May 2021 |
| **CHAPTER 5: DATA ANALYSIS** | | | |
| **Introduction** | 30 minutes | 2 October 2021 | 3 October 2021 |
| **Data Exploration** | 2 hours | 2 October 2021 | 3 October 2021 |
| **Data Cleaning** | 3 hours | 2 October 2021 | 3 October 2021 |
| **Data Modeling** | 2 hours | 7 October 2021 | 9 October 2021 |
| **Model Evaluation** | 1 hour | 8 October 2021 | 9 October 2021 |
| **Summary** | 30 minutes | 8 October 2021 | 9 October 2021 |
| **CHAPTER 6: RESULTS AND DISCUSSION** | | | |
| **Introduction** | 30 minutes | 8 October 2021 | 9 October 2021 |
| **Results and Discussion** | 30 minutes | 8 October 2021 | 9 October 2021 |
| **CHAPTER 7: CONCLUSION AND REFLECTIONS** | | | |
| **Conclusion and reflections** | 1 hour | 10 October 2021 | 11 October 2021 |
| **CHAPTER 8: IMPLEMENTATION** | | | |
| **Coding (research and actual coding)** | Entire semester | 19 July 2021 | 10 October 2021 |
| **Deployment** | 2 hours | 10 October 2021 | 11 October 2021 |
| **Testing** | 3 hours | 10 October 2021 | 12 October 2021 |

The reason the end date is either one day after the start date or even more is because the end date reflects the day that the part done was finalised and check one last time and noth the actual duration.

# CHAPTER 2: LITERATURE REVIEW

## 2.1 Introduction

Any research project’s success is contingent on thorough investigation and analysis prior to the start of the project. As a result, the purpose of this section of the report is to compare the system being developed with similar current systems that offer similar or the same functionality. This will provide me with a solid understanding of what is now available and how beneficial my system will be once it is deployed. As a result, this section will be divided into two primary subsections: domain research study and comparison of related systems. Many different sorts of resources will be used during the literature review in order to gather as much information as possible. These resources include research papers and publications, journals and articles.

## **2.2** **Domain research**

### 2.2.1 Water Quality Monitoring

71% of the Earth’s surface is covered with water (U.S. Geological Survey, 2021). Of this 71%, 97% of it is found in oceans, meaning it is too salty for consumption or use and only 3% of it is fresh water, with 2.5% of it inaccessible due to it being either in the form of water vapour in the air and soil, in glaciers and ice caps or it is too deep below the Earth’s surface to be excavated at a reasonable cost (Bureau of Reclamation, 2020). Due to all of this, water is an integral part of our everyday lives no matter how small it is. In order to prevent more water sources from becoming polluted, a water quality monitoring system is developed. Water quality monitoring is the act of ensuring that the characteristics of the water being observed remain unchanged to maintain its standard. The traditional way of performing this act is by manually collecting samples then sending them back to a laboratory to be analysed (Chen & Han, 2018) (Das & Jain, 2017).

However, this method has many disadvantages that cannot be ignored which include the following. The first disadvantage is that it requires specially trained staff and equipment that is specific to this sort of area are required to evaluate the quality of the water (Adu-Manu et al., 2017). Next, it is hard to monitor the quality of the water in real-time because it is time consuming to collect samples and send them back to a laboratory then wait for the results (Adu-Manu et al., 2017). Due to this, it is hard to also detect changes immediately as it will take time. Thirdly, it is expensive and resource consuming in terms of the modelling and application of a system such as this (Adu-Manu et al., 2017). Therefore, the creation of an automatic system that makes use of sensors to measure certain metrics such as temperature, pH and turbidity and compare them with pre-set measurements for each metric has been the breakthrough in this field. Several researchers have already adopted water quality monitoring systems or developed ones of themselves.

Almost all of the water quality monitoring systems that were previously done were applied using Internet of Thing (IoT) and this is because of the connectivity that it allows between multiple devices which allows them to communicate and exchange data (Das & Jain, 2017). A few examples of this type of system include the following: a very simple water quality monitoring system developed by Moparthi et al. which only makes use of multiple pH sensors and an Arduino board to collect information with the data being transferred back to the Arduino IDE installed on a computer (Moparthi et al., 2018). Another system is a low cost and real-time water quality monitoring system was developed by Das and Jain in 2017 which makes use of a pH sensor, conductivity sensor and temperature sensor. These sensors capture information and then send them to a website or a smartphone (Das & Jain, 2017). The system developed by Chen and Han on the other hand, is more complex and includes a data acquisition module fitted with four sensors which are a turbidity sensor, a conductivity sensor, dissolved oxygen sensor and a pH or oxidation reduction potential sensor. A power supply module is also fitted along with a data transmission module, data storage module and a data redistribution module (Chen & Han, 2018).

Another system developed by Myint et al. made use of 5 sensors which are an ultrasonic sensor, a pH sensor, a digital thermometer sensor, a turbidity sensor and a carbon dioxide sensor (Myint et al., 2017). Data is then displayed on a computer by using Python code with the data being transferred through a radio-frequency module. Lastly, a system developed by Huan et al. consisted of four parts, a perception layer which was responsible for collection information, a transport layer which comprised of the communications base station and the network, the platform layer which was used to store the data and finally the application layer, which controlled the operation of the aerator and the bottom module (Huan et al., 2020). The way it collected information was by employing a temperature sensor, a pH sensor and a dissolved oxygen sensor.

### 2.2.2 Waterborne Diseases

Any kind of disease is a threat to the general health worldwide but when the disease is transferrable through a medium that people, animals and plants consume and use every day, this threat is increased exponentially. According to the World Health Organisation (2019), it is estimated that over 2 billion people worldwide utilise a water source that is contaminated with faeces (World Health Organisation, 2019) and according to the United Nations (2017), approximately 80% of wastewater is discharged into the environment without being adequately treated (United Nations, 2017) . Furthermore, Xagoraraki and O’Brien (2019) reported that between 1.5 and 2 million people pass away yearly due to waterborne diseases and these same diseases cause even more illnesses in people (Xagoraraki & O’Brien, 2019). Some of the most common waterborne diseases include the following:

1. Typhoid – caused by the Salmonella typhi S. paratyphi bacteria (Bhardwaj et al., 2019), is a disease that causes headaches, constipation, diarrhoea, high fevers and vomiting. The disease is believed to infect more than 20 million people worldwide and, in some cases, the disease can be fatal (World Health Organisation, 2018). This bacterium is transferrable through recreational water sources (Bhardwaj et al., 2019) such as lakes and coastal waters that are reused for recreational purposes such as swimming and fishing.
2. Escherichia Coli (E. coli) – generally harmless bacteria that live in the intestines of healthy people, but the disease is caused by specific strains such as E. coli 0157. This disease causes diarrhoea which may vary from mild to bloody, nausea, kidney failure and stomach cramping and tenderness (Bhardwaj et al., 2019). The number of people affected by this diseases worldwide is unknown because most people take this disease as a common illness so they don’t visit the hospital for it but the Centre of Disease Control (CDC) estimates that in the United States, approximately 265 thousand people are affected by the illness yearly (National Institude of Allergy and Infectious Diseases, 2012). If these are the estimates in a country as developed as the United States, only one can imagine how affected the rest of the world is, especially in developing countries. Thisbacteria is transferable through wastewater, drinking water, recreational water and even municipal water (Bhardwaj et al., 2019).
3. Hepatitis A – caused by the Hepatitis A virus is a disease that causes liver disease, diarrhoea, jaundice, malaise and nausea (Bhardwaj et al., 2019). According to the World Health Organisation (2015), there is an estimated number of 1.4 million people getting infected yearly (World Health Organisation, 2021). The virus is transmitted through marine waters, wastewater and recreational water sources (Bhardwaj et al., 2019).
4. Dysentery – caused by the bacteria Shigella spp., dysentery is a disease that causes mild to bloody diarrhoea, nausea, blood and mucus present in the stool and fevers (Bhardwaj et al., 2019). The disease is believed to cause 165 million cases yearly, with 1.1 million of them being fatalities in developing countries (Williams & Berkley., 2016). The disease is transmitted through recreational waters.
5. Giardiasis – caused by the Giardia duodenalis parasite, the disease causes diarrhoea, nausea, intestinal problems and stomach cramps and pain (Bhardwaj et al., 2019). The disease causes more than a million yearly cases according to the CDC (Centre for Disease Control and Prevention, 2021). The disease is transmitted through recreational water sources, municipal water sources and river streams (Bhardwaj et al., 2019).

### 2.2.3 Supervised Machine Learning Algorithms

According to Faggella (2020), machine learning is the study of teaching computers to learn and act like people, and to enhance their learning over time in an advanced manner by being independent, using relevant information fed to them in the form of observations and true interactions (Faggella, 2020). Additionally, these algorithms can provide an insight in to how quite difficult it is to solve some problems by learning different in surroundings. There are three types of machine learning which are supervised machine learning, unsupervised machine learning and semi-supervised machine learning (Lee & Shin, 2020). Supervised machine learning is mainly used for the purpose of classifying or predicting future situations since the way that it works is by creating general hypotheses from externally provided examples (Chinta, 2020). The machine learning algorithm is trained by giving it a dataset which is the input data and specifying the preferred output, followed by it comparing the previously specified preferred output and the computed output and minimising error rates by repetitive learning and readjusting the internal mapping functions’ values. This learning only stops when the accuracy levels of the results are satisfactory or if other cessation criteria are met (Lee & Shin, 2020). Some of the most commonly used supervised machine learning algorithms are Linear Regression, Logistic Regression, Naïve Bayes, Decision Trees (Nasteski, 2017), Random Forest, Support Vector Machines, Neural Networks (Chinta, 2020) and K-Nearest Neighbours (Lee & Shin, 2020). Machine learning has long been used many different areas and industries. An example of supervised machine learning in terms of classification is its use in the e-commerce industry to classify the sentiment of some form of text, such as a product review. This helps companies ascertain negative reviews from positive ones. Another example in terms of prediction is in protective maintenance, whereby a model would study equipment generated data which includes performance data as well as historical data to predict when a piece of equipment might need repair (Lee & Shin, 2020). Additionally, according to Uddin et al., supervised machine learning has been used for diseases prediction (Uddin et al., 2019), making it suitable for use in this project. Uddin et al. also found out that although Support Vector Machines are most commonly used for disease prediction in most papers, Random Forest shows the highest accuracy followed by Support Vector Machines (Uddin et al., 2019). This is going to be considered in the selection of the algorithm for this project as even though Random Forest shows the highest accuracy, most of the articles considered for the project were using to predict heart diseases and none of them tried predicting waterborne diseases, meaning this accuracy may not be as high for this project.

### 2.2.4 Support Vector Machines

Support Vector Machine is one of the most popular machine learning algorithms used to analyse both classification and regression problems and it is based on the supervised machine learning model (Pisner & Schnyer, 2020). Some real-world applications of Support Vector Machine include image classification, face detection and text categorisation (Gour, 2019). Additionally, according to Uddin et al. (2019), Support Vector Machines are some of the most reliable algorithms used for disease detection (Uddin et al., 2019). The goal of a Support Vector machine is to locate a hyperplane in a dimensional space of N. which is the number of features, which clearly classifies the data points (Gandhi, 2018). If the number of features is 2 then the hyperplane is a line but if the number of features is 3 then the hyperplane is a 2-dimensional plane.

Chart, scatter chart

Description automatically generated

*Figure 1: How the hyperplane will look depending on the number of features*

*Source: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47*

This hyperplane is meant to classify the data points by having the data points fall on either side of it. This plane is also meant to have the maximum margin possible, which is the maximum distance between data points of each class (Uddin et al., 2019). This helps future data to be predicted more efficiently.

*Chart, scatter chart

Description automatically generated*

Figure 2: Small Maximum Margin and Large Maximum Margin

*Source: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47*

The figure above shows an example of a small maximum margin and a large maximum margin.

Chart, scatter chart

Description automatically generated

Figure 3: Hard Margin (Left) and Soft Margin (Right) (Pisner & Schnyer, 2020)

The figure above shows a hard margin, which means that no training errors occurred at all whereas the soft margin allows a certain amount of training errors to occur (Pisner & Schnyer, 2020). This can be seen as a green dot is on the right side, where only red dots are meant to be and a red dot on the left side, where only green dots are meant to be.

### 2.2.5 Long Short-Term Memory

Long Short-Term Memory is an algorithm which is a variation of a Recurrent Neural Network, which helps solves the vanishing gradient problem of Recurrent Neural Networks. This problem occurs when the Recurrent Neural Network is going through its backpropagation through time phase whereby it struggles with short-term memory and the gradients used in the model become smaller as they update, which causes them to carry no significant value to the modelling being done (Arbel, 2018). The way that Long Short-Term Memory solves this is explained in the diagram below.

*Diagram

Description automatically generated*

Figure 4: Long Short-Term Memory Gates

*Source: https://aditi-mittal.medium.com/understanding-rnn-and-lstm-f7cdf6dfc14e*

As shown above, Long Short-Term Memory consists of three main gates which are the input gate, the output gate and the forget gate. The presence and activation of the forget gate is what solves the vanishing gradient problem as it frequently updates the network on whether it should discard the information or not in every phase of the learning process (Arbel, 2018). Aldhyani et al. (2020) adds to this by saying that it keeps reminding provided the input gate is closed and the forget gate is open (Aldhyani et al., 2020).

## 2.3 Similar Systems

A study performed by Geetha and Bansal (2020) used Decision Trees to predict water quality using different variables which are pH, temperature, dissolved oxygen, turbidity and total suspended solids. The results obtained from the confusion matrix drawn show that the accuracy of the model is 98.28% and when compared with other traditional methods of predicting water quality, which are the standard water quality index formula and parallel ID3, the decision tree showed a superior accuracy (Bansal & Geetha, 2020).

Another study performed by Ahmed et al. in 2019 showed that when using the 4 variables temperature, pH, turbidity and total dissolved solids to predict the water quality index, Gradient Boosting was found to be the most efficient algorithm, but once total dissolved solids was dropped as a variable, Polynomial Regression was found to be the most efficient. When it came to predicting the water quality criteria, multilayer perceptron was found to be the most efficient (Ahmed et al., 2019).

A previous study performed by Shafi et al in 2018 used four different machine learning algorithms, namely Support Vector Machine, Single Layer Neural Network, k-Nearest Neighbours and Deep Neural Networks to classify water quality based on the variables temperature, pH and turbidity. Their study found out that the Deep Neural Network outperformed the other algorithms with an accuracy of 93% (Shafi et al., 2018).

A study performed by Aldhyani et al. (2020) found out that for the prediction of WQI, the Nonlinear Autoregressive Neural Network performs slightly better than the Long Short-Term Memory algorithm. When it came to predicting the water quality criteria, the Support Vector Machine was found to perform the best when compared to k-Nearest Neighbours and Naïve Bayes with an accuracy of 97.01% (Aldhyani et al., 2020).

In a study done by Muharemi et al. in 2018, the results shown when comparing the performance of a Logistic Regression algorithm, a Support Vector Machine, Neural Network and a Linear Discriminant Analysis algorithm show that the Logistic Regression algorithm obtained the best results for its overall results, with the F1-score being computed to 0.584 (Muharemi et al., 2018).

Muharemi et al. then further researched these results using a time series-based dataset and they found out that the best results were obtained by using the Support Vector Machine as compared to the Logistic Regression, Linear Discriminant Analysis, Recurrent Neural Network, Deep Neural Network, Long Short-term Memory and Simple Neural Network with the F1-score being computed to 0.9891 (Muharemi et al., 2019). Moreover, they added that Support Vector Machine and Logistic Regression are less sensitive as compared to Deep Neural Network, Long Short-term Memory and Recurrent Neural Network (Muharemi et al., 2019).

Another study performed by Jalal and Ezzedine (2019) applied a Decision Tree and a Support Vector Machine based on the parameters pH, temperature, turbidity, E.coli, free residual chlorine, arsenic, nickel, calcium, magnesium, nitrate and intestinal enterococci. The results obtained show that the Support Vector Machine, specifically the Linear Support Vector Machine, is more accurate with an accuracy of 98% and a precision of approximately 96% (Jalal & Ezzedine, 2019).

The table below shows a summary of the similar systems discussed in this section.

*Table 2: Summary of Similar Systems*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Title** | **Authors** | **Machine Learning Algorithm** | **Evaluation Matrix** |
| **1** | A Machine Learning Approach towards Automatic Water Quality Monitoring  (Bansal & Geetha, 2020) | Geetha and Bansal | Decision Tree | Confusion Matrix, Accuracy  Decision tree had an accuracy of 98.28% |
| **2** | Efficient Water Quality Prediction Using Supervised Machine Learning (Ahmed et al., 2019) | Umair Ahmed, Rafia Mumtaz, Hirra Anwar, Asad A. Shah, Rabia Irfan, José García-Nieto | Gradient Boosting, Polynomial Regression, Multiple Linear Regression, Random Forest, Support Vector Machine, Ridge Regression, Lasso Regression, Elastic Net Regression, Multi-layer Perceptron, Gaussian Naïve Bayes, Logistic Regression, k-Nearest Neighbour, Decision Tree, Bagging Classifier, Stochastic Gradient Descent | Accuracy, Recall, F1-score, Precision, Mean Absolute Error, Mean Square Error, Root Mean Squared Error, R Squared Error  Gradient boosting was best for water quality index with MAE of 1.9642, MSE of 7.2011, RMSE of 2.6835 and RSE of 0.7485.  Multilayer perceptron had an accuracy of 0.8507, precision of 0.5659, recall of 0.5640, and F1 score of 0.5649 |
| **3** | Surface Water Pollution Detection using Internet of Things (Shafi et al., 2018) | Uferah Shafi, Rafia Mumtaz, Hirra Anwar, Ali Mustafa Qamar, Hamza Khurshid | Support Vector Machine, Single Layer Neural Network, k-Nearest Neighbours, Deep Neural Networks | Accuracy, Precision, Recall  Accuracy of 93%, precision of 94% and recall of 93% |
| **4** | Water Quality Prediction Using Artificial Intelligence Algorithms (Aldhyani et al., 2020) | Theyazn H. H Aldhyani, Mohammed Al-Yaari, Hasan Alkahtani, and Mashael Maashi | Long Short-term Memory, Nonlinear Autoregressive Neural Network, Support Vector Machine, k-Nearest Neighbours and Naïve Bayes | F1-score, Accuracy, Specificity, Sensitivity, Precision, Mean Square Error, Pearson’s Correlation, Coefficiant (R%)  LSTM R% = 96.17  SVM had accuracy of 97.01%, sensitivity of 99.23%, specificity of 97.78%, precision of 94.93%, F1-score of 98.54% |
| **5** | Approaches to Building a Detection Model for Water Quality: A Case Study (Muharemi et al., 2018) | Fitore Muharemi, Doina Logofătu, Christina Andersson, Florin Leon | Logistic Regression algorithm, Support Vector Machine, Neural Network, Linear Discriminant Analysis | Accuracy, Recall, Precision, F1-score  Logistic regression had the best results with an F1-score of 0.5842 |
| **6** | Machine learning approaches for anomaly detection of water quality on a real-world data set (Muharemi et al., 2019) | Fitore Muharemi, Doina Logofătu & Florin Leon | Support Vector Machine, Logistic Regression, Linear Discriminant Analysis, Recurrent Neural Network, Deep Neural Network, Long Short-Term Memory, Simple Neural Network | Precision, Recall, F1-score  SVM had a F1-score of 98.91% |
| **7** | Performance analysis of machine learning algorithms for water quality monitoring system (Jalal & Ezzedine, 2019) | Dziri Jalal, Tahar Ezzedine | Decision Tree, Support Vector Machine | Accuracy, Precision  Linear SVM had accuracy of 98% and precision of approximately 96%. |

## 2.4 Summary

In conclusion, this section of the research paper discusses the literature review of water quality monitoring for the prediction of waterborne diseases using machine learning. Firstly, the title is broken down into separate sub-sections in order to perform domain research on them and find out more about each area. The areas discussed were water quality monitoring, waterborne diseases, supervised machine learning and the specific machine learning algorithms that are going to be implemented in this project which are Support Vector Machine and Long Short-Term Memory. Next, some systems that are similar to what I am doing were discussed, with the parameters used in each study, the algorithm used, and the evaluation metrics used. It was found that most papers used Support Vector Machines for their systems and most of the evaluation metrics used ranged from a combination of precision, recall, accuracy and F1-score.

# CHAPTER 3: TECHNICAL RESEARCH

## **3.1 Programming language chosen**

When it comes to machine learning, one of the biggest obstacles that people run into when conducting a project is picking a suitable programming language. This is because there is no one best programming language that is best for a machine learning, meaning all of them have their pros and cons. To make this selection easier, there a few criteria that can be focused on when picking an appropriate programming language. According to Wu (2019), some criteria that need to be considered when picking a programming language are a support ecosystem, performance considerations of the project and whether the language will be able to meet those performance considerations, whether the language has libraries that can be used for the project and the flexibility of a language to name a few (Wu, 2019).

According to Springboard India (2020), Karczewski (2020) as well as Voskoglou (2017), some of the best programming languages to use are Java, Python and R (Springboard India, 2021) (Voskoglou, 2017) (Karczewski, 2020). A table comparing the three is described below.

*Table 3: Comparison of Java, Python and R*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Java** | **Python** | **R** |
| **Ease of Learning** | Easy to learn with a steady learning curve due to its easy syntax and efficiency | Easy to learn, especially when compared to other languages, with a steady learning curve due to its readability, easy syntax and efficiency | Steep learning curve due to its syntax being different than more traditional programming languages and time to learn code, power and flexibility |
| **Speed** | Fast | Fast | Slow compared to other programming languages |
| **Libraries** | Vast eco-system of libraries | Vast eco-system of libraries | Vast eco-system of libraries |
| **Cost** | Free | Free | Free |
| **Flexibility** | Highly flexible due to its support for object-oriented and functional programming as well as its ability to run a large number of platforms along with its scalability making it great for large projects (Springboard India, 2021) | Highly flexible due to its support for object-oriented, functional, procedural and imperative programming as well as its cast eco-system of libraries (Springboard India, 2021) | Highly flexible due to the ability to mix tools, usability of data analysis, data visualisation and data sampling to name a few (Springboard India, 2021) |
| **Machine Learning** | Libraries such as Java-ML, MALLET, RapidMiner and Weka allow for great machine learning applications (Slesar, 2021) | Libraries such as NLTK, NumPy, Sci-kit, Sci-kit-learn, and Tensorflow allow for great machine learning applications (Springboard India, 2021) | CARET, dplyr, PARTY and randomFOREST allow for great machine learning applications (Springboard India, 2021) |

Voskoglou concludes that in most cases, developers end up using the language they are using for most of their other projects for machine learning as well. She also finishes off by saying that Python is the best choice for first time experience with machine learning whereas Java is the best for business backgrounds (Voskoglou, 2017).

### 3.1.1 Python

Based on the table seen above, Python has been selected to be used for the implementation of the project. Python is an open-source high-level programming language which places emphasis on readable and brief codes so that understandability can be attained easily. Its straightforward syntax makes it easier for programmers to focus on what to write instead of how to write a piece of code (Springboard India, 2021). It is very flexible due to the fact that it supports object-oriented programming, functional programming, procedural and imperative programming so it can be used to create both small and large projects (Springboard India, 2021). According to Karczewski (2020), Python excels in two categories which are machine learning and web development (Karczewski, 2020). Additionally, it is widely adopted and is the primary programming language chosen for machine learning by many of the world’s most powerful companies in the IT field including Google, Instagram, YouTube, Uber and Amazon to name a few (Springboard India, 2021). The massive in-built library and package collection that is available for Python with a zero-learning curve helps increase throughput and reduce development and compilation time (Springboard India, 2021). Some of these libraries include NLTK which deals with text data and natural language processing as well as Librosa for audio data. When it comes to machine learning some suitable libraries include NumPy that deals with numerical data for the application of mathematical operations (Malik, 2019), Scikit-learn, which is arguably the most useful library for machine learning due to containing a lot of tools including classification, clustering and regression and Pandas, which is used for data analysis with tasks such as data cleaning, normalisation, visualisation and analysis (ActiveState, 2021)

## 3.2 IDE (Interactive Development Environment) chosen

An Integrated Development Environment, otherwise known as IDE, is an application or software that helps programmers by combining all the different writing parts and characteristics of writing or coding programme which includes a code editor, compiler and debugger all found in the same place (Gillis & Silverthorne, 2018). They also have features such as autocomplete and syntax highlighting (Codecademy, 2021). An IDE offers many benefits to programmers which include an increase in productivity due to it allowing them to quickly set up their programme and the tools offered in IDEs are standardised (Gillis & Silverthorne, 2018). Next, a single integrated development environment also offers most if not all of the required dev-test tools (Gillis & Silverthorne, 2018).

Since the programming language used is Python, the IDE to be chosen has to be one that can use Python for coding. Some of the most commonly used IDEs for Python include Spyder, Google Colaboratory, Atom and Jupyter Notebook which are all suitable for a machine learning project (Sharma R. , 2021). Depending on the algorithms chosen as well as whether the IDE can handle the algorithm or not, two IDEs have been selected for the development of this project. These are Google Colaboratory and Spyder.

According to Vasconcellos (2018), Spyder includes core data science libraries such as NumPy, SciPy and Matplotlib, as well as the ability to add plugins. (Vasconcellos, 2018). According to Sharma (2021), Spyder uses an interactive code execution style that allows you to compile a single line of code, a segment of code, or the entire code in one go (Sharma R. , 2021). Lastly, with the static code analysis function in Spyder, you may find superfluous variables, mistakes, and syntax issues in your code without even compiling it (Sharma R. , 2021).

On the other hand, Google Colaboratory provides a number of benefits of using over other IDEs. The first is that it is an online IDE from Google and so anything that is done on it is saved on the Cloud rather than on your local disk. Next, it also allows users to benefit from free dedicated GPU and TPU usage which accelerate projects no matter the size (Yalçın, 2020).

In summary, the IDE to be used for the project will depend on the algorithms chosen. If they are heavy on the computer and I find that I need the free GPU and TPU, I will opt for Google Colaboratory but they are light and my computer can handle them then I will use Spyder.

## 3.3 Libraries/Tools chosen

In programming terms, a library is a collection of reusable code, and it often describes a collection of core modules. This reusable code can be anything from a single simple function or collection of functions, data structures, variables to complete classes. Libraries are often used to reduce development time as well as save resources. On Spyder and Colab, these libraries can easily be imported using the “import()” function.

Since the project being implemented is a machine learning project, the following libraries are going to be utilised.

*Table 4: Libraries Used*

|  |  |
| --- | --- |
| **Library/Tool** | **Justification** |
| Scikit-learn | Generally considered to be the best library for machine learning, it offers various machine learning algorithms such as clustering, classification, regression and decision trees (Scikit-learn, 2021) |
| Pandas | Used for various tasks when working with data including cleaning, normalisation and visualisation (ActiveState, 2021) |
| Tensorflow | Used to load and pre-process data, build, train and reuse models (TensorFlow, 2021) |
| Flask | Web framework used to make the creation of web applications on Python easier (Dyouri, 2020) |
| Tkinter | The most universal way of creating a GUI on Python (GeeksforGeeks, 2020) |
| Keras | API framework for creating and assessing deep learning models (Keras, 2021) |
| Pickle | Used to export the best model that will be used in the deployment stage |

## 3.4 Operating System chosen

The operating system (OS) is the most important software that runs on a computer. Its role is to control the computer’s memory, processes, hardware and software as well as providing resources for tasks to run successfully. The operating system that will be used for this project is the latest version of Windows 10.

### 3.4.1 Hardware Specification

Table : Hardware Specification

|  |  |
| --- | --- |
| **Laptop** | HP Pavilion Gaming Laptop 15 |
| **Processor** | Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.21 GHz |
| **RAM** | 8.00 GB (7.89 GB usable) |
| **System Type** | 64-bit operating system, x64-based processor |
| **Input** | Keyboard and Mouse |
| **Storage** | Windows SSD 500 GB, 1 TB HDD |

### 3.4.2 Software Specification

Table : Software Specification

|  |  |
| --- | --- |
| **Software** | **Justification** |
| Spyder | IDE that will be used for the implementation of the project and the coding of the programme |
| Microsoft Visio | Creation of the Gantt chart for the project timeline |
| Microsoft Word | Documentation purposes |
| Microsoft Excel | Dataset is saved in the form of .csv |

## 3.5 Summary

In short, this section of the research paper discusses the programming language chosen for the implementation of the project. This was reached by comparing some of the most commonly used programming languages for machine learning, which are Java, Python and R. These programming languages were compared in terms of ease of learning, speed, libraries, cost, flexibility and their performance in machine learning. Among these three, it was concluded that Python is going to be used for the execution of the project. In order to programme using a programming language, and IDE is necessary and so, the next part of this chapter discusses the possible IDEs that are going to be used for the project. Ultimately, PyCharm and Google Colaboratory were selected as the potential candidates for the IDE, with the final selection being dependent on the algorithm chosen and how heavy it is. Next, a discussion of the possible libraries and tools to be used was carried out and finally, the hardware and software specification was carried out to determine the type of device to be used to implement this project.

# CHAPTER 4: METHODOLOGY

## 4.1 Introduction

Selecting the right project methodology is one of the most important steps in conducting a project. This is because they provide a consistent framework to define and manage important project-related aspects such as resources, timelines, team members and money. They also help identify, control and reduce project risks (Gil, 2015). Due to its importance, choosing the right project management methodology is not a random process and it takes many factors into consideration. Some of these factors include flexibility to change in terms of the requirements and time, the cost, the project scalability, the difficulty of the project and the required skills (Smartsheet, 2021). In terms of data mining, there are a few popular methodologies. These are KDD (Knowledge Discovery in Databases), SEMMA (Sample, Explore, Modify, Model, Access) and CRISP-DM (Cross-Industry Standard Process for Data Mining) (Quantum, 2019).

KDD is the process of extracting what is regarded as knowledge through the utilization of data mining approaches according to the specification of measurements and thresholds and applying them on a database in order to uncover hidden patterns (Shafique & Qaiser, 2014). The stages of KDD are shown in the figure below.

Diagram

Description automatically generated

Figure 5: KDD Methodology

*Source:* [*http://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1\_kdd.html*](http://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1_kdd.html)

The first stage is selection, whereby either a target dataset is created, or a selection of variables are concentrated on which exploration is to be carried out (Quantum, 2019). The next stage is pre-processing, whereby cleaning and data pre-processing techniques are utilised to obtain data that is consistent without any noise. Transformation then takes place whereby the data is reduced or transformed so that the data is easier to handle (Quantum, 2019). Data mining then takes place whereby an appropriate data mining task is selected to be used in the project. This selection depends on the data mining goal and based on this, a task such as classification, clustering or regression to name a few, is selected to be applied. Once the mined patterns have been obtained, the last stage is the evaluation whereby the patterns are interpreted. This step may include visualisation (Shafique & Qaiser, 2014).

SEMMA is a data mining methodology developed by the SAS Institute which enables and facilitates the comprehension, organisation, innovation and maintenance of data mining projects (Shafique & Qaiser, 2014). The stages can be seen in the figure below.

Diagram

Description automatically generated

Figure 6: SEMMA Methodology

*Source:* [*https://www.linkedin.com/pulse/introduction-semma-data-mining-using-sas-enterprise-abbasaliyeva/*](https://www.linkedin.com/pulse/introduction-semma-data-mining-using-sas-enterprise-abbasaliyeva/)

The first stage is sampling, whereby a sample of the data is extracted by obtaining a fraction of a dataset that is adequate in size enough to contain relevant information while also being small enough to manipulate quickly (Quantum, 2019). Next, an understanding of the data is attained in the exploration stage to understand what type of data is being dealt with as well as if there are any abnormalities in the sample. Relationships between the data are studied and through this, a more in-depth understanding is achieved. Next, the data is modified, and in this stage, variable are selected, transformed and cleaned. Outliers are also studied and analysed here. Depending on the data mining goal, modelling of the data takes place and the type of model used is selected. Lastly, once the model has been obtained, it is then it is then assessed for performance by seeing how practical and reliable it is (Shafique & Qaiser, 2014).

CRISP-DM is the most popular data mining methodology according to Saltz (2020), which has become a standard way of applying data mining projects (Saltz, 2020). It is a methodology that helps plan, manage and apply data mining projects. The stages can be seen in the figure below.

Diagram

Description automatically generated

Figure 7: CRISP-DM Methodology

*Source:* [*https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=dm-crisp-help-overview*](https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=dm-crisp-help-overview)

The first stage is the business understanding, whereby an understanding of the project requirements and objectives takes place. Next, the data understanding takes place whereby the data that is selected is studied and analysed through exploration and description. The data preparation then takes place whereby a sample is selected and cleaned along with other pre-processing techniques being applied. Modelling then follows whereby modelling techniques are selected based on the project objective, the data is split into training and testing sets and the model is built. Once the model has been built, it is then assessed to see if it meets the success criteria. If that does happen to be the case, the model is then deployed using a deployment plan, and the project is reviewed.

While all three of these methodologies are suitable for data mining projects, the most suitable one for this type of project is going to be CRISP-DM. CRISP-DM is one of two methodologies which allows recursion from any step within the methodology, with KDD being the other one. More importantly, CRISP-DM was selected because it is the only methodology which allows the deployment of the model, which is a necessary step for the completion of this final year project.

## 4.2 CRISP-DM Methodology

This section of the document will discuss the methodology being applied in more detail. As mentioned before, the methodology that is going to be used is CRISP-DM. This methodology consists of 6 main steps which are shown in the figure below.

Diagram

Description automatically generated

Figure 8: CRISP-DM Methodology

*Source:* [*https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=dm-crisp-help-overview*](https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=dm-crisp-help-overview)

**Phase 1: Business Understanding**

The first stage of the CRISP-DM methodology is the business understanding phase. In this phase, a comprehension of what the business would like to achieve through the implementation of this project. This is done by understanding the objectives of the business, the project and the requirements along with assessing the current state of the project (Smart Vision Europe, 2021). An understanding of the problem has already been attained as seen in section 1.2 which resulted in a problem statement being generated. The background of the project problem has also been performed as seen in section 1.1. the objectives of this project have also already been reached as seen in section 1.6.2. Along with this, a project plan has also been developed as seen in section 1.8.

**Phase 2: Data Understanding**

The next phase of the methodology is the data understanding phase. This phase involves obtaining the dataset upon which the project will be carried on and exploring this dataset in order to familiarise more with it. The data is then described in terms of the quantity, the type of data being dealt with and what each column in the dataset is concerned with (Smart Vision Europe, 2021). Relationships in the data are explored and variables which may pass as target variables are identified. Next, the quality of the data is measured by identifying missing data or wrong data (Smart Vision Europe, 2021). The dataset has already been obtained which consists of data about pollutants measured in certain water sources in India. The dataset that I am going to be using contains approximately 2000 rows, 1991 to be more specific which contains information about the water quality including the water temperature which is of interval data type, its pH which is of ordinal data type, dissolved oxygen in the water which is of interval data type, conductivity which is of interval data type, biological oxygen demand which is of interval data type, faecal coliform which is of interval data type and Nitratenan N+ which is of interval data type. Once explored, the dataset seemed to contain noise. Some entries contained “NAN”, which stands for Not a Number. This could imply that these entries are incorrect. Upon initial inspection, it does not look like the dataset contains a variable that could be used as a target variable and therefore, a target variable is going to be created based on some factors. This target variable is going to determine whether or not the water source contains contaminated water or not, so it is going to be of Boolean type.

**Phase 3: Data Preparation**

Once a business understanding has been obtained, the next phase in the methodology is the data preparation stage. In this stage, a sample is selected from the dataset based on a few factors such as the number of rows in the dataset and the quality of the data. Next, the data is cleaned off of any noise through the use of pre-processing techniques (Smart Vision Europe, 2021). This also consists of replacing empty values or completely removing them. New variables may also be derived from current variables. An example of this could be calculating the area from the available variables which are length and width. Data is also integrated if there are multiple datasets or data obtained from multiple sources. Since the size of this dataset is already small, the entire dataset will be treated as the sample size. Based on what was mentioned, I will be cleaning the dataset of noise such as the “NAN”. According to Păpăluță (2020), the best way to deal with “NAN” entries in a dataset is either by completely removing them or by inputting a value in place of them (Păpăluță, 2020). This value could be obtained by calculating the mean or the median of the rest of the column and imputing the result in place of “NAN”. As mentioned before, the target variable may be constructed based on the values in the dataset. Since I do not have multiple datasets, no data integration will take place.

**Phase 4: Modelling**

Once the data is prepared, the data is then ready to be modelled. The modelling technique is selected based on the project objective. In this case, the project objective is to be able to predict and therefore, the modelling technique to be applied is going to be a predictive model. The two models that are going to be used are Support Vector Machines and Long Short-Term Memory algorithms. These will be applied for machine learning as well as deep learning for this research.

The dataset is also going to be split into a training set and a testing set. The split is going to be 70% for the training set and 30% for the testing set. After that, the data is passed through the model and the model is built and assessed.

**Phase 5: Evaluation**

Once the model has been built and assessed, the next step is to evaluate the model. Before that, a confusion matrix will be derived.

Table

Description automatically generated

Figure 9: Confusion Matrix

*Source: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62*

TP stands for true positive, which means that the predicted output is positive and the result is also positive whereas FP stands for false positive, which means the predicted output is positive and the result is negative. FN stands for false negative which means the predicted output is positive and the actual result is positive whereas TN stands for true negative, which means the predicted output is negative and the actual result is negative (Narkhede, 2018). The model is going to be assessed based on its recall, accuracy, precision and F1-score. These are going to be calculated based on the formulas shown below.

**Recall** =

**Accuracy** =

**Precision** =

**F1-score** = 2 =

Recall is the percentage of correctly predicted positive observations to the total number of observations in the actual class whereas accuracy is percentage of correctly predicted positive observations to the total number of observations (Shung, 2018). Precision is the accurately predicted positive observations as a percentage of all the expected positive observations (Shung, 2018). Lastly, F1-score is the weighted average of recall and precision which helps in evening out unbalanced classes (Shung, 2018).

Besides that, the results of the models are also going to be assessed based on if they meet the project goals and objectives mentioned previously.

**Phase 6: Deployment**

Once the model has been fully approved of, the model will be deployed using Spyder and Streamlit. A monitoring and maintenance plan will also be set in place in case anything happens to the model both during the deployment and after it has been deployed. Lastly, a report summarising the results, the success criteria and the actual deliverables of the project will be generated.

### 4.2.1 Process Flow Overview

Diagram

Description automatically generated

Figure 10: Process Flow Overview

The figure above shows the flow of the system and the project to provide more insight in to how the final system is going to be derived. Firstly, the dataset will be obtained from Kaggle and then pre-processing techniques such as cleaning, normalisation and construction will be applied onto it. Next, feature selection will be performed to select only the important variables to be used for the project. The dataset will then be split into a training and testing set in a 70% to 30% ratio. Once the dataset is trained and tested, it will then be passed through the predictive modelling algorithms and model comparison will take place to select the best model. Lastly, the model is going to be deployed using suitable techniques.

## 4.3 Summary

In summary, there are three possible methodologies that could be used for a data mining project which are SEMMA, KDD and CRISP-DM. Among these three, it was found out that the most suitable methodology to be used is CRISP-DM due to its recursive nature as well as its deployment stage, which is necessary for the completion of this project.

Under CRISP-DM, there are 6 stages. The first stage is business understanding which entails investigating the project objectives, deliverables and problem statement. The second stage, which is data understanding, involves obtaining the dataset and exploring it to identify hidden relationships between the different variables. Once the dataset has been analysed, the next stage is the data preparation stage, which involves pre-processing the dataset to get it ready for the algorithm chosen. Furthermore, new variables may be created in this stage and the data is then cleaned to remove noise which include empty records and duplicate records. Some records were identified to have the word “NAN” in them, which stands for Not a Number. The way this is going to be dealt with is either through imputation using the mean of the column or the median of the column. Next, the algorithm chosen will be modelled. These algorithms are going to be the Support Vector Machines and Long Short-Term Memory algorithms. Once the models have been constructed, the next stage is to evaluate the results obtained and this is going to be done using the recall, precision, accuracy and F1-score. Once the model has reached a level of satisfaction, it is then going to be deployed using a Python GUI.

# CHAPTER FIVE: DATA ANALYSIS

## 5.1 Introduction

The dataset obtained was from Kaggle, containing historical data about various pollutants measured from Indian water sources across periods of time. The dataset mainly contains historical data from 2003 to 2014. The dataset is a structured one, containing labelled rows and columns making the machine learning techniques chosen suitable. Initially, the necessary libraries needed to perform this project are loaded as shown in the figure below.

Text

Description automatically generated

Figure 11: Libraries Imported for Usage

As can be seen above, the libraries are separated based on what they are going to be used for in the project.

## 5.2 Data Exploration

Data exploration comprises mostly of tasks that aid data scientists in better understanding their data by providing insights into the phenomena being modelled, evaluating the data's quality, and locating or creating characteristics that increase model accuracy. Creating graphs and charts, estimating statistics, altering data, detecting anomalies, and so on are all examples of data exploration (Rojas et al., 2017).

First of all, once the data is imported using the pandas library, the data types of each column in the dataset are explored. The output is as shown below.

Table

Description automatically generated with medium confidence

Figure 12: Column data types before conversion

As seen above, all except for one column, that column being the “year” column, are of the object type. This comes even though columns such as “Temp” and “PH” should also be numerical. Initially, these column names are changed to match the data contained them better and to also make handling them easier. As such, “LOCATIONS” was changed to “Address”, “STATE” was changed to “Location”, “D.O. (mg/l)” was changed to “DO”, “CONDUCTIVITY (µmhos/cm)” was changed to “Conductivity”, “NITRATENAN N+ NITRITENANN (mg/l)” was changed to “NI”, “FECAL COLIFORM (MPN/100ml)” was changed to “Fec\_col” and “TOTAL COLIFORM (MPN/100ml)Mean” was changed to “Tot\_col”.

## 5.3 Data Cleaning

### 5.3.1 Data Conversion

The first step carried out in cleaning the data is to convert all of the columns containing numerical data in them to numerical type. The way that this was done is by creating a function which traverses through each column and converts the data inside of it to float. This is done using the function below.

Text

Description automatically generated

Figure 13: Function used to convert data to float data type

As can be seen from the figure above, the conversion of data occurs starting from the third column and it ends at the column before the last. Inside each column, each value is converted into float. The dataset is passed through this function and to check if it worked or not, the data types are checked again with the output being shown below.

Table

Description automatically generated

Figure 14: Column data types after conversion

Once the conversion is complete, the next step is to check for NULL values inside of the dataset in order to deal with them in various ways which will be discussed below.

### 5.3.2 Dealing with NULL Values

The number of NULL values in each column is first computed. The results are displayed below.

Table

Description automatically generated with medium confidence

Figure 15: Number of NULL values in each column

Based on the fact that some columns are categorical, the function above is not going to be able to calculate the number of NULL values inside of them since they are set as string instead of true NULL values. Due to this, the following code is used to convert these NULL values to actual NAN values.

Graphical user interface, text

Description automatically generated

Figure 16: Conversion to actual NAN values

The function above traverses through the entire dataset and replaces any instance of “NAN” with “np.nan”, which is a true NAN value. Once this is done, the number of NULL values can be recalculated for all the columns.

Text

Description automatically generated

Figure 17: Number of NULL values in each column

The figure above shows how many NULL values are in each column in the dataset in order of least to most. The NULL value treatment is going to differ for each column based on the data type contained in it. For numerical columns, the median of the column is taken and used to impute and replace the NULL values. Therefore, the dataset will be split into based on its numerical columns and its categorical columns.

Text

Description automatically generated

Figure 18: Median imputation for NULL values in numerical columns

A different variable, “df\_num” is used to store only the numerical columns in the dataset. The “sklearn” library is then used to obtain the “SimpleImputer”, which takes in a strategy used to impute NULL values into columns. This imputer is used to impute the median of each column in place of the NULL values.

The NULL value imputation for the categorical values is going to be carried out in a different way since imputations using mean or median would not work because of the data type. Therefore, the NULL values are going to be filled by using the relationship between the three categorical values.

Table

Description automatically generated with medium confidence

Figure 19: Figuring out relationships

As can be seen from the figure above, it can be seen that all locations with the station code “1330” are in the location “TAMILNADU”. This can be then used to impute all the NULL values with that station code with “TAMILNADU”, which is done using the code snippet below. Additionally, some “Location” entries can be seen to contain the full address which should be placed in the “Address” column and that the address contains the location at the end.

Graphical user interface

Description automatically generated with medium confidence

Figure 20: Imputing NULL "Location" entries with "TAMILNADU"

Based on this, it is safe to assume that there is a relationship between the three categorical columns. Therefore, a function is going to be created for each column to impute the NULL values. These 3 functions with their outputs can be seen below.

Table

Description automatically generated with low confidence

Figure 21: Function used to impute NULL values in the "Address" column

Table

Description automatically generated with low confidence

Figure 22: Function used to impute NULL values in the "STATION CODE" column

A picture containing text

Description automatically generated

Figure 23: Function used to fill in NULL values in the "Location" column

Once all three functions are ran, the number of NULL values in the columns is re-checked to ensure that all imputations have taken place successfully.

Graphical user interface, text, application

Description automatically generated

Figure 24: Re-checking NULL values after imputation

As can be seen from the figure above, 2 NULL values remain in the “STATION CODE” column and 10 NULL values remain in the “Location” column. These NULL values are investigated using the code below.

Table

Description automatically generated

Figure 25: Remaining NULL values after imputation

Based on the assumption that was previously made whereby there are relationships between the three columns, each station code is going to be investigated to check if any imputations may have been missed using the previous functions.

Graphical user interface

Description automatically generated

Figure 26: Further investigating the NULL values

Once again, it can be seen that location can be seen at the end of some of the addresses shown above. Therefore, manual imputation of the location will take place using the code below.

Text

Description automatically generated

Figure 27: Manual imputation of NULL values in the "Location" column

Another checking of NULL values for the entire dataset takes place and any rows still containing NULL values will be removed since no other attribtutes can be used for imputation and a final checking of NULL values then takes place to ensure that no more NULL values are in the dataset.

Graphical user interface, text, application

Description automatically generated

Figure 28: Checking and removing rows containing NULL values

Table

Description automatically generated with medium confidence

Figure 29:Final checking of NULL values

### 5.3.3 Removing Outliers

Graphical user interface

Description automatically generatedChart, histogram

Description automatically generatedNext, the numerical columns were plotted using kerndel density estimation (KDE) to find out the probability density of a continuous variable at various levels (Pandas Development Team, 2021). The results obtained are displayed below.

Graphical user interface, histogram

Description automatically generatedGraphical user interface

Description automatically generated

Chart, line chart

Description automatically generated

Figure 30: KDE plots for numerical columns

Seeing as to how most of the graphs shown have a Gaussian-like curve, the next step would be to try and remove some of the outliers using z-score. Firstly, the z-score for each row is calculated using the “zscore” function. These z-score are the imputed into the dataset as an extra column. Next, outliers are removed by detecting outliers with a score of more than 3. KDE plots are once again plotted to see if there is any difference or not.

Chart, histogram

Description automatically generatedChart, line chart, histogram

Description automatically generated

Graphical user interface

Description automatically generatedChart

Description automatically generated with medium confidenceChart, line chart

Description automatically generated

Figure 31: KDE plots after zscore

As can be seen from the above plots, minor improvements to the shape of the graph can be seen across all of the plots above.

### 5.3.4 WQI Calculation

Next, the Water Quality Index is calculated for the dataset to determine which water sources are safe for usage or not. The goal of a WQI is to convert sample constituents and concentrations from other assessment methods into a single value for the water quality state of a water resource. WQIs are a simple and effective instrument for assessing water quality in bodies of water, allowing for the examination of water quality (Noori et al., 2019). In order to calculate the WQI for the dataset, several steps come into play. First, the standard value for each parameter being used is taken into consideration. The parameters that are going to be used are dissolved oxygen, pH, conductivity, biological oxygen demand, nitrate, decal coliform and total coliform. The standard value for dissolved oxygen is 10, pH is 8.5, conductivity is 1000, biological oxygen demand is 5, nitrate is 45, fecal coliform is 100 and total coliform is 100 based on the Bureau of Indian Standards (Bureau of Indian Standards, 2012). Next, the ideal value for each parameter is taken into consideration. These are as follows; 14.6 for dissolved oxygen, 7 for pH, conductivity is 0, biological oxygen demand is 0, nitrate is 0, fecal coliform is 0 and total coliform is 0. The next step is to then find the weights of each parameter being used for the calculation of WQI. Based on (Feather, 2020), the weight , Wi, of a parameter is calculated using the formula *Wi = K / Si*, where K is a constant calculated using the formula *K = 1 / ∑ (1/Si)*. Based on this, the weight of each parameter is calculated and the results are shown below.

Graphical user interface, text, application

Description automatically generated

Figure 32: Unit weight for each parameter

Using all of these values, the WQI is calculated using the formula *WQI = ∑ qi × wi*, where qi is calculated using the formula  *qi = 100 × ( Vi / Si)*. Based on these formulas, the WQI is calculated for the entire dataset. Firstly, the columns “temp” and “year” are dropped from the dataset since they are not used in the calculations. Next, the following code snippet, which iterates through all of the rows, is used to calculate the WQI.

Text

Description automatically generated

Figure 33: WQI calculation for the dataset

As can be seen from the last block of text, the WQI is calculated and the values are inserted in a separate column. This column is then inserted into the dataset. The next thing that is done is the removal of noise. This is done by checking if there are any WQI values below 0 as these would be considered incorrect. Also, any WQI values above 100 are removed as these are considered to be too polluted (Kachroud et al., 2019). This is done using the following code.

Graphical user interface

Description automatically generated

Figure 34: Checking and removal of incorrect WQI values

As can be seen from the code snippet above, no negative WQI values can be seen. However, 403 instances of WQI values above 100 are present. These are therefore removed from the dataset. Once the WQI has been obtained, a classification for the WQI can then be made. According to (Krishan et al., 2016), water can be classified into five main categories which are very polluted, ranging from 0-24, poor, ranging from 24-49, fair which ranges from 50-74, good which ranges from 75-94 and excellent which ranges from 95-100. For the sake of making this project simpler, there will be four categories, whereby good and excellent will be combined. This classification can be seen in the code snippet below.

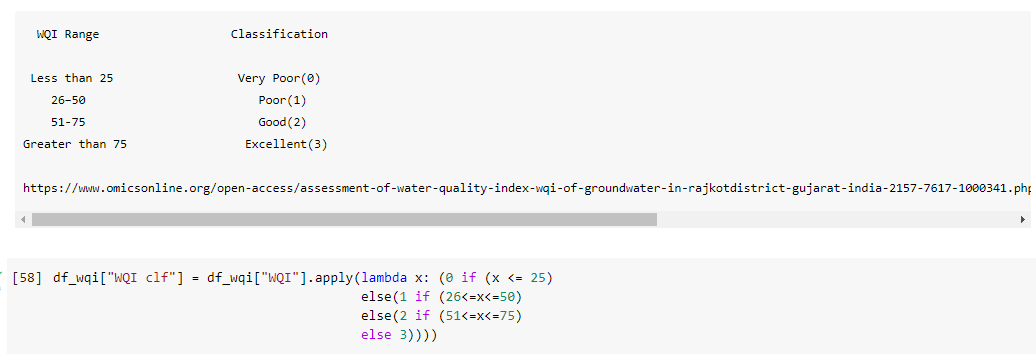


Figure 35: WQI classification

Based on this, a new column, “WQI clf” is created and added to the dataset. The target variable that is going to be created is based on this “WQI clf”. To make the target variable, which will be called “Is\_Potable” meaning safe drinking water, easier to predict, it will be made binary whereby 0 means that the water source is unsafe and 1 means that the water source is safe. In order for a water source to be considered safe, it needs to have a “WQI clf” of either 2 or 3 and anything less than that is considered to be unsafe. The target variable is therefore created and added to the dataset.

### 5.3.5 Data Transformation

Graphical user interface, application

Description automatically generated

Figure 36: Complete dataset description

The figure above shows the complete dataset being described. As can be seen above, the data is quite unbalanced as it is still all over the place. This will be addressed shortly.

Chart, pie chart

Description automatically generated

Figure 37: Target variable graph

The figure above shows the distribution of the target variable in the dataset. As can be seen, the data is quite unbalanced. This will be addressed shortly.

As mentioned before, the data in the dataset is unbalanced. Therefore, standardisation procedure will be undergone in order to balance the dataset out. The standardisation method that will be used will be “MinMaxScaler” which works by translating all values into the range 0 and 1, which means that their minimum and maximum values will be 0 and 1, respectively. (Loukas, 2020). The data that will be standardised is only the input variables from the dataset. Therefore, all of the other columns in the dataset are dropped. This is done using the code snippet below.

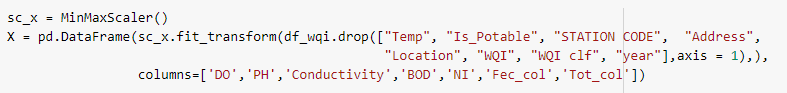


Figure 38: Data standardisation

The standardised is then explored to make sure that it is suitable for usage. This is done using the “describe()” function and by drawing boxplots of all of the input variables as seen below.

A picture containing box and whisker chart

Description automatically generated

Figure 39: Input variables described and plotted

Despite there being a few outliers in the standardised data, they will be kept and this data will be used as they are mainly present in only two columns, which are “PH” and “Conductivity”.

## 5.4 Data Modelling

Once the data has been fully cleaned and transformed, the next step is to build the models. The models selected for this project are Support Vector Machine (SVM) and Long-Short Term Memory (LSTM).

### 5.4.1 Data Splitting

A picture containing graphical user interface

Description automatically generated

Figure 40: Splitting dataset

The figure above shows the code used to split the dataset into the training and test sets. The data is split based on a 70-30 split. The split is then assigned to the respective X and y values, whereby X represents the 7 input variables and y represents the target variable “Is\_Potable”. As mentioned previously, the data was quite unbalanced. Additionally, the number of samples is quite low. Therefore, oversampling was used to resolves these problems using the training sets as seen below.

Text

Description automatically generated

Figure 41: Data oversampling

### 5.4.2 Basic Classifier Training

As mentioned previously, the models used were SVM and LSTM, as they can both be used for classification projects.

Text

Description automatically generated

Figure 42: Non-optimised SVM model

First, SVM is called from the scikit library. The model is then fitted using the training sets before it is used to predict the testing sets. The results of the model are obtained using the “classification\_report()” function.

Text

Description automatically generated

Figure 43: LSTM model

In the beginning, the “Sequential()” function is used to initialise the neural network. Next, an embedding occurs which is used to convert positive indices into fixed-size dense vectors (TensorFlow, 2021). Then, an LSTM layer is added along with a dropout layer to prevent overfitting in the model and a dense layer which outputs a single value prediction. The activation used on the LSTM model is sigmoid since the target variable is binary (Sharma S. , 2017). The model is then compiled with the loss set to “binary\_crossentropy” as it is stuitable for classification problems of binary target variables and the optimizer is set to “adam” with the evaluation metrics set to “accuracy”. Lastly, the model is fit using “x\_smote” and “y\_smote” as input, the epochs set to 150 and the validation set to “X\_test” and “y\_test”.

Text

Description automatically generated with medium confidence

Text

Description automatically generated

Figure 44: LSTM model prediction

The model is then used to predict the values of “X\_test” and the values that are greater than 0.5 are saved as these are considered the right values. The model is then evaluated using mean squared error, r-square and the classification report.

### 5.4.3 Optimised Models

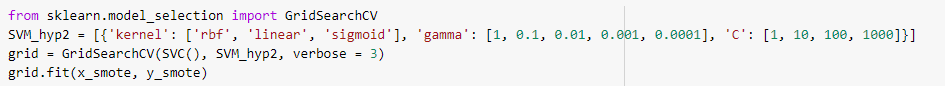




Figure 45: Optimised SVM model building

In order to add additional value to the project, 3 optimised SVM models were built to try and increase the accuracy of the models. The optimised models also use GridSearchCV which is the process of determining the ideal hyperparameter values for a particular model by performing hyperparameter tuning (Mujtaba, 2020). The hyper parameters used are kernel, gamma and C. GridSearchCV uses the Cross-Validation method to test all possible combinations of the values supplied in the dictionary and assesses the model for each one. As a result, after using this function, we can retrieve the accuracy/loss for each combination of hyperparameters and choose the one that performs the best (Mujtaba, 2020). For each misclassified data point, the C parameter applies a penalty. When c is small, the penalty for misclassified points is minimal, hence a large-margin decision boundary is chosen at the expense of a higher number of misclassifications. When c is big, SVM seeks to reduce the number of misclassified cases by imposing a high penalty, resulting in a decision boundary with a smaller margin (Yıldırım, 2020). The RBF's gamma parameter controls the influence distance of a single training point. Low gamma values suggest a broad similarity radius, resulting in the grouping of more points (Yıldırım, 2020). To be considered in the same group as points with high gamma values, the points must be quite close to each other (or class). As a result, models with high gamma values are more likely to overfit (Yıldırım, 2020).

Next, the best hyperparameter values are printed and these are used to build the actual optimised SVM model.

Text, letter

Description automatically generated

Figure 46: RBF SVM model building

The final optimised model is built with the best hyperparameter values and the model is fit and the classification report is printed.

## 5.5 Model Evaluation

### 5.5.1 SVM Models

Table

Description automatically generated

Figure 47: Non-optimised SVM model evaluation

The figure above shows the evaluation for the non-optimised SVM model that wasbuilt. The initial accuracy of the non-optimised SVM model is 91.02%.



Text, table

Description automatically generated

Figure 48: Optimised SVM model evaluation

The figure above shows the evaluation for the optimised SVM model that were built. The most optimal hyperparameter values for the optimised SVM model are the “rbf” kernel, 1 for the C value and 1 for gamma. Based on this hyperparameter tuning, the accuracy of the optimised model increased to 93.62%, showing an increase of 2.6%.

### 5.5.2 LSTM Model

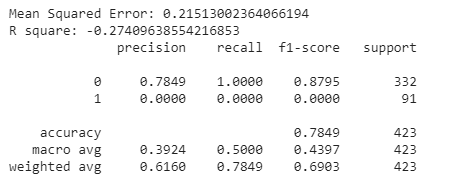


Figure 49: LSTM model evaluation

The figure above shows the evaluation of the LSTM model that was built. The model accuracy came to 78.49%. This means that the best model that was built is the optimised SVM model. Therefore, this model was exported and saved using the pickle library using the code below, and deployed on Python IDE.

A picture containing logo

Description automatically generated

Figure 50: Exporting optimised SVM model

## 5.6 Model Deployment

For the model deployment, the model is imported into the Spyder IDE and the libraries used are imported using the code snippet below.

A picture containing text

Description automatically generated

Figure 51: Model and library importing

Text

Description automatically generated

Figure 52: Obtaining input values

The above code snippet is used to collect the input values from the user for the 7 input variables used for the model. Each value is collected and passed to the function “classify\_water()” which is used to classify if the water is safe or not. If the water is safe, an output of 1 will be seen and a message informing the user that the water is safe will be output. Otherwise, the waer is unsafe and that is output for the user.

Text

Description automatically generated

Figure 53: Classify\_water() function

The code snippet above shows the functionality of the “classify\_water()” function. The values obtained from the user are converted to float data type to account for any decimal points. Next, instead of using MinMaxScaler since the formula makes use of the minimum value and maximum value of the entire column in the dataset. When a user inputs values to be checked for classification, the only value in the array is the value that the user inputted, meaning the minimum and maximum values will be the same as the value that the user input. Therefore, the formula used for the MinMaxScaler was instead used, with the calculation of the formula performed through hard coding. Once the standardisation happens, the values are then passed to the model to classify the water source and the classification value is returned. The model is then deployed on Streamlit.

A screenshot of a computer

Description automatically generated with medium confidence

Figure 54: Model deployment on localhost

The figure above shows how the model looks on localhost. The user is asked for the 7 input variable values and once they press the “Classify” button, the model will classify the water source safety.

## 5.7 Summary

To sum everything up, a total of 3 models were built. The first was a non-optimised support vector machine and an optimised support vector machine was also trained. Besides that, a single-layered long-short term memory model was also built. These models were all evaluated on accuracy, F1-score, recall and precision.

The best model can then be taken and trained more using mre recent data in order to classify water sources more accurately and easily. This is because after the data cleaning process, approximately 1000 rows were removed from the original dataset. There was an attempt to fix this using oversampling but at the end of the day, the more real data there is, the better the model will be at classifying.

# CHAPTER 6: Results and Discussion

## 6.1 Introduction

As mentioned previously, the models were evaluated based on the metrics precision, accuracy, recall and F1-score. A table summarising the evaluation metrics for all of the models can be seen below.

## 6.2 Results and Discussion

Table : Evaluation metrics summary

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Non-optimised SVM** | **Optimised SVM** | **LSTM** |
| **Precision** | 91.71% | 93.64% | 61.60% |
| **Recall** | 91.02% | 93.62% | 78.49% |
| **F1-score** | 91.24% | 93.64% | 69.03% |
| **Accuracy** | 91.02% | 93.62% | 78.49% |

As can be seen from the table above which summarises the evaluation metrics for all of the models built, the best resulst were obtained using the optimised SVM model which used GridSearchCV and “rbf” as its kernel.

The precision of the model, which is a metric that measures how many correct positive forecasts have been made (Brownlee, 2020), comes out to 93.64%. The recall, which is a metric that measures how many correct positive predictions were produced out of all possible positive predictions (Brownlee, 2020), comes to 93.62%. The F1-score, which allows you to integrate precision and recall into a single metric (Brownlee, 2020), comes out to about 96.64% and finally, the accuracy of the model, which is the number of correctly classified data instances divided by the total number of data instances (B, 2019) comes out to 93.62% as well.

# CHAPTER 7: CONCLUSIONS AND REFLECTIONS

The ability to model and anticipate water quality is critical for environmental preservation, especially for under developed countries like Sudan. The future water quality can be measured by developing a model employing powerful artificial intelligence algorithms. The WQI was predicted using advanced artificial intelligence algorithms, specifically the LSTM model, in this proposed methodology. The WQI data was also classified using machine learning algorithms like SVM. Some statistical parameters were used to evaluate and examine the proposed models.

The performance of the SVM algorithm has obtained the best accuracy of the classification of the water sources, according to the WQI forecast. Following the evaluation of the suggested model's resilience and efficiency in forecasting the WQI, the generated models can be fed more data, both current and historical, in order to improve the models' efficiency. These models can then be deployed in other under-developed countries to help in the identification of safe water sources.

I feel that the research that was done for this paper was quite thorough, but I also feel that there is always room for improvement and additional research to be carried out which might improve the quality of the research. Having said that, I am quite happy and proud of the amount of work as well as the quality of work that I have done.

In conclusion, completing the project report has improved my preparedness for starting and finishing undertakings. Throughout the research, I have gained a great deal of knowledge and skills for which I am grateful, and it has inspired me to believe in my own ability to conduct future research using the skills and knowledge I have gained. Finally, despite the significant study conducted, there is still room for improvement in terms of the models' ability to classify data using real data.

# References

ActiveState. (2021). *What Is Pandas In Python? Everything You Need To Know*. Retrieved May 27, 2021, from ActiveState: https://www.activestate.com/resources/quick-reads/what-is-pandas-in-python-everything-you-need-to-know/

Adu-Manu, K. S., Tapparello, C., Heinzelman, W., Katsriku, F. A., & Abdulai, J.-D. (2017). Water Quality Monitoring Using Wireless Sensor Networks: Current Trends and Future Research Directions. *ACM Transactions on Sensor Networks, 13*(1), 1-41. https://doi.org/https://doi.org/10.1145/3005719

Ahmed, U., Mumtaz, R., Anwar, H., Shah, A. A., Irfan, R., & García-Nieto, J. (2019). Efficient Water Quality Prediction Using Supervised Machine Learning. *Water, 11*(11), 2210. https://doi.org/https://doi.org/10.3390/w11112210

Aldhyani, T. H., Al-Yaari, M., Alkahtani, H., & Maashi, M. (2020). Water Quality Prediction Using Artificial Intelligence Algorithms. *Applied Bionics and Biomechanics-, 2020*, 1-12. https://doi.org/https://doi.org/10.1155/2020/6659314

Arbel, N. (2018, December 21). *How LSTM networks solve the problem of vanishing gradients*. Retrieved June 8, 2021, from Medium: https://medium.datadriveninvestor.com/how-do-lstm-networks-solve-the-problem-of-vanishing-gradients-a6784971a577

B, H. N. (2019, December 11). *Confusion Matrix, Accuracy, Precision, Recall, F1 Score*. Retrieved from Medium: https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd

Bansal, S., & Geetha, G. (2020). A Machine Learning Approach towards Automatic Water Quality Monitoring. *Journal of Water Chemistry and Technology, 42*(5), 321-328. https://doi.org/https://doi.org/10.3103/S1063455X20050045

Barton, A. (2021). *WATER IN CRISIS - SUDAN*. Retrieved June 1, 2021, from The Water Project: https://thewaterproject.org/water-crisis/water-in-crisis-sudan#:~:text=Sudan%20faces%20ecological%20crises%20like,a%20lack%20of%20agricultural%20production.&text=Eighty%20percent%20of%20the%20country,rural%20and%20fed%20by%20rainwater.

Bhardwaj, N., Bhardwaj, S. K., Bhatt, D., Lim, D. K., Kim, K.-H., & Deep, A. (2019). Optical detection of waterborne pathogens using nanomaterials. *TrAC Trends in Analytical Chemistry, 113*, 280-300. https://doi.org/https://doi.org/10.1016/j.trac.2019.02.019

Brownlee, J. (2020, August 2). *How to Calculate Precision, Recall, and F-Measure for Imbalanced Classification*. Retrieved from Machine Learning Mystery: https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/

Bureau of Indian Standards. (2012). *DRINKING WATER — SPECIFICATION ( Second Revision ).* Bureau of Indian Standards.

Bureau of Reclamation. (2020, April 11). *Water Facts - Worldwide Water Supply*. Retrieved May 28, 2021, from Central California Area Office: https://www.usbr.gov/mp/arwec/water-facts-ww-water-sup.html#:~:text=3%25%20of%20the%20earth's%20water,water%20is%20available%20fresh%20water.

Butler, A. (2018, October 14). *Underdeveloped: Healthcare in Developing Nations*. Retrieved May 26, 2021, from Orbis Biosciences: https://orbisbio.com/underdeveloped-healthcare-in-developing-nations/

Centre for Disease Control and Prevention. (2021, February 26). *Transmission*. Retrieved June 2, 2021, from Centre for Disease Control and Prevention: https://www.cdc.gov/parasites/giardia/infection-sources.html

Chen, Y., & Han, D. (2018). Water quality monitoring in smart city: A pilot project. *Automation in Construction, 89*, 307-316. https://doi.org/https://doi.org/10.1016/j.autcon.2018.02.008

Chinta, R. R. (2020). Review on Various Supervised machine learning algorithms used in Data science. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 9*(11), 2978-2982.

Codecademy. (2021). *What is an IDE?* Retrieved May 27, 2021, from Codecademy: https://www.codecademy.com/articles/what-is-an-ide

Das, B., & Jain, P. (2017). Real-Time Water Quality Monitoring System using Internet of Things. *2017 International Conference on Computer, Communications and Electronics (Comptelix).* Jaipur: IEEE. https://doi.org/10.1109/COMPTELIX.2017.8003942

Dyouri, A. (2020, April 16). *How To Make a Web Application Using Flask in Python 3*. Retrieved June 1, 2021, from Digital Ocean: https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3#:~:text=Flask%20is%20a%20small%20and,only%20a%20single%20Python%20file.

Faggella, D. (2020, February 26). *What is Machine Learning?* Retrieved June 3, 2021, from Emerj The AI Research and Advisory Company: https://emerj.com/ai-glossary-terms/what-is-machine-learning/

Feather, G. (2020, August 3). *Water Quality Index (WQI)*. Retrieved from YouTube: https://www.youtube.com/watch?v=HTkNmmMoUzE&ab\_channel=GreenFeather

Gandhi, R. (2018, June 8). *Support Vector Machine — Introduction to Machine Learning Algorithms*. Retrieved June 8, 2021, from Towards Data Science: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

GeeksforGeeks. (2020, January 7). *Python GUI – tkinter*. Retrieved June 1, 2021, from GeeksforGeeks: https://www.geeksforgeeks.org/python-gui-tkinter/

Gil, M. (2015, October 22). *Why use a project management methodology?* Retrieved May 21, 2021, from Nae: https://nae.global/en/why-use-a-project-management-methodology/#:~:text=The%20general%20aim%20of%20project,in%20a%20continuous%20improvement%20process.

Gillis, A. S., & Silverthorne, V. (2018, September). *Integrated development environment (IDE)*. Retrieved May 27, 2021, from Search Software Quality: https://searchsoftwarequality.techtarget.com/definition/integrated-development-environment

Gour, R. (2019, March 8). *8 Unique Real-Life Applications of SVM*. Retrieved June 8, 2021, from Medium: https://medium.com/@rinu.gour123/8-unique-real-life-applications-of-svm-8a96ca43313

Huan, J., Li, H., Wu, F., & Cao, W. (2020). Design of water quality monitoring system for aquaculture ponds based on NB-IoT. *International Journal of Mechanical Engineering and Robotics Research, 9*(8), 1170-1175. https://doi.org/https://doi.org/10.1016/j.aquaeng.2020.102088

IBM Cloud Education. (2020, July 15). *Machine Learning*. Retrieved May 26, 2021, from IBM: https://www.ibm.com/cloud/learn/machine-learning

Jalal, D., & Ezzedine, T. (2019). *Performance analysis of machine learning algorithms for water quality monitoring system.* Tunis: IEEE.

Kachroud, M., Trolard, F., Kefi, M., Jebari, S., & Bourrié, G. (2019). Water Quality Indices: Challenges and Application. *Water, 11*(2), 1-26. https://doi.org/https://doi.org/10.3390/w11020361

Karczewski, D. (2020, June 18). *What Is The Best Language For Machine Learning In 2021?* Retrieved from Ideamotive: https://www.ideamotive.co/blog/what-is-the-best-language-for-machine-learning

Keras. (2021). *Keras*. Retrieved June 1, 2021, from Keras: https://keras.io/

Krishan, G., Singh, S., CP, K., Gurjar, S., & NC, G. (2016). Assessment of Water Quality Index (WQI) of Groundwater in Rajkot District, Gujarat, India. *Journal of Earth Science & Climatic Change, 7*(3). https://doi.org/10.4172/2157-7617.1000341

Lee, I., & Shin, Y. J. (2020). Machine learning for enterprises: Applications, algorithm selection, and challenges. *Business Horizons, 63*(2), 157-170. https://doi.org/https://doi.org/10.1016/j.bushor.2019.10.005

Lenntech. (2021). *Waterborne diseases*. Retrieved May 26, 2021, from Lenntech: https://www.lenntech.com/processes/disinfection/deseases/waterborne-diseases-contagion.htm#:~:text=Contagion%20by%20pathogenic%20microorganisms&text=Some%20waterborne%20pathogenic%20microorganisms%20spread,main%20symptom%20(figure%201)

Loukas, S. (2020, May 28). *Everything you need to know about Min-Max normalization: A Python tutorial*. Retrieved from Towards Data Science: https://towardsdatascience.com/everything-you-need-to-know-about-min-max-normalization-in-python-b79592732b79

Malik, F. (2019, April 1). *Why Should We Use NumPy?* Retrieved May 27, 2021, from Medium: https://medium.com/fintechexplained/why-should-we-use-numpy-c14a4fb03ee9

Mark, A. K. (2019, October 10). *Sudan Water Crisis*. Retrieved June 1, 2021, from https://storymaps.arcgis.com/stories/60088f40ca654e48b22ef9df3a45060e

Moparthi, N. R., Mukesh, C., & Sagar, P. V. (2018). Water Quality Monitoring System Using IOT. *2018 Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)* (pp. 1-5). Chennai: IEEE. https://doi.org/10.1109/AEEICB.2018.8480963

Muharemi, F., Logofătu, D., & Leon, F. (2019). Machine learning approaches for anomaly detection of water quality on a real-world data set. *Journal of Information and Telecommunication, 3*(3), 294-307. https://doi.org/https://doi.org/10.1080/24751839.2019.1565653

Muharemi, F., Logofătu, D., Andersson, C., & Leon, F. (2018). Approaches to Building a Detection Model for Water Quality: A Case Study. In F. Muharemi, D. Logofătu, C. Andersson, F. Leon, & Cham (Ed.), *Modern Approaches for Intelligent Information and Database Systems* (pp. 173-183). Springer.

Mujtaba, H. (2020, September 29). *Hyperparameter Tuning with GridSearchCV*. Retrieved from Great Learning: https://www.mygreatlearning.com/blog/gridsearchcv/

Myint, C. Z., Gopal, L., & Aung, Y. L. (2017). Reconfigurable smart water quality monitoring system in IoT environment. *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)* (pp. 435-440). Wuhan: IEEE. https://doi.org/10.1109/ICIS.2017.7960032

Narkhede, S. (2018, May 9). *Understanding Confusion Matrix*. Retrieved June 8, 2021, from Towards Data Science: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Nasteski, V. (2017). An overview of the supervised machine learning methods. *Horizons. b, 4*, 51-62. https://doi.org/10.20544/HORIZONS.B.04.1.17.P05

National Institude of Allergy and Infectious Diseases. (2012, March 15). *E. coli*. Retrieved June 2, 2021, from National Institude of Allergy and Infectious Diseases: niaid.nih.gov/diseases-conditions/e-coli

Noori, R., Berndtsson, R., Hosseinzadeh, M., Adamowski, J. F., & Abyaneh, M. R. (2019). A critical review on the application of the National Sanitation Foundation Water Quality Index. *Environmental Pollution, 244*, 575-587. https://doi.org/https://doi.org/10.1016/j.envpol.2018.10.076

Pandas Development Team. (2021). *pandas.DataFrame.plot.kde*. Retrieved from Pandas: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.plot.kde.html

Păpăluță, V. (2020, January 13). *What’s the best way to handle NaN values?* Retrieved May 27, 2021, from Towards Data Science: https://towardsdatascience.com/whats-the-best-way-to-handle-nan-values-62d50f738fc#:~:text=NaN%20or%20Not%20a%20Number,are%20represented%20as%20None%20value.

Pisner, D. A., & Schnyer, D. M. (2020). Chapter 6 - Support vector machine. In D. A. Pisner, D. M. Schnyer, A. Mechelli, & S. Vieira (Eds.), *Machine Learning* (pp. 101-121). Academic Press.

Quantum. (2019, August 20). *Data Science project management methodologies*. Retrieved May 27, 2021, from Medium: https://medium.datadriveninvestor.com/data-science-project-management-methodologies-f6913c6b29eb

Reid, K. (2021, April 16). *Global water crisis: Facts, FAQs, and how to help*. Retrieved May 26, 2021, from World Vision: https://www.worldvision.org/clean-water-news-stories/global-water-crisis-facts

Rojas, J. A., Kery, M. B., Rosenthal, S., & Dey, A. (2017). Sampling techniques to improve big data exploration. *2017 IEEE 7th Symposium on Large Data Analysis and Visualization (LDAV).* Pheonix: IEEE. https://doi.org/10.1109/LDAV.2017.8231848

Saltz, J. (2020, November 30). *CRISP-DM is Still the Most Popular Framework for Executing Data Science Projects*. Retrieved May 27, 2021, from Data Science Process Alliance: https://www.datascience-pm.com/crisp-dm-still-most-popular/

Scikit-learn. (2021). *Scikit-learn Machine Learning in Python*. Retrieved May 27, 2021, from Scikit-learn: https://scikit-learn.org/stable/

Shafi, U., Mumtaz, R., Anwar, H., Qamar, A. M., & Khurshid, H. (2018). *Surface Water Pollution Detection using Internet of Things.* Islamabad: IEEE.

Shafique, U., & Qaiser, H. (2014). A Comparative Study of Data Mining Process Models (KDD, CRISP-DM and SEMMA). *International Journal of Innovation and Scientific Research, 12*(1), 217-222.

Sharma, R. (2021, January 4). *6 Best Python IDEs for Data Science & Machine Learning [2021]*. Retrieved May 27, 2021, from upGrad: https://www.upgrad.com/blog/python-ides-for-data-science-machine-learning/

Sharma, S. (2017, September 6). *Activation Functions in Neural Networks*. Retrieved from Towards Data Science: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

Shung, K. P. (2018, March 15). *Accuracy, Precision, Recall or F1?* Retrieved June 8, 2021, from Towards Data Science: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

Silver, N. (2020, June 30). *Why Is Water Important? 16 Reasons to Drink Up*. Retrieved May 26, 2021, from Healthline: https://www.healthline.com/health/food-nutrition/why-is-water-important

Slesar, M. (2021). *Top 10 Java Machine Learning Tools and Libraries*. Retrieved May 27, 2021, from ONIX: https://onix-systems.com/blog/top-10-java-machine-learning-tools-and-libraries

Smart Vision Europe. (2021). *What is the CRISP-DM Methodology?* Retrieved May 27, 2021, from Smart Vision Europe: https://www.sv-europe.com/crisp-dm-methodology/

Smartsheet. (2021). *How to Choose the Right Project Management Methodology*. Retrieved May 21, 2021, from Smartsheet: https://www.smartsheet.com/content-center/best-practices/project-management/project-management-guide/how-choose-project-management-methodology

Springboard India. (2021, October 6). *Best language for Machine Learning: Which Programming Language to Learn*. Retrieved May 27, 2021, from Springboard: https://in.springboard.com/blog/best-language-for-machine-learning/

TensorFlow. (2021). *Introduction to TensorFlow*. Retrieved May 27, 2021, from TensorFlow: https://www.tensorflow.org/learn

TensorFlow. (2021). *tf.keras.layers.Embedding*. Retrieved from TensorFlow: https://www.tensorflow.org/api\_docs/python/tf/keras/layers/Embedding

U.S. Geological Survey. (2021). *How Much Water is There on Earth?* Retrieved May 28, 2021, from U.S. Geological Survey: https://www.usgs.gov/special-topic/water-science-school/science/how-much-water-there-earth?qt-science\_center\_objects=0#qt-science\_center\_objects

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making, 19*(1), 1-16. https://doi.org/https://doi.org/10.1186/s12911-019-1004-8

United Nations. (2017). *The United Nations World Water Development Report 2017. Wastewater: The Untapped Resource.* Paris: UNESCO.

United States Environmental Protection Agency. (2021, September 28). *Drinking Water*. Retrieved May 26, 2021, from United States Environmental Protection Agency: https://www.epa.gov/report-environment/drinking-water#:~:text=If%20drinking%20water%20contains%20unsafe,chronic%20diseases%20such%20as%20cancer.

Vasconcellos, P. H. (2018, December 22). *Top 5 Python IDEs For Data Science*. Retrieved May 27, 2021, from Datacamp: https://www.datacamp.com/community/tutorials/data-science-python-ide

Voskoglou, C. (2017, May 5). *What is the best programming language for Machine Learning?* Retrieved May 27, 2021, from Towards Data Science: https://towardsdatascience.com/what-is-the-best-programming-language-for-machine-learning-a745c156d6b7

Williams, P., & Berkley., J. A. (2016). *DYSENTERY (SHIGELLOSIS).* World Health Organisation.

World Health Organisation. (2018, January 31). *Typhoid*. Retrieved June 2, 2021, from World Health Organisation: https://www.who.int/news-room/fact-sheets/detail/typhoid

World Health Organisation. (2019, June 14). *Drinking-water*. Retrieved May 26, 2021, from World Health Organisation: https://www.who.int/news-room/fact-sheets/detail/drinking-water#:~:text=In%202017%2C%2071%25%20of%20the,at%20least%20a%20basic%20service.

World Health Organisation. (2021, July 27). *Hepatitis A*. Retrieved June 2, 2021, from World Health Organisation: https://www.who.int/news-room/fact-sheets/detail/hepatitis-a

Wu, J. (2019, August 29). *How to Choose A Programming Language for a Project*. Retrieved May 27, 2021, from Better Programming: https://betterprogramming.pub/how-to-choose-a-programming-language-for-a-project-7c7a3e5a4de6

Xagoraraki, I., & O’Brien, E. (2019). Wastewater-Based Epidemiology for Early Detection of Viral Outbreaks. *Women in Water Quality*, 75-97. https://doi.org/10.1007/978-3-030-17819-2\_5

Yalçın, O. G. (2020, November 24). *4 Reasons Why You Should Use Google Colab for Your Next Project*. Retrieved June 1, 2021, from Towards Data Science: https://towardsdatascience.com/4-reasons-why-you-should-use-google-colab-for-your-next-project-b0c4aaad39ed#:~:text=Google%20Colab%20is%20an%20excellent,such%20as%20GPUs%20and%20TPUs.

Yıldırım, S. (2020, June 1). *Hyperparameter Tuning for Support Vector Machines — C and Gamma Parameters*. Retrieved from Towards Data Science: https://towardsdatascience.com/hyperparameter-tuning-for-support-vector-machines-c-and-gamma-parameters-6a5097416167

# **Appendice****s**

## Poster

Graphical user interface, text

Description automatically generated

Figure 55: FYP Poster

## Confidentiality Document

Text

Description automatically generated

Figure : Confidentiality Document

## Library Cataloguing Details

Graphical user interface, text, application, email

Description automatically generated

Figure : Library Cataloguing Details

## Project Meeting Log Sheets

Calendar

Description automatically generated

Figure 58: Project Meeting Log Sheets 1 and 2

Calendar

Description automatically generated with low confidence

Figure 59: Project Meeting Log Sheets 3 and 4

Calendar

Description automatically generated with low confidence

Figure 60: Project Meeting Log Sheets 5 and 6

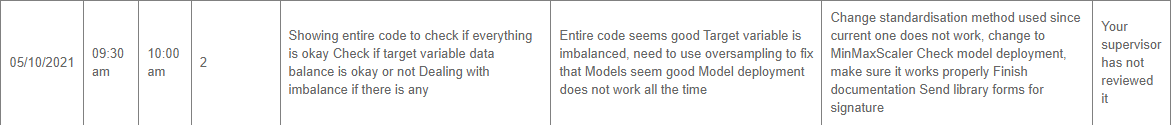


Figure 61: Project Meeting Log Sheet 7

## Project Proposal Form

Text, letter

Description automatically generated

Figure 62: Project Proposal Form 1

Text, letter

Description automatically generated

Figure 63: Project Proposal Form 2

Text, letter

Description automatically generated

Figure 64: Project Proposal Form 3

## Project Specification Form

Text, letter

Description automatically generated

Figure 65: Project Specification Form 1

Text, letter

Description automatically generated

Figure 66: Project Specification Form 2

Text, letter

Description automatically generated

Figure 67: Project Specification Form 3

Graphical user interface, text, application, letter

Description automatically generated

Figure 68: Project Specification Form 4

Text, letter

Description automatically generated

Figure 69: Project Specification Form 5

Text, letter

Description automatically generated

Figure 70: Project Specification Form 6

## Ethics Form

Table

Description automatically generated

Figure 71: Ethics Form 1

Table

Description automatically generated

Figure 72: Ethics Form 2

Text, letter

Description automatically generated

Figure 73: Ethics Form 3

Graphical user interface, text, application, email

Description automatically generated

Figure 74: Ethics Form 4

## Gantt Chart

Chart, bar chart

Description automatically generated

Figure 75: Gantt Chart for FYP Semester 1 (IR)

Chart, bar chart

Description automatically generated

Figure 76: Gantt Chart for FYP Semester 2 (FYP)