BABEŞ-BOLYAI UNIVERSITY CLUJ-NAPOCA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
SPECIALIZATION [Secție]

DIPLOMA THESIS

Precision Agriculture: Predicting crop yield based on environmental data.

Supervisor
Lecturer, PhD. Adriana M. COROIU

Author
Crisan Dragos

2024

ABSTRACT

This paper aims to illustrate how integrating AI into agriculture can potentially enhance production of the fields. It also seeks to encourage documenting the environmental conditions and outcomes associated with each crop field for future development with AI and agriculture data with aim on improving efficiency and yield. Workers in this field face challenges in considering all factors when making decisions, often relying on intuition. However, this paper will delve into these natural factors that significantly influence crop outcomes, outlining the potential long-term impact it could have in agriculture, regarding the knowledge farmers posses about the quality of the soil. Introducing a tool that would make it easier for them to make decisions while taking such data into account would greatly benefit them. By providing chemical data about the soil at the start of the year, this machine learning algorithm can predict the yield of different crops. This can later be developed to use these results to recommend what crop would be best for the current year, or what the chemical levels of the soil should be for their desired crop. This model was initially trained on a public dataset from India. However, it can easily be adapted to other countries with varying climates, provided there is access to relevant training data.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Objective 1

1.3 Composition of the thesis 2

2 Theoretical Insights **3**

2.1 World population 3

2.2 Arable land 4

2.3 Food security 6

2.4 The major three plant nutrients: N, P, K 7

2.4.1 Potassium(K) 7

2.4.2 Nitrogen(N) 9

2.4.3 Phosphorus(P) 10

2.4.4 Supplementation trough fertilization 12

2.5 Artificial Intelligence 13

2.5.1 Definition and Origins 13

2.5.2 The branches of AI 15

2.5.3 Machine Learning 16

2.5.4 Extreme Gradient Boost 18

2.5.5 Neural Networks 19

3 Integrating AI into Agriculture **22**

3.1 State of the art 22

3.2 My Vision 25

3.3 Model Training 26

3.3.1 Data extraction 26

3.3.2 Data transformation 30

3.3.3 Extreme Gradient Boosting ALgorithm 31

3.3.4 Convolutional Neural Network Algorithm 33

3.3.5 Fully Connected Neural Network Algorithm 33

4 Application building **35**
4.1 Database 35
4.1.1 Technology used 35
4.1.2 Code description 36
4.2 Server 36
4.2.1 Technology used 36
4.2.2 Code description 36
4.3 Frontend 37
4.3.1 Technology used 37
4.3.2 Code description 38
4.4 Similar aproaches 41

5 Conclusions and future developments **42**
5.1 Conclusions 42
5.2 Further improvements and developments 42

Bibliography **I**

List of Figures

2.1 Population Demographic [RRGM+23] 3
2.2 Population by age group demographic[RRGM+23] 4
2.3 Median age in Romania [worb] 5
2.4 Arable land[NBVJB11] 5
2.5 Desert soil[Tru24] 6
2.6 Number of people with food insecurity, 2016-2022[GRF] 7
2.7 Impact of different K levels on cereals[oIiawTTBS20] 8
2.8 Grasslands K defficiency[oIiawTTBS20] 9
2.9 Two fields with different K levels[oIiawTTBS20] 9
2.10 Nitrogen cycle[oAF05] 10
2.11 Leaves with Phosphorus defficiency 11
2.12 Nitrogen cycle[cyc13] 11
2.13 Mobility of Macronutrients[oIiawTTBS20] 12
2.14 The 5 components of an information system 13
2.15 The 6 branches of AI[Ana] 15
2.16 The subsets of Machine Learning 16
2.17 Visual representation of supervised learning 17
2.18 Visual representation of unsupervised learning[oIiawTTBS20] 17
2.19 Visual representation of a Gradient Boost using Trees 18
2.20 Visual representation of a Neural Network 19
2.21 Visual representation of a Neural Network 20

3.1 Depiction of the crop Dataset statistics 26
3.2 Image of the soil Dataset statistics 27
3.3 Picture of the water Dataset 27
3.4 Visual representation of frequency and data spread 28
3.5 Diagram of density of data around crop yield 29
3.6 XGB Encoder code 30
3.7 XGB Encoder code 30
3.8 Visualisation of predictions versus actual values 31
3.9 Comparison between the errors to statistically show the invariance . 31

3.10 R^2 Score 32
3.11 Before and after hyperparameter tuning comparison 32
3.12 Actual versus predicted Crop Yields 33
3.13 Actual versus predicted Chemical Trio 34

4.1 Database Schema 35
4.2 Code of the integration of machine learning algorithms into the server 36
4.3 Code representation of the Login and Register Request 37
4.4 Code representation of the XGB Request 38
4.5 Visual representation of the frontend structure 38
4.6 Visual representation of login page 39
4.7 Visual representation of login page after receiving the login token . . 39
4.8 Visual representation of crop prediction page 40
4.9 Visual representation of crop prediction page 41

Chapter 1

Introduction

1.1 Motivation

In recent years, our climate has undergone significant changes, deeply impacting the agricultural sector. On top of the constant change of the climate making it harder for farmers to predict which would be the best crop to be planted, our society continues to expand while arable land is finite. At the same time, not all farmers consider all the knowledge there is about crops. Reaching better productivity out of the current resources we have would be a great improvement on top of making certain decisions easier for farmers. We were determined to look for options to implement artificial intelligence into our agricultural sector in a way that would make farming smarter and more efficient.

1.2 Objective

The objective of this project is to encourage the advancement of implementing AI into this field and bringing more efficiency and effectiveness to farming practices. We want to promote both the collecting and utilization of data by showing the effects it can have. With future developments in this field, we anticipate simpler decision-making processes for our farmers, coupled with reducing the waste of resources such as fertilizer. Our approach helps workers manage and organise by providing accurate predictions about the crop yields their fields will have and if this project were to develop further it could ensure optimal utilization of resources where they are most needed, which provides greater efficiency and sustainability. Additionally, the data collected, and the progress achieved in this field could have the potential to stimulate further innovations in the agricultural sector.

1.3 Composition of the thesis

This Paper is divided into four main chapters: Theoretical background, Integrating AI into Agriculture, Application building and a Synthesis.

First, this paper will describe the background for the field, discussing the importance of improving the productivity of the crops due to the population situation as well as the climate change situation. The key chemical factors to take into consideration when talking about improving productivity of crops are the three chemical elements: potassium (K), nitrogen (N), and phosphorus (P). In this chapter we will delve into their significance for various crops and how achieving optimal balances of these elements can lead to better annual crop yields. We will also showcase the notions of artificial intelligence that were used in the creation of the program.

Second, we will present other implementations in the field and compare them to our vision and what we have achieved. We will continue by showing the data we are working on and how we chose to process it in order to prepare it for the algorithms. Next, we with the presentation of the AI implementation featured in this project, containing descriptions for all machine learning algorithms chosen and the reasons behind choosing them, as well as their respective results. Here we will also talk about the possible developments of the project, as the second and third machine learning algorithms were built for this purpose, to prepare the path for future shifts.

Third, we will present the application building of the project. Here we will talk about the technologies used, the structure of the application and the communication between the layers. We will present the Database, followed by the Server and the Frontend, with visual representations of the code and the website.

Lastly, we will summarize the contents of my paper while bringing together the conclusions made along the way. These conclusions will be examined, and their impact will be discussed, along with future developments that can be made on this project.

Chapter 2

Theoretical Insights

2.1 World population

The population of the earth experienced a rapid increase to 7 billion in 2011. It was estimated at that time that by the year 2025 we would reach 8 billion, and by 2050 it would reach 9 billion. As we have reached the year 2024, we can already see that the first prediction was accurate, as we currently reached 7.95 billion people. The following graph serves as visual aid to illustrate the demographic expansion2.1.

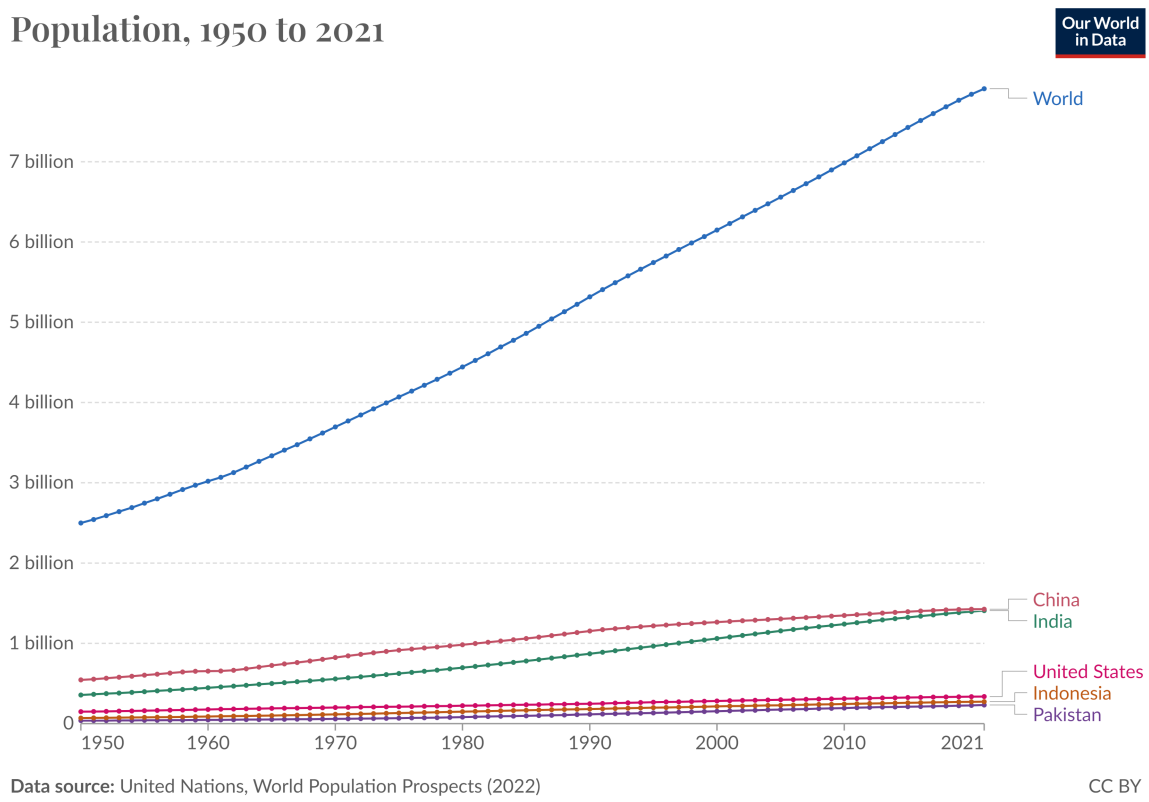

Figure 2.1: Population Demographic [RRGM+23]

The growth of population has multiple implications on the demand for food, one

of them being the shift in age distribution2.2. As the average age of the population rises, so does the percentage of elderly people, affecting available agriculture labor force. This means that with this increase, there is a corresponding decline in the proportion of people eligible to work on crop fields. This shift enforces the need of finding simpler and more efficient methods in agriculture to meet the increasing demand for food.

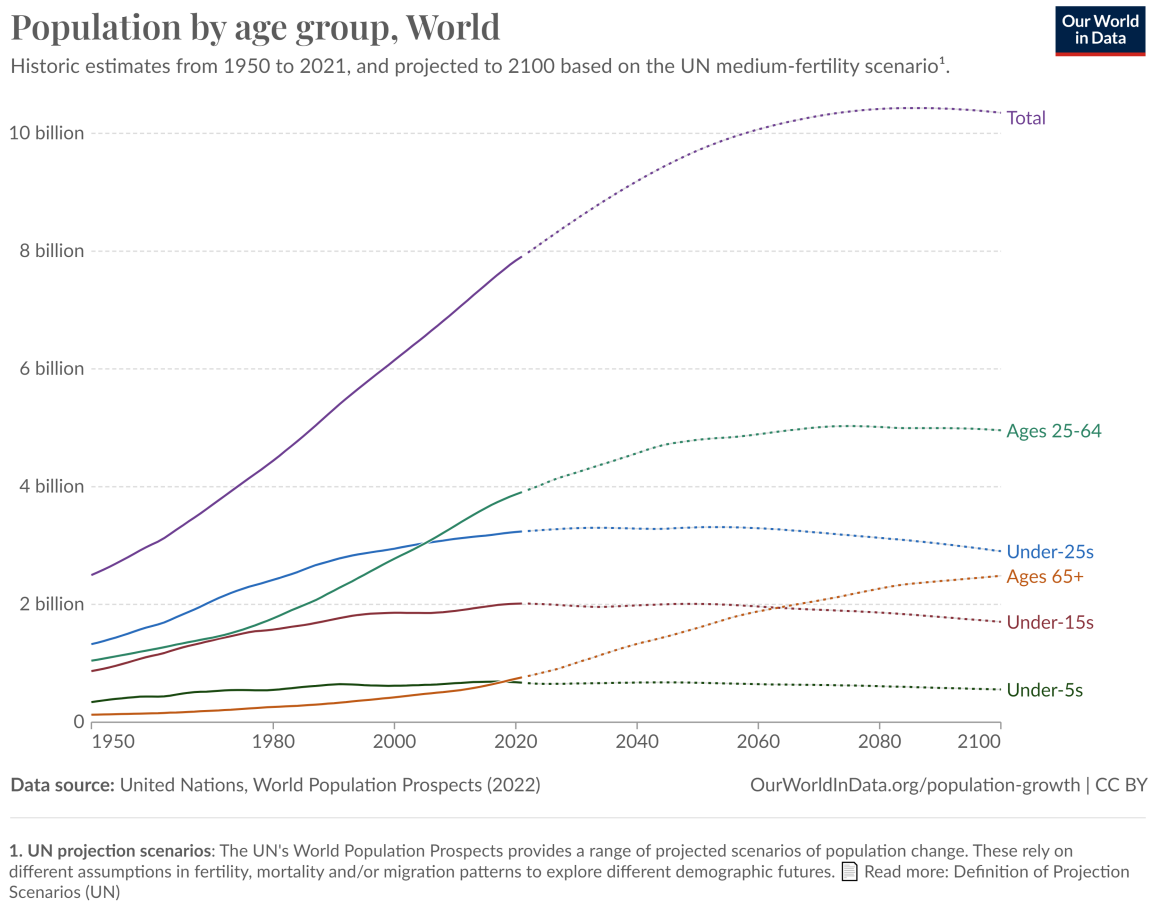

Figure 2.2: Population by age group demographic[RRGM+23]

This trend is not unique to other countries or just the general demographic of the world. Romania has been mirroring the global trend towards aging population2.3. Not to mention the fact that more and more of the teen workforce are migrating to urban areas in search for job opportunities. This trend is further exacerbating the decline in the agricultural workforce.

2.2 Arable land

Although the population of the earth continues to rise, it is crucial to acknowledge that the land suitable for cultivation remains constant. As we reflect on the predictions made regarding the earth's population growth, it becomes clear that the

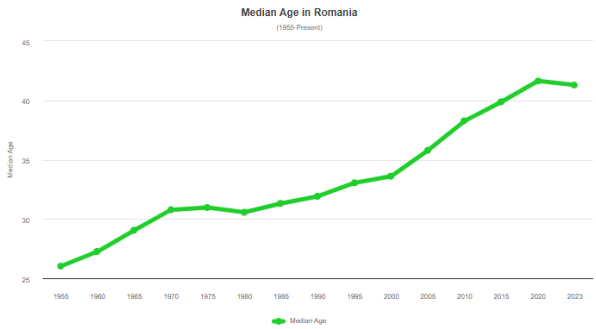

Figure 2.3: Median age in Romania [worb]

agricultural district should have at least a similar growth rate. The static availability of cultivable land creates a necessity for innovation in terms of productivity and efficiency.

The expansion of agricultural land continues to take place, predominantly in the developing countries. Data shows that the expansion of arable land continues to be an important factor in the agricultural growth in sub-Saharan Africa, Latin America and East Asia. However, in the developed countries there is a decline in arable land. This decline has been accelerating over time. This is further illustrated in the following graph2.4.

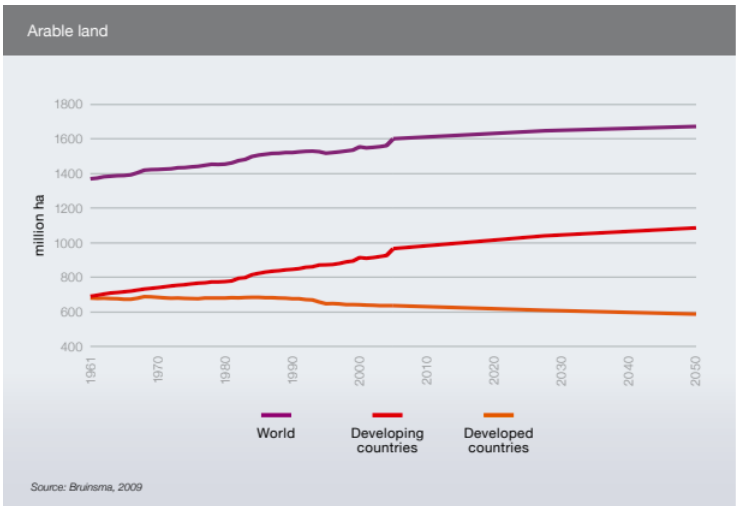

Figure 2.4: Arable land[NBVJB11]

From this graph we can conclude that although the arable land available is increasing, the rate of this increase is decelerating over time. Also, if we look further into the future, as more countries become developed and more land is sacrificed for the expansion of the cities due to the growth of the population, we can anticipate that it can start declining.

On top of the slow deceleration in the creation of new arable land, another very important factor to take into consideration is the degradation of existing arable land. This poses a global environmental challenge, as 25 percent of the total arable land has been degraded. Scientists recently approximated that around 24 billion tons of fertile soil was lost per year, with the biggest contributing factor being unsustainable agriculture practices. This alarming reality underscores the necessity of innovation in the field[GEF].
Examples of soil degradation can also be found in Romania, where it is estimated that the country loses roughly one thousand hectares annually due to desertification in the southern hemisphere, in regions like Dolj or Mehedinti, which rely heavily on the agriculture industry[Ian24].

Figure 2.5: Desert soil[Tru24]

We need to explore new sustainable methods for our agriculture sector, driven not only by considerations of production and efficiency but also by the urgency of addressing these alarming factors. Considering this, it becomes essential that future development and innovation in the agriculture district occurs, raising the efficiency of the farms.

2.3 Food security

Agriculture-driven growth and food security are at risk. Due to multiple shocks, such as Covid-19, extreme weather conditions, pests and conflicts, the food system has been greatly impacted. Climate change is also having a great impact as it is changing the conditions the crops grow in and altering productivity. The goal of ending global hunger by 2030 is currently off track[Wora].

The findings of the Global Report on Food Crises 2023 suggests that more than 250 million people were in Crisis or worse, as the food insecurity levels have increased for the fourth consecutive year. Weather extremes, economic shocks and conflicts are the most common and recurring factors that are affecting the agriculture security[GRF].

For a better visualisation of the prevalence of acute food insecurity among the countries in GRFC, we can refer to the following graph, showing data over the years 2016-2022.

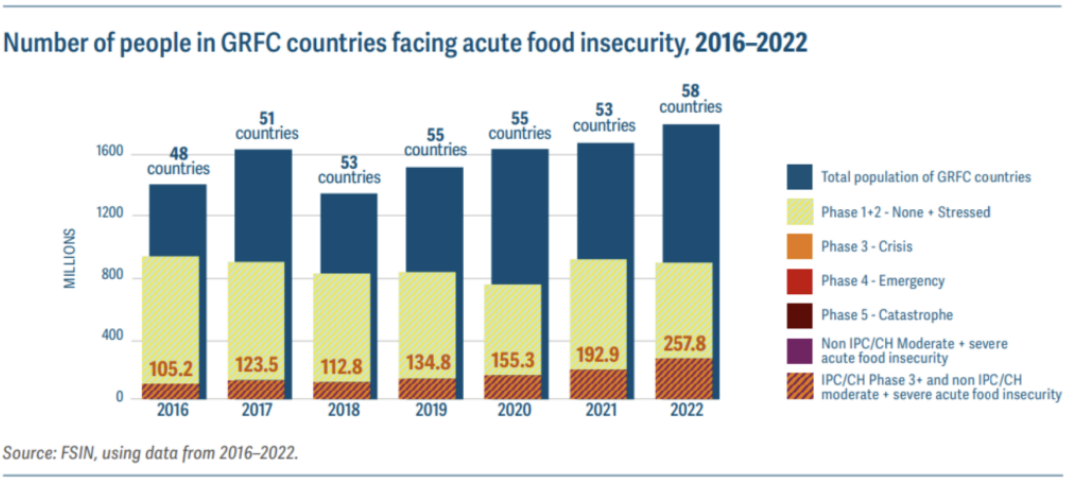

Figure 2.6: Number of people with food insecurity, 2016-2022[GRF]

While food waste and distribution are driving factors, agriculture productivity significantly influences the global availability of crop resources.

2.4 The major three plant nutrients: N, P, K

Essential nutrients needed by the plants to grow can be found naturally in the soil. Certain soils may lack the necessary nutrients for optimal plant growth and nutrition. In such cases, the levels of the nutrients in the soil are raised using fertilizers. The most important nutrients with the highest impact on the growth of the plant are Nitrogen(N), Phosporus(P) and Potassium(K). I will now delve deeper into the importance of all of these macronutrients and their impact on the plant.

2.4.1 Potassium(K)

As one of the three primary plant macronutrients, potassium enhances the overall growth of the plant, especially fruit development and aroma. It is classified as

macronutrient because plants take a large quantity of K during their life cycle. Potassium has a high impact on the plants resistance to diseases, it's movement of water, nutrients and carbohydrates in the plant tissue. He is also envolved with enzyme activation within the plant, for an approximate number of over sixty plant enzymes for plant growth. As we can tell, Potassium has a high impact on the development of the plant, especially on cereals that have high K demand, which is further enforced by experiments with using different levels of the nutrient2.7.

Figure 2.7: Impact of different K levels on cereals[oIiawTTBS20]

Potassium deficiency, often called "hidden hunger", can be quite difficult to identify during growth season. While K plays such a key role at cellular level, visual symptoms of K deficiency are hard to identify. For a better identification, we can turn to the soil, where we can check if the K concentrations are sub-optimal for grass or crop production. This deficiency can manifest in various ways, including stunted growth, yellowing or browning of leaves, and decreased overall vigor. For instance, in grasslands, potassium deficiency may result in sparse vegetation and reduced forage quality. For reference, in the following image we can see parts of the field that are either rich or deficient in Potassium2.8.
Similarly, in cultivated grasses, such as turfgrass or pasture grasses, potassium deficiency can lead to diminished yields, poor root development, and increased vulnerability to pests and diseases. Diminished yields can easily be observed as in the example bellow2.9.

Figure 2.8: Grasslands K defficiency[oIiawTTBS20]

Figure 2.9: Two fields with different K levels[oIiawTTBS20]

2.4.2 Nitrogen(N)

The importance of nitrogen in the development of the plant cannot be overstated. Nitrogen is a crucial component of amino acids, proteins, and chlorophyll, essential for various metabolic processes. It plays an essential role in the development of the vegetative parts of the plant, as well as in the ability of the plant to perform photosynthesis. Nitrogen is a common element that is present in approximately 78% of the earths atmosphere. As the compound cycles through the air, soil and water, it undergoes transformations and takes different forms, resulting in nitrogen-based compounds and molecules, which are much needed for the growth of plants. We can see this cycle represented in the figure down below2.10. But before it can be used by plants, it needs to be converted to a form that are available for plants. This process is called mineralization. Following this process, the nutrients can be absorbed through the roots of the plant.

Preferably, it would be best both economically and environmentally if the nitrogen would never leave this cycle. However, as we can see in the figure2.10, leakages can occur. When nitrogen leaves the cycle in high quantities it can lead to environmental harm, such as water contamination, air pollution, climate change and eutrophication. On top of that, a soil that is too rich in nitrogen can also affect the

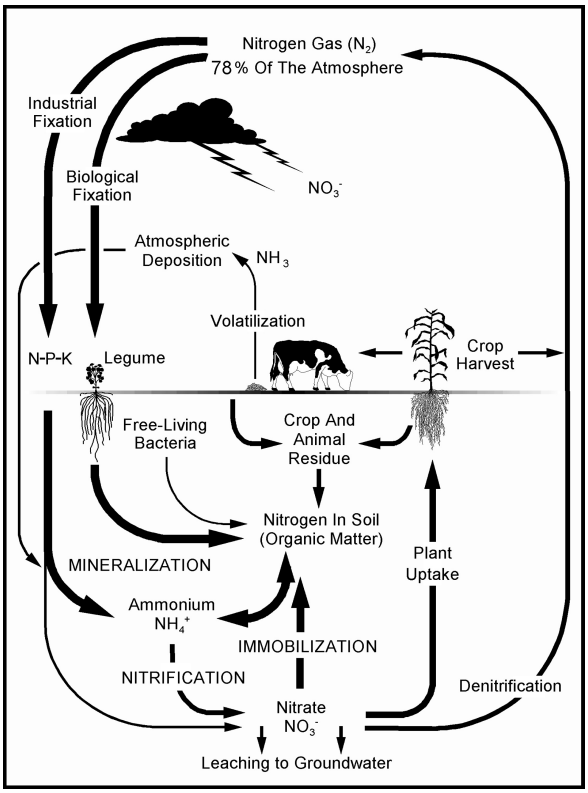

Figure 2.10: Nitrogen cycle[oAF05]

plant negatively, as it constrains the crops absorption of other nutrients. This further proves the importance of optimized usage of fertilizers to supplement the levels of nitrogen.

2.4.3 Phosphorus(P)

Phosphorus takes his rightful place together with Nitrogen and Potassium among the three macronutrients essential for plant growth. It has a high impact on flowering and fruit formation, it enables photosynthesis and helps the plant cope with stress.

One of the easiest signs of phosphorus deficiency is the color of older and lower leaves. Their color turns into a darker yellow or green and can have spots that are eitherbronze, brown or slightly blue, as we can see in the following examples2.11.

Phosphorus circulates through a cycle involving rocks, water, soil, sediments, and living organisms. Over time, rain and weathering break down rocks, releasing phosphate ions and other minerals, which are then dispersed into the soil and water. Plants absorb inorganic phosphate from the soil, which is then incorporated into organic molecules such as DNA. When plants or animals die and decay, the organic phosphate is returned to the soil, where it can be made available to plants again through mineralization by soil bacteria. Excess Phosphorus from the soil can also

(a) Caption for Image 1

(b) Caption for Image 2

Figure 2.11: Leaves with Phosphorus defficiency

end up in waterways travel all the way into oceans, where it becomes embedded into sediments over time. We can watch this cycle through the following visual representation2.12

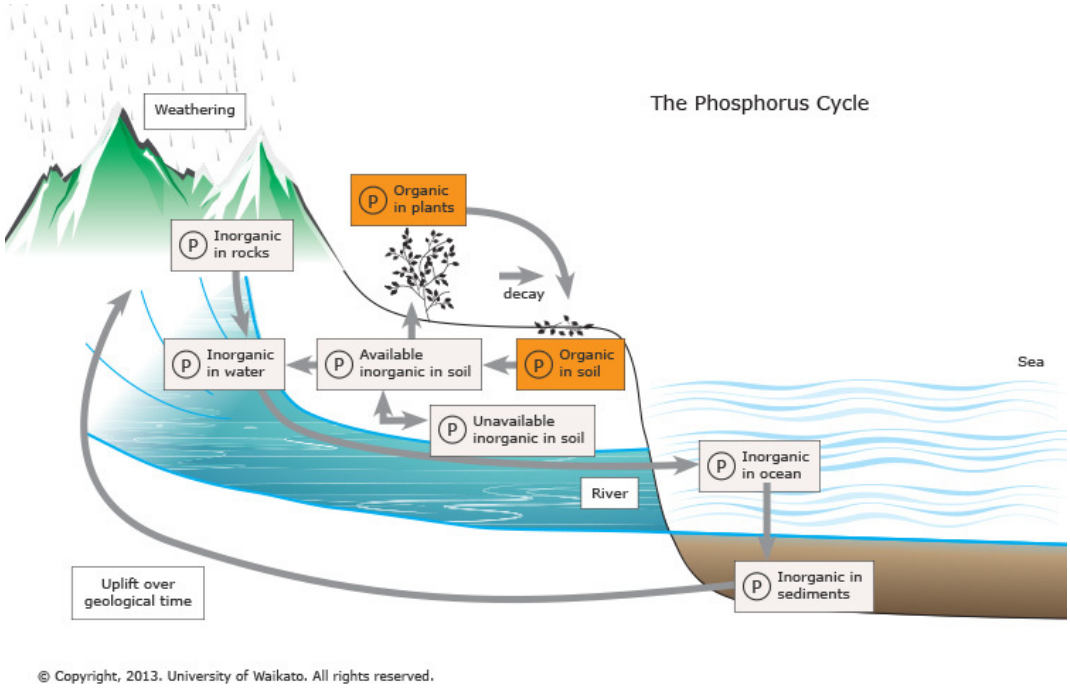

Figure 2.12: Nitrogen cycle[cyc13]

It can also be harmful to the environment as it leaks outside the cycle, by contributing to eutrophication, which occurs when excessive nutrients stimulate the growth of algae and other aquatic plants. When these die and decompose, they consume oxygen, leading to suffocation of the fish and other organisms.

2.4.4 Supplementation trough fertilization

Each of these three macronutrients has specifics when it comes to fertilizers, but one key factor about them is their mobility. This directly affects how easily they can be replenished in the soil through fertilization. Down below we can see a representation of the mobility each macronutrient has in the soil.2.13. These different mobilities make for different fertilizing practices. For example, Nitrogen(N), being highly mobile, makes it susceptible to leaching, necessitating frequent application to meet the plant demands. Phosporus, on the other hand, exhibits lower mobility due to its strong adsorption to soil particles, leading to fixation and reduced availability to plants. As a consequence, fertilizers for Phosporus supplementation are generally applied in advance, during soil preparation season. Potassium, beeing somewhat in the middle when it comes to mobility, allowing for flexibility in fertilizer application timing.

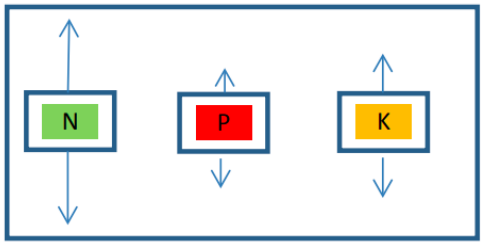

Figure 2.13: Mobility of Macronutrients[oIiawTTBS20]

As we concluded from the previous chapters about the three macronutrients, excess can be very harmful to the environment. This makes it really important for the farms to use the right amount of fertilizer, as it is not only wasteful and costly, but it also has negative impacts on the surroundings.

2.5 Artificial Intelligence

2.5.1 Definition and Origins

When we speak about "artificial intelligence", we often think about a program that can take available information, learn from it and make certain decisions, in a way that mimics how the human brain works. It represents the ability of solving certain problems in an efficient way, while being under certain constraints or circumstances. However, AI is not about creating new knowledge, but about gathering, processing and using already existing data to make decisions, predictions or to draw conclusions. It rests on three basic pillars, Data, Hardware and Software. The Data represents the information collected which is later organised in a special manner to ease the automation process. The data inputs can vary, from number, text to images and videos, each raising the level of complexity of the process. The hardware is the engine, representing the computing power necessary to process all the data in a fast and accurate way. The software represents the collection of tasks and instructions used as guidance to model and transform data into new information, conclusions or to establish patterns. Just like the computer, AI is becoming a tool that enhances our capabilities as humans to create projects or learn information, at a higher speed[Ibe].

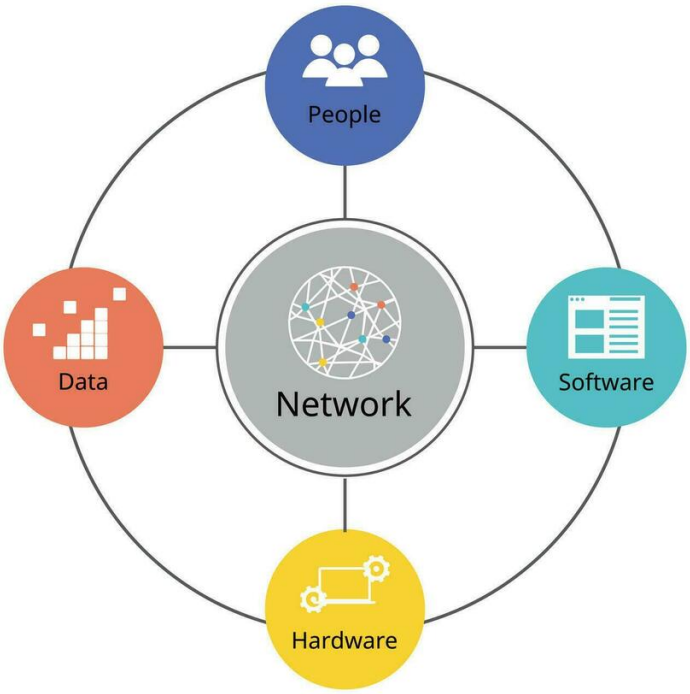

Figure 2.14: The 5 components of an information system

The term "artificial intelligence" was brought to light by John McCarthy in 1956 and drove the process of developing the first AI programming language, called LISP, in 1960. More complex systems were created during the 70's and 80's, as the field started to attract attention and funding. This technology has since evolved evolved exponentially, especially thanks to the development of deep learning, which added layered artificial neural networks to process and interpret complex data structures.

The idea of creating a program that could mimic the way the human brain works is not something new. It has been thought upon since the ancient times, with the first breakthroughs being the invention of Warren McCulloch and Walter Pitts in 1943, which was called the "model of artificial neurons", followed by the paper published by Alan Turing in 1950 called "Computing machinery and intelligence" in the magazine Mind where he questioned the ability of the computing machines to think. He later on created an experiment which was named the "Turing Test", whose purpose was determining whether a machine could have intelligent behaviour similar to a human or not.

2.5.2 The branches of AI

AI is divided into six branches, each with unique features, essential for creating smart systems capable of solving complex problems. The said branches are: Machine Learning, Neural Networks, Expert Systems, Robotics, Natural Language Processing and Fuzzy Logic, as we can see in the following representation2.15.

Figure 2.15: The 6 branches of AI[Ana]

Machine learning is a branch of Artificial Intelligence focused on creating algorithms and statistical models that allow computers to learn from data. This learning enables them to enhance their performance over time without needing explicit programming. In essence, it empowers machines to make decisions based solely on the provided data, eliminating the need for human intervention.
Neural networks are a type of machine learning algorithm inspired by the human brain's structure and function. They aim to replicate the neural connections through layers of "neurons" that collaboratively process information. Each layer's output is passed to the next neuron, ultimately reaching a conclusion at the network's end.
Expert systems are specialized artificial intelligence programs designed to solve problems within a specific domain. They utilize a knowledge base—a repository of information and rules about the subject—and an inference engine, a program that applies this knowledge to make decisions or solve issues.
Robotics encompasses the engineering and science fields involved in designing, building, and programming robots. These robots can perform various tasks autonomously or under human supervision, equipped to move and interact with their surroundings.
Natural Language Processing (NLP) is a domain of artificial intelligence dedicated to enabling computers to understand human language. NLP allows machines to recognize and interpret spoken or written words as humans do. This technology is increasingly popular for automating customer service and enhancing data analysis.

Fuzzy logic is a reasoning approach used in artificial intelligence systems that mimics human thought by allowing for varying degrees of truth between the absolute states of true or false. It accommodates "partial truths" or uncertainty in decision-making processes[Pau].

2.5.3 Machine Learning

As described previously, machine learning is on the idea of having the algorithm learn without human aid, outside of the input data provided to it. It is essentially an algorithm that learns from the provided data how to make predictions and learn overtime and improve. The efficient program of the algorithm allows it to process high amounts of data, which would take abundant time if it were done by humans, and allows it to make decisions based on patterns and correlations the algorithm finds.

Machine learning subsets

Machine learning consists of various subsets, each focusing on different aspects of the supervision process.

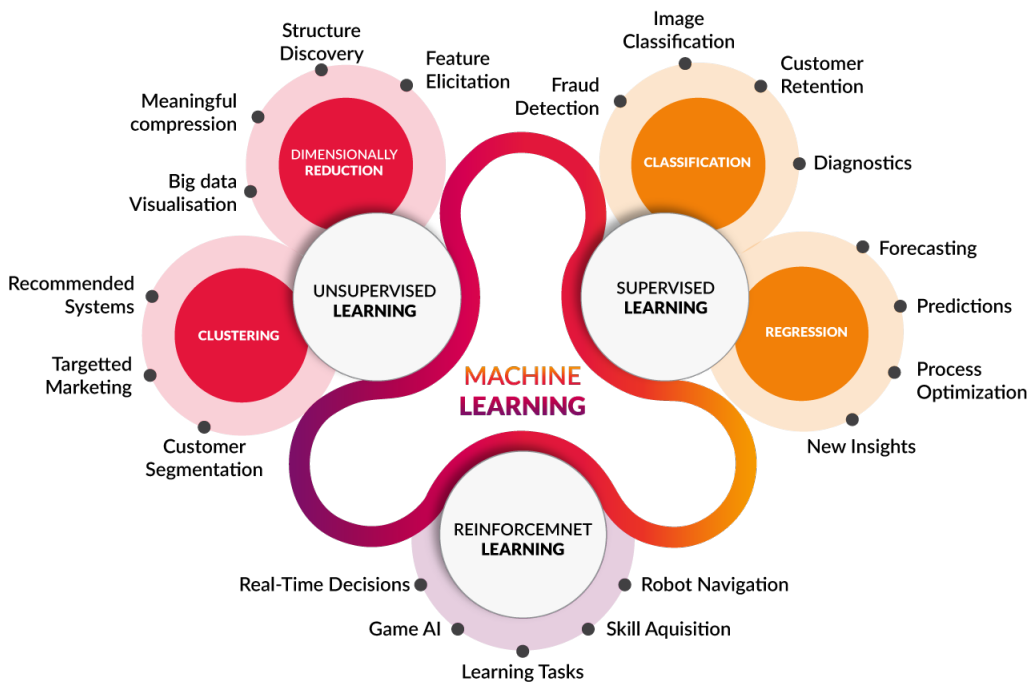

Figure 2.16: The subsets of Machine Learning

Supervised machine learning uses labeled datasets to train algorithms to accurately classify data or predict outcomes. As the model receives input data, it continually

adjusts its parameters until it is properly fitted. This occurs through a process called cross-validation, which helps to prevent the model from overfitting or underfitting. Supervised learning is widely used to solve a variety of real-world problems at scale, such as classifying emails into spam or non-spam folders. Supervised learning commonly employs methods such as neural networks, naïve Bayes, linear regression, logistic regression, random forest, and support vector machines (SVM).

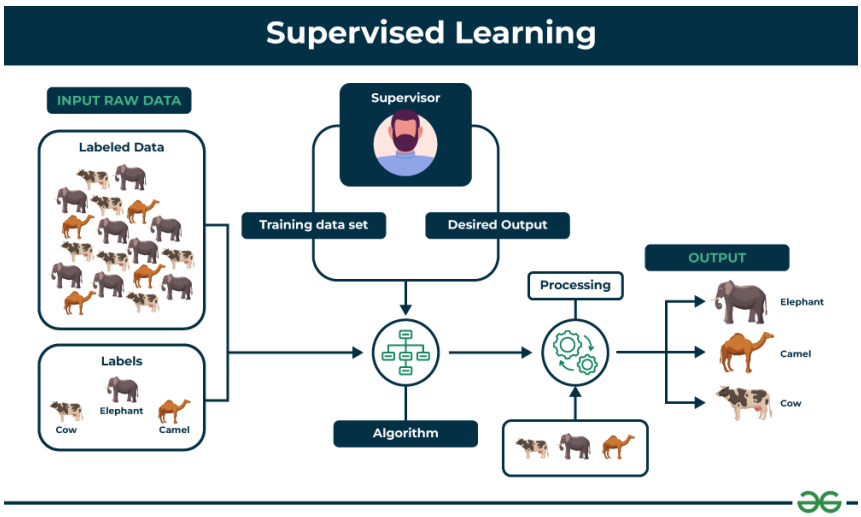

Figure 2.17: Visual representation of supervised learning

Unsupervised machine learning Unsupervised learning, takes unlabeled datasets and uses algorithms to analyze and cluster them. The purpose of this process is finding hidden patterns and creating groups of them without any human assist. This method is ideal for exploratory data analysis, cross-selling strategies, customer segmentation, and image and pattern recognition. Additionally, unsupervised learning uses dimensionality reduction to achieve a smaller number of features then the previous one. The common techniques used are Principal component analysis (PCA) and singular value decomposition (SVD). Other algorithms that we can meet in this category of learning are neural networks, k-means clustering, and probabilistic clustering methods.

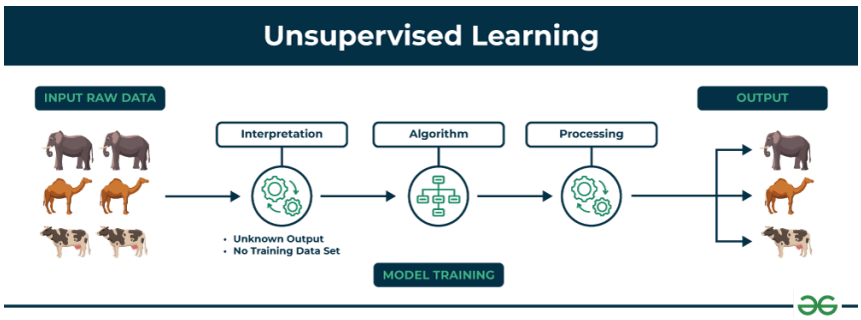

Figure 2.18: Visual representation of unsupervised learning[oIiawTTBS20]

Semi-Supervised machine learning strikes a balance between the two learning methods presented previously. It utilizes a smaller labeled dataset to help classify and extract features from a larger, unlabeled dataset. This approach is useful especially for cases where the labeled data to train a supervised learning algorithm is limited or when labeling data is too costly. Semi-supervised learning can leverage the benefits of both supervised and unsupervised methods to improve model performance.

2.5.4 Extreme Gradient Boost

Extreme gradient boost is a part of the machine learning subset, that is particularly known for it's ability to efficiently reading data and making predictions. It excels in both classification and regression tasks. It has the ability to combine multiple decision tree's, while also having features like regularization, making it a really good choice especially when it comes to big amounts of data that need to be processed.

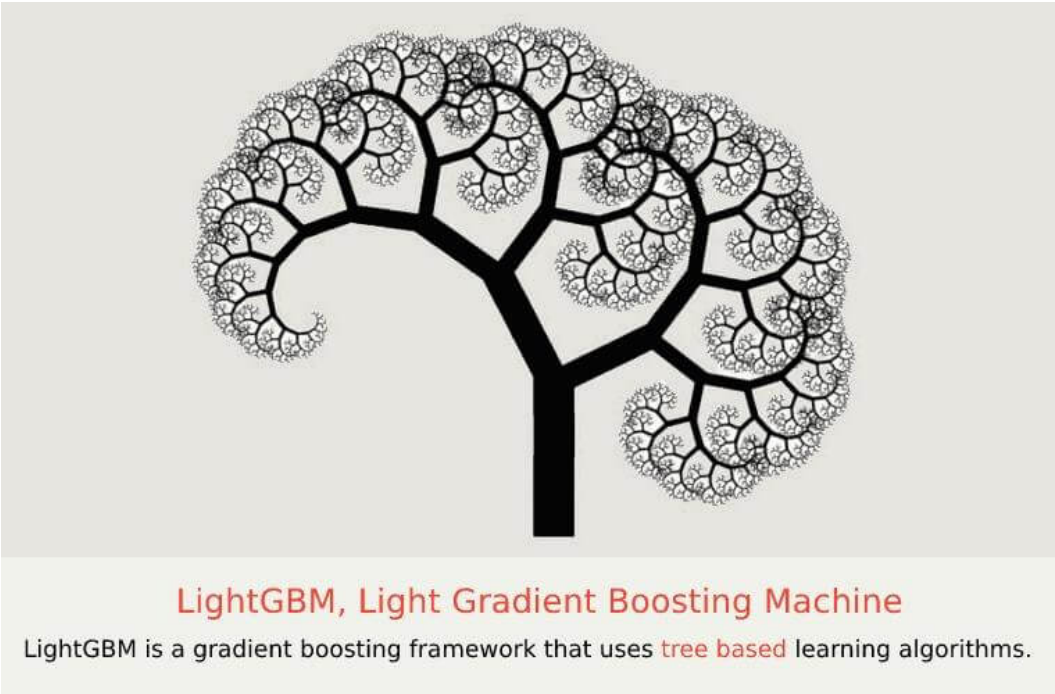

Figure 2.19: Visual representation of a Gradient Boost using Trees

To further understand how this algorithm works, we will delve into the impact of weak learners, for example the Decision Trees. A weak learner is a model that produces slightly better results than a guessing model. Boosting is the concept of combining multiple weak learners, up to the number of hundreds, to build a strong learner with great improvements on the accuracy.

2.5.5 Neural Networks

Each Neural Network is formed from an input layer, where the data is received, a hidden layer and an output layer for the results. The hidden layer is formed of multiple levels of processing information, through various mathematical equations that are applied to the data. Neural networks, as it's name suggests, is an attempt at mimicking the way human brains work, with neurons being represented by parts of the algorithm where the data is processed and only if the result passes a certain threshold, it is passed onto the next "neuron", until it reaches the end where the algorithm makes the conclusion and reaches a result.

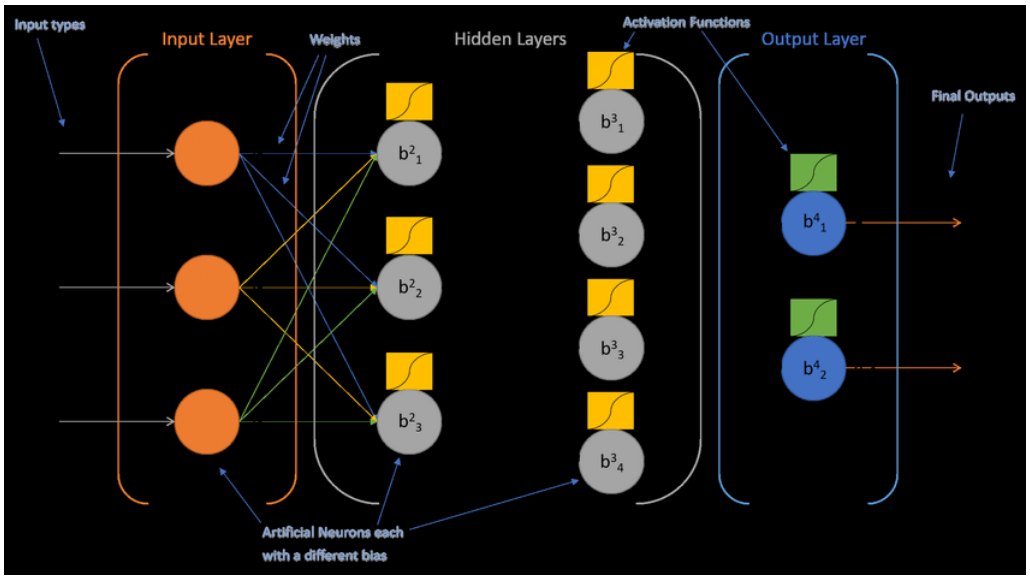

Figure 2.20: Visual representation of a Neural Network

Neural networks rely on more time, as they gradually improve their accuracy on the data provided with each generation of learning. Once the training is complete they algorithm becomes a powerful and fast tool. Google is known to be using neural networks in it's search algorithm. Another field where this algorithm excels is finding multiple patterns in between the layers of processing the data, which allows it to process complex data, such as images and videos, with the downside of the long periods of time needed to train on such data. Another application of this ability is on data where certain temporal connections can be made and certain information relies on different periods of time and impact the result differently. This feature will be later mentioned in this paper. Neural networks have two subsets, Convolutional neural networks and Recurrent neural networks. The first one is used in the cases of processing images, videos or finding complex patterns between the data, while the second is more used for identifying feedback loops, being mainly used for stock market predictions.

Convolutional Neural Networks

are a subset of neural networks and stand at the base of deep learning algorithms. They keep the threshold rule from neural networks, where certain calculations are made and if the result passes that threshold it is passed to the next layer/neuron. They add more layers to the initial concept, such as the Convolutional layer, max pooling layer and the dense layer, as we can see in this visual representation2.21.

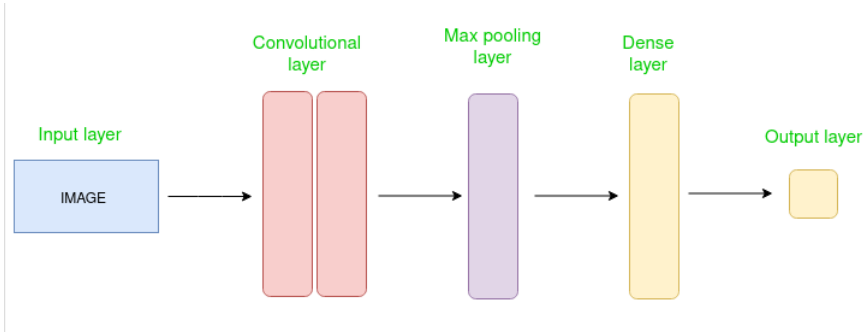

Figure 2.21: Visual representation of a Neural Network

Input layers are the medium through which we provide data to the model. Generally it receives images or sequences of images, but in the case of the presented project it will receive data about the soil. This layer holds the raw input data, which can include various soil properties such as moisture, pH level, and nutrient content across different time periods.

Convolutional layers are used to extract features from the input dataset. It uses Kernels, which are a set of applicable learnable filters, usually in the form of matrices, to process the data. hese filters slide over the input data and compute the dot product between the kernel weights and the corresponding input data patch. The result is referred to as feature maps.

Activation layers Activation layers introduce nonlinearity to the network by applying an activation function element-wise to the output of the previous layer. Common activation functions include ReLU (Rectified Linear Unit): max(0, x), Tanh, and Leaky ReLU. The activation function ensures that the network can capture complex patterns. The volume remains unchanged, so if the input volume is 32x32x12, the output will also be 32x32x12.

Pooling layers conduct dimensionality reduction, reducing the number of parameters in the input. In comparison to the convolutional layer, the pooling layer filter system has no weights, instead applying aggregation function on the values. This layer has two types of pooling: Max pooling, which, as the filter moves across the input, sends the max value to the output array, and Average Pooling, which sends the average value. This layer, although it loses information in the process, brings efficiency and reduces the complexity of the algorithm.

Fully-connected layers, also known as the dense layers, is where each neuron is connected to every neuron in the previous layer. This layer is used to combine all the features learned by the convolutional and pooling layers to make a final prediction. In the context of soil data, this layer can help predict outcomes such as soil fertility, crop suitability, or other relevant agricultural metrics based on the features extracted from the input data.

Chapter 3

Integrating AI into Agriculture

3.1 State of the art

As stated in the previous chapters, the three macronutrients, Nitrogen, Phosphorus and Potassium are the most influential factors when it comes to soil. Aside from the soil factors, the climate and weather also play a huge role. When it comes to the integration of machine learning in agriculture, due to the accessibility of the weather and climate data, there have been numerous initiatives that have sought to forecast crop production based on meteorologic data.

Artificial intelligence framework for modeling and predicting crop yield to enhance food security in Saudi Arabia

This project, [MHAA22], aims to ease the decision making in the agriculture field by using climate data. The project is using a dataset of temperature, insecticides, rainfall, and crop yields for potatoes, rice, sorghum, and wheat from 1994 to 2016 from Saudi Arabia in order to train the model to predict future crop yields and to highlight the impact these factors have on the crop yield.
As for the methods of training, the machine learning method used in this project was Artificial Neural Networks(ANN), mainly multilayer perceptron, which is a fully connected multi layer neural network.
The results are then processed through different statistical methods, such as the mean square error (MSE), the root-mean-square error (RMSE), normalized root mean square error (NRMSE), Pearson's correlation coefficient (R%), and the determination coefficient (R^2). The best performing MLP model is characterised by values of R = 100% and R^2 = 96.33.

Machine learning approach for forecasting crop yield based on climatic parameters

This project, [YC19], focuses on the prediction of wheat fields, as they are a very predominant crop in our agriculture industry. The project is situated in Australia and uses both climate data and satellite data, in combination to create a new and unique set of data with better accuracy. The period of time this data was extracted from was from 2000 to 2014
. They used a popular regression method (LASSO, as a benchmark) along with other machine learning algorithms, such as random forest, neural network and vector machine, to build the desired model. They used some special technologies to process the satellite data, such as enhanced vegetation index and solar-induced chlorophyll fluorescence. This helped the prediction of the productivity of the crops. The conclusion of the results is that combining climate and satellite data can enhance the performance of yield prediction, with R^2 reaching approximately 0.75.

Another approach showcases the applicability of AI in predicting recommendation of crops for certain data, using both data about the soil as well as informations about the different growing preferences of crops.

AI Crop Predictor and Weed Detector Using Wireless Technologies: A Smart Application for Farmers

An example showcasing the new approach is [ID20], where the aim of the project is providing recommendations in terms of crops to the farmers, based on different factors. These factors are collected using the IoT devices and WSN, which play a crucial role in collecting real-time data from the agricultural fields. This data includes temperature, annual precipitation, total available land size, past crop grown history, and other relevant resources.
The project also takes the provided data about the fields, as well as other historical data about the crops and uses Naïve Bayes algorithm for generating the recommendations. This method has achieved an accuracy of 89.29%, outperforming other algorithms like regression and support vector machines in this context.

In contrast to these approaches, with a more in depth study and research needed in the field of gathering data, we find models that use soil data in order to learn about agriculture district. This provides a more exact and influential information, from where we can find better accuracy in regards to the crops.

Statistical and machine learning methods for crop yield prediction in the context of precision agriculture

This study, [Bur22], focuses adding crop yield, soil properties, and topographic characteristics as data in order to improve agricultural management. By comparing different analytical techniques, the research aims to quantify these relationships using high-resolution data from multiple fields in Southwestern Ontario, Canada. The following is an analysis of the approach and methodologies used in this study.
The dataset used contains 145,500 statistics regarding plants such as corn and soybean yield, along with other characteristics, such as topographic and soil, with some key attributes being pH, organic matter,phosphorus, zinc, potassium, elevation.
It uses Multiple Linear Regression (MLR), Artificial Neural Networks (ANN), Decision Trees and Random Forests to study the predictions and accuracy . Random forest provided the best results, with values of R^2 of 0.85 for corn and 0.94 for soybeans.

Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review

This study, [AC18] tries to predict accurate estimation of crop yield and effective nitrogen management are pivotal in contemporary agriculture. Remote sensing (RS) systems have emerged as key tools in developing mechanisms that suport the decision making for modern farming systems, aiming to raise yield production and better the nitrogen utilization while minimizing expenses and negative environmental impact. RS-based methods, however, entail processing large volumes of remotely viewed data from diverse platforms, leading to a shift towards machine learning (ML) techniques.
This paper delves into research developments over the past 15 years focusing on ML-based approaches for precise prediction of the crop yield and the approximation of the nitrogen status. The swift advancement in sensing technologies and machine learning (ML) algorithms offers affordable and thorough solutions for enhanced crop and environmental monitoring, aiding informed decision-making in agriculture. This review underscores the potential of utilizing targeted sensor platforms and ML methods, integrating diverse sensor modalities with expert knowledge, and developing hybrid systems that combine various ML and signal processing techniques for precision agriculture (PA) applications in the near future.

3.2 My Vision

There have been countless attempts at using climate data and temperature as leading factors when it comes to predicting crop yield. While they are extremely important factors that highly influence the production, I wanted to base my paper on soil chemical balances, as they have an even greater impact on the plants. On top of that, the climate data has room for failure, taking into account the fact that farmers use irrigation methods when the rain is not enough.
With all that being said, the best outcome would be achieved if these two types of data would be used together. However, there are not that many datasets on soil data with crop production. In my research, I have managed to find, after a long search, a dataset that had everything I needed to prove the impact some chemical data collected at the start of the year, with some additional irrigation data, has on the prediction of crop yield. The dataset was done on Indian data and can be found here [Sur19].
The main purpose of this paper is to showcase the importance of soil data when integrating AI into agriculture, the potential accuracy increase if used together with the climate data and to stimulate further data collection, especially about fertilizers used. When all of these are combined, they could lead to a very intuitive project that could analyse all parameters and recommend farmers not only what crops they should plant, but also how much fertilizers they should use and in what periods of time.
This project has three different machine learning algorithms. The first one, extreme gradient boost, is used as it is the most efficient on the current data. The reason for building another machine learning algorithm on Convolutional Neural Networks has to do with the future expansion of the project. When more detailed data will be gathered and the notion of different times where the chemical data was collected from the soil is introduced, the gradient boost will slowly lose efficiency and accuracy in favor of the CNN model that could better understand the implications of time. The third algorithm's purpose is for attempting a new output with the given data, where instead of predicting crop yield based on soil data, we try to predict the values of the three most important chemicals in the soil based on all the other data. This is an attempt at exploring the possibilities of recommending values of the three chemicals necessary for the growth of different plants chosen, which could lead to the development of recommending crops and fertilizers to be used for optimal production.

3.3 Model Training

3.3.1 Data extraction

The dataset used, [Sur19], is a dataset containing agricultural data from the state of Rajasthan, India, which was gathered from the years 2018-2019. It is divided into four files, each containing different important information, in an organised way. For my project, I've only used one, as the fourth, containing prices of different crops, is not that relevant.
The first dataset focuses on crop metrics, such as the crop production, the area that was planted, the localisation of the plantation, the season and the plant used. A better visual representation of the types with the amounts of unique data and examples can be found below.

Crop Production Data

Column	Non-Null Count	Dtype	Unique Values
District	1000	object	10
Crop	1000	object	23
Season	1000	object	2
Area (hectares)	1000	float64	956
Yield (quintals)	1000	float64	1000
Production (metric tons)	1000	float64	1000

Sample Data

	District	Crop	Season	Area (hectares)	Yield (quintals)	Production (metric tons)
0	Jodhpur	Wheat	Kharif	16490.142459	30.691986	5061.152230
1	Kota	Gram	Rabi	16943.065614	45.818232	7763.013072
2	Jaipur	Wheat	Rabi	14297.539876	29.752629	4253.893981
3	Hanumangarh	Coriander	Rabi	19737.638447	42.026511	8295.040749
4	Hanumangarh	Citrus	Rabi	13591.576842	38.195933	5191.429648

Figure 3.1: Depiction of the crop Dataset statistics

The second dataset focuses on soil data, with information such as the values of the three main chemicals needed for plant growth, organic matter and soil type, as presented below.

Soil Analysis Data

Column	Non-Null Count	Dtype	Unique Values
District	1000	object	10
Soil Type	1000	object	8
pH Level	1000	float64	1000
Organic Matter (%)	1000	float64	1000
Nitrogen Content (kg/ha)	1000	float64	1000
Phosphorus Content (kg/ha)	1000	float64	1000
Potassium Content (kg/ha)	1000	float64	1000

Sample Data

	District	Soil Type	pH Level	Organic Matter (%)	Nitrogen Content (kg/ha)	Phosphorus Content (kg/ha)	Potassium Content (kg/ha)
0	Jaipur	Chalky (Calcareous)	6.546096	1.569807	27.931972	29.438438	42.782766
1	Bhilwara	Nitrogenous	6.832259	2.243018	22.263480	25.413455	37.644377
2	Jodhpur	Sandy	7.453182	2.662898	23.564182	13.014409	37.082003
3	Jaipur	Clay	8.019189	1.240327	15.839222	17.744206	42.758704
4	Jaipur	Sandy	8.100131	1.768419	27.942867	25.769504	30.651292

Figure 3.2: Image of the soil Dataset statistics

The third dataset contains information regarding the water availability and the irrigation methods, to provide a quantification of the amount of water the plantation has received.

Water Usage Data

Column	Non-Null Count	Dtype	Unique Values
District	1000	object	10
Crop	1000	object	23
Irrigation Method	1000	object	4
Water Consumption (liters/hectare)	1000	float64	1000
Water Availability (liters/hectare)	1000	float64	1000

Sample Data

	District	Crop	Irrigation Method	Water Consumption (liters/hectare)	Water Availability (liters/hectare)
0	Udaipur	Garlic	Tube Well	11609.330886	11335.919980
1	Bhilwara	Gram	Drip Irrigation	12648.406038	12091.900254
2	Udaipur	Guava	Canal Irrigation	10063.162448	9810.943169
3	Jodhpur	Barley	Sprinkler Irrigation	10110.539993	10754.217313
4	Alwar	Gram	Canal Irrigation	9320.451018	8695.537819

Figure 3.3: Picture of the water Dataset

These three datasets are then merged into one, so they can later be processed by the machine learning algorithms. The first two are merged on the common District column, while the third is merged with the first two based on both District and Crop columns, resulting in the desired merged dataset. In the next step, unnecessary columns are dropped, such as Production (metric tons) and Water Consumption

(liters/hectare). The Yield and Water availability are more than enough, making these two just another way of expressing the same data.
For a better visualisation of the numerical data from the dataset, we have created some graphical representations. To view the frequency with which certain values among the numerical data of the dataset are present, we have created a graphic view, that divides data into intervals(bins) and shows the frequency of data points falling into each bin. This helps us understand the shape and spread of the data. Alongside this is a box plot, showing the graphical summary of the distribution of a dataset, with the box representing the interquartile range and the line inside representing the median.

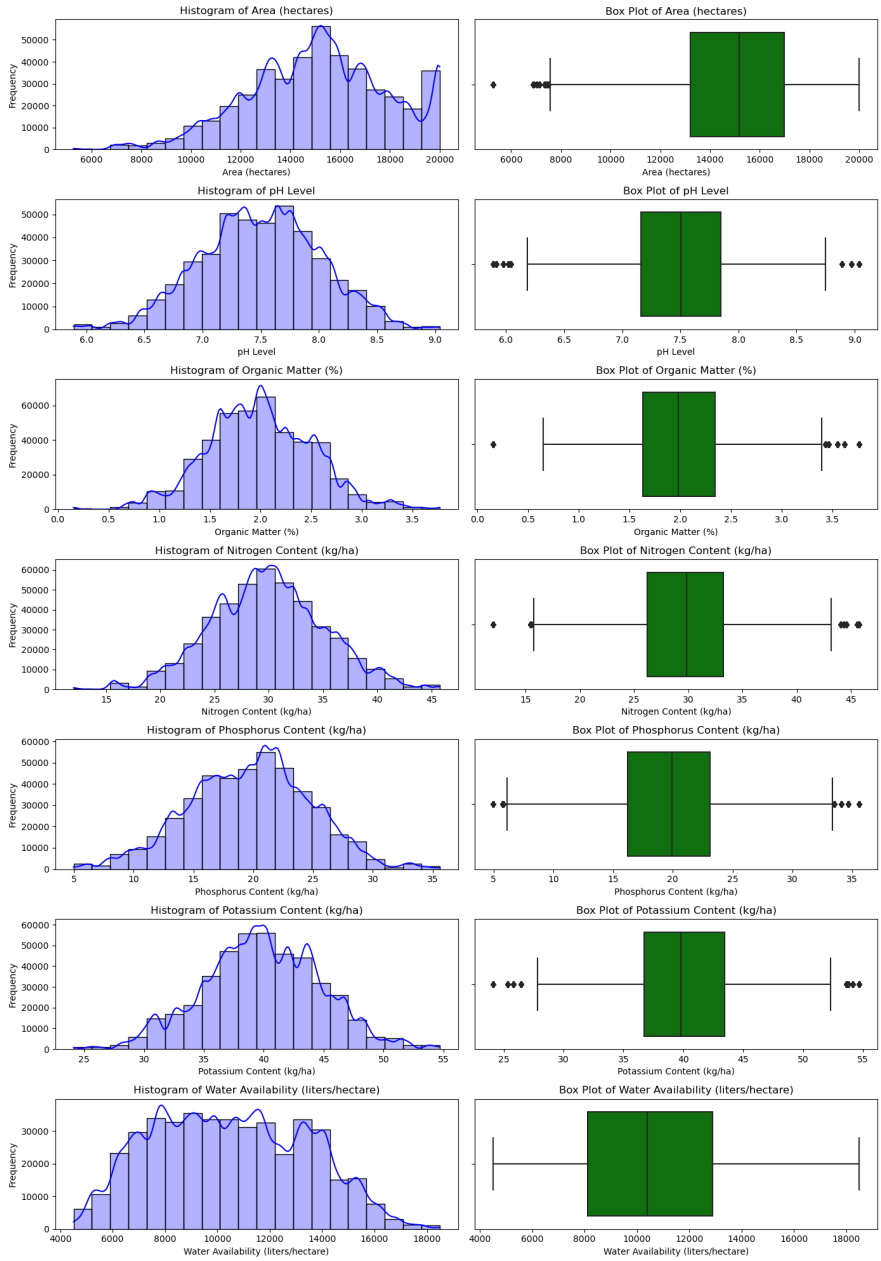

Figure 3.4: Visual representation of frequency and data spread

Each and every one of these factors have a high impact on the productivity of the crops. This is further reinforced by this Hexbin representation, where the gradient of the hexes describe the density of data points, all of them being represented in contrast with crop yield.

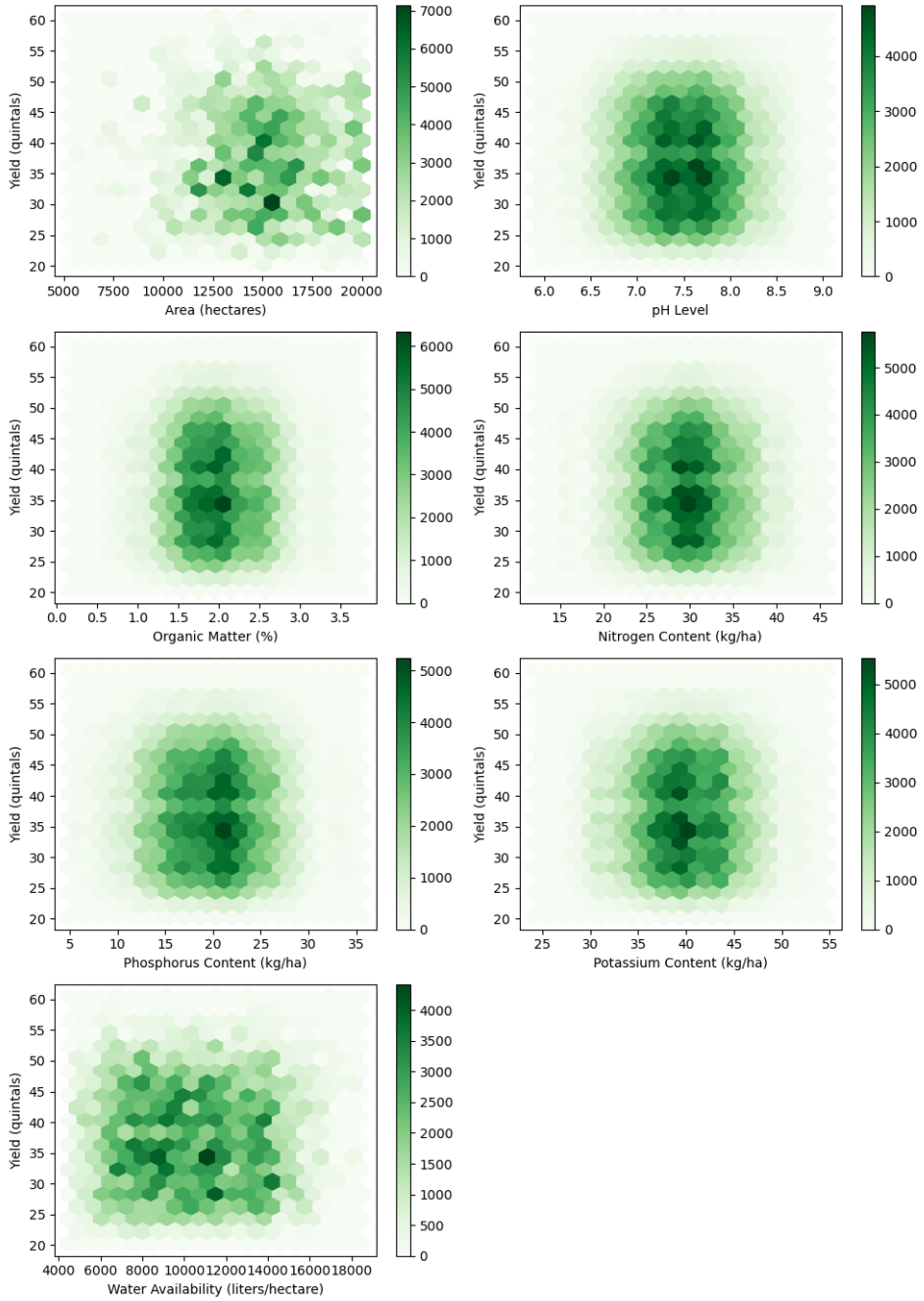

Figure 3.5: Diagram of density of data around crop yield

As we can see, around the values of Phosphorus, organic matter, Potassium and Nitrogen that the density converges to some central points, indicating potential correlations or optimal ranges that plants may favor for growth and yield. On the other hand, on water availability we can see more scattered central points of density on

the water availability, as different crops have a lot of different preferances regarding water.

3.3.2 Data transformation

After we have merged the datasets, we need to do some further transformation on the data to prepare it for the machine learning algorithm. Since there are 5 columns that have data containing characters(names), we have to encode them. We have two different methods of encoding for the two different machine learning algorithms.
For the XGB model, we split them into numerical columns, which we keep like this and categorical columns, which we convert into category datatype, so that we can use the feature of XGB which allows categorical data natively.

```
num_cols = ['Area (hectares)', 'pH Level', 'Organic Matter (%)', 'Nitrogen Content (kg/ha)', 'Phosphorus Content (kg/ha)',
       'Potassium Content (kg/ha)', 'Water Availability (liters/hectare)']
cat_cols = ['District', 'Crop', 'Season', 'Soil Type', 'Irrigation Method']
encoder = OneHotEncoder()
database[cat_cols] = database[cat_cols].astype("category")
```

Figure 3.6: XGB Encoder code

For the CNN model, we use the same OneHotEncoder, but this time, the one-hot encoded features are converted into Data Frame with appropriate column names. We continue by extracting the original numerical features and concatenating them to the one-hot encoded categorical features.

```
encoder = OneHotEncoder()
X_encoded = encoder.fit_transform(X[['District', 'Crop', 'Season', 'Soil Type', 'Irrigation Method']])
feature_names = encoder.get_feature_names_out(['District', 'Crop', 'Season', 'Soil Type', 'Irrigation Method'])
X_categorical = pd.DataFrame(X_encoded.toarray(), columns=feature_names)
X_numeric = X.drop(columns=['District', 'Crop', 'Season', 'Soil Type', 'Irrigation Method'])
#print(feature_names)
#print(X_categorical.iloc[0])
```

```
# Export encoder
```

```
joblib.dump(encoder, 'encoder.pkl')
```

```
['encoder.pkl']
```

```
# Merge encoded categorical features and numerical features
```

```
X_processed = pd.concat([X_categorical, X_numeric], axis=1)
X_processed.head()
```

Figure 3.7: XGB Encoder code

In both cases, the encoder is exported, as we need it later to process the natural information when receiving it from the frontend and to transform it into the encoded data that can be processed by the machine learning algorithm.

3.3.3 Extreme Gradient Boosting ALgorithm

The first approach at creating a machine learning algorithm uses extreme gradient boosting, as it is highly efficient with tabular data, which is what we have here. Since the data is not that complex and doesn't have intricate correlations between the information, it is a little better than the CNN module, at least in this step.
Before creating the algorithm, we must divide the data. For the most accurate result, the data was split into three: training, testing and validation. In order to avoid overfitting, the proportions of these three were chosen as 80% for training and 10% each for testing and validation.
As for the specifics of the xgb model used, we chose the tree method hist, as it has the ability to use categorical values, which are very much present in five of our columns.
After the predictions are done, we use Mean Squared Error to calculate he average of the squares of the differences between the actual and predicted values and Root Mean Squared Error to get the square root of the MSE, for an examination on the loss of the algorithm. For a better visualisation of these errors and to further confirm that the errors do not vary that much, thus, proving that the system is not overfitted, we can see the figures in the following representation.

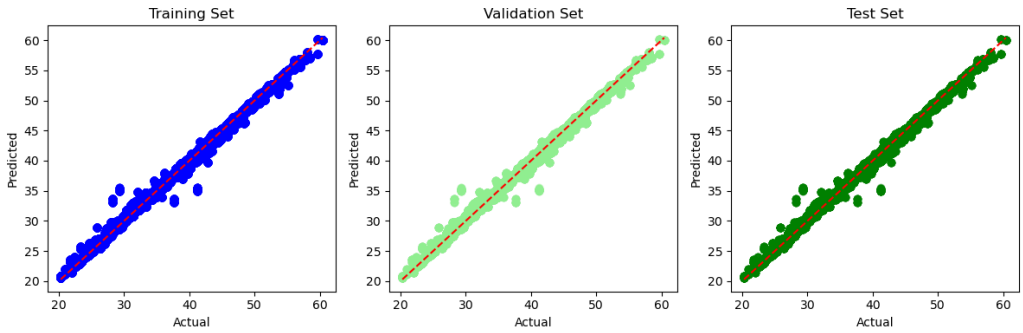

Figure 3.8: Visualisation of predictions versus actual values

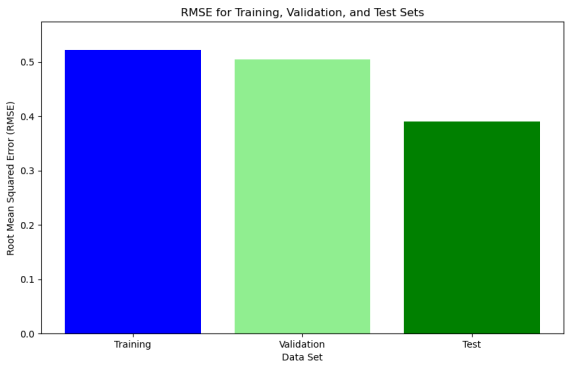

Figure 3.9: Comparison between the errors to statistically show the invariance

When it comes to the proportion of the variance explained by the model, this machine learning algorithm reaches the R^2 Score of 0.9978, an amazing performance on the provided data.

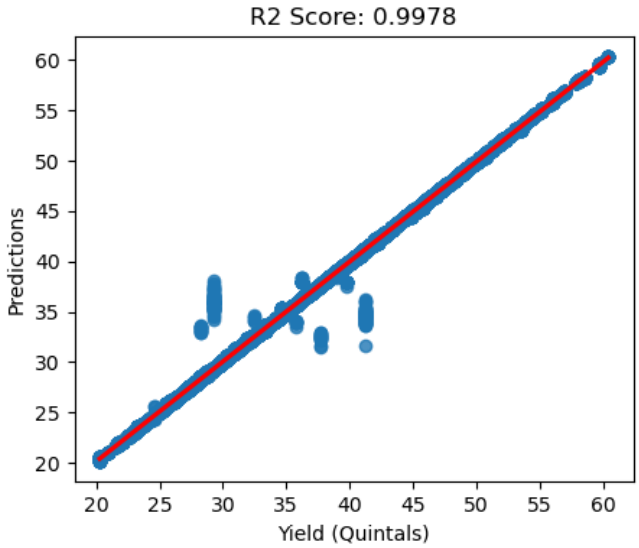

Figure 3.10: R^2 Score

To further improve the performance of the model, we apply some hyperparameter tuning before training. During training, we also employed early stopping, which halts the training process if the evaluation metric (RMSE in this case) does not improve for a specified number of rounds. This prevents the model from overfitting to the training data. After tuning the hyperparameters, I evaluated the model's performance on the test set. Even if the model was already performing great, we can see in the following visualisations that this helped, especially on reducing the error.

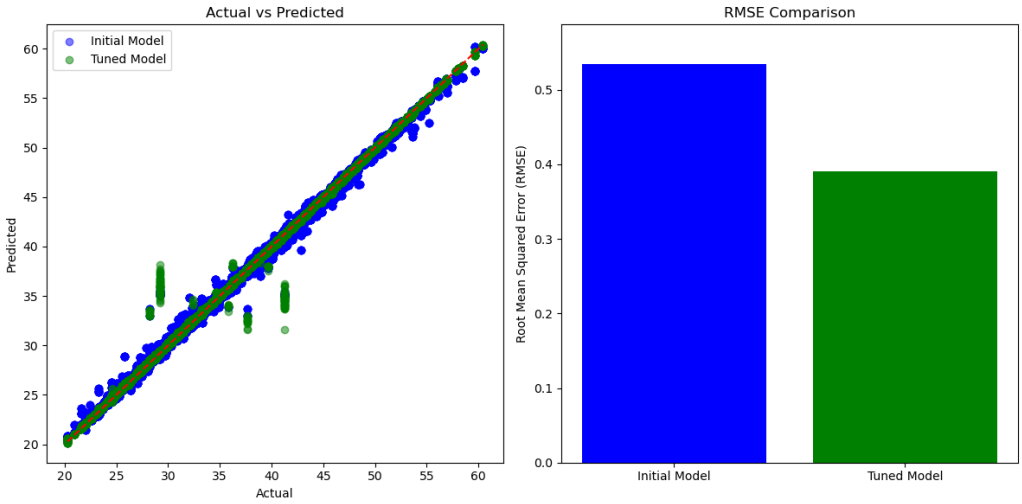

Figure 3.11: Before and after hyperparameter tuning comparison

3.3.4 Convolutional Neural Network Algorithm

Another method would be using Neural Networks. For preparing the data we had to once again encode the data. This time, to prepare it for the CNN algorithm, we had to transform them into an array. Then we followed a similar approach, only this time the proportions we chose in order to achieve the best results, which was 60% training, 20% validation and 20% testing. Although the error on this model was higher, reaching 3.52, the R^2 value gets close to the XGB model, with a beautiful score of 0.9552.

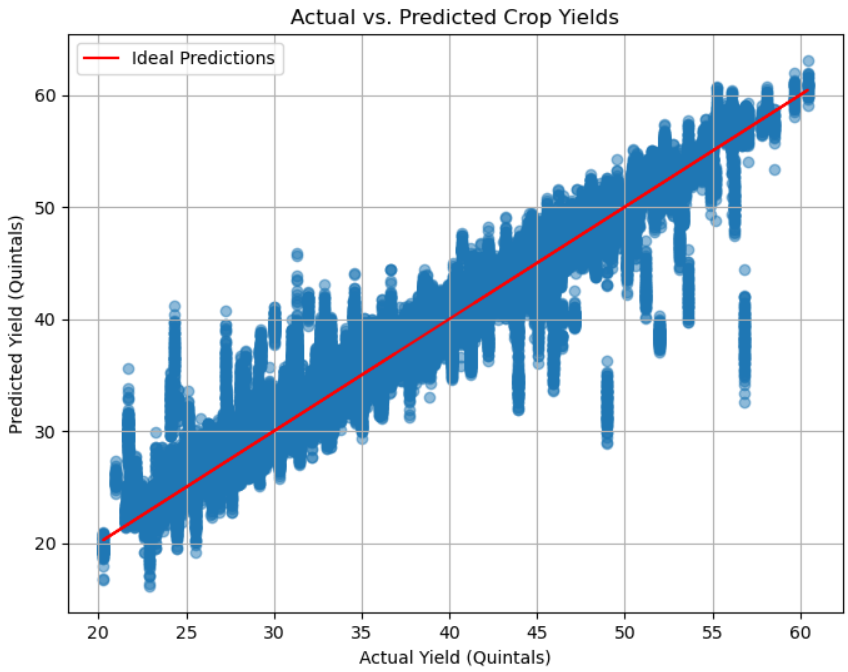

Figure 3.12: Actual versus predicted Crop Yields

This experiment had two purposes. The first one is to bring a comparison to show the efficiency the XGB model currently has over other models. The second reason is, as previously mentioned, to carve the path for future development, where multiple data from different periods of time would be collected and an algorithm able to handle more complex connections between the data would be necessary.

3.3.5 Fully Connected Neural Network Algorithm

Following the idea of experimenting for the purpose of future development, we created a new model, where we changed the desired result from crop yield to the three big chemical substances in the soil. This model uses Fully Connected Neural network in an attempt to guess these three chemical substances based on the other

data, with the crop yield added as input data. The purpose behind this is to create recommendations of balances between these chemicals in the soil in order to create prosperous conditions for the desired plant. Although there are multiple outputs and the data does not contain all desired inputs for an accurate response, the algorithm provided good results, as we can see in the visual representation below.

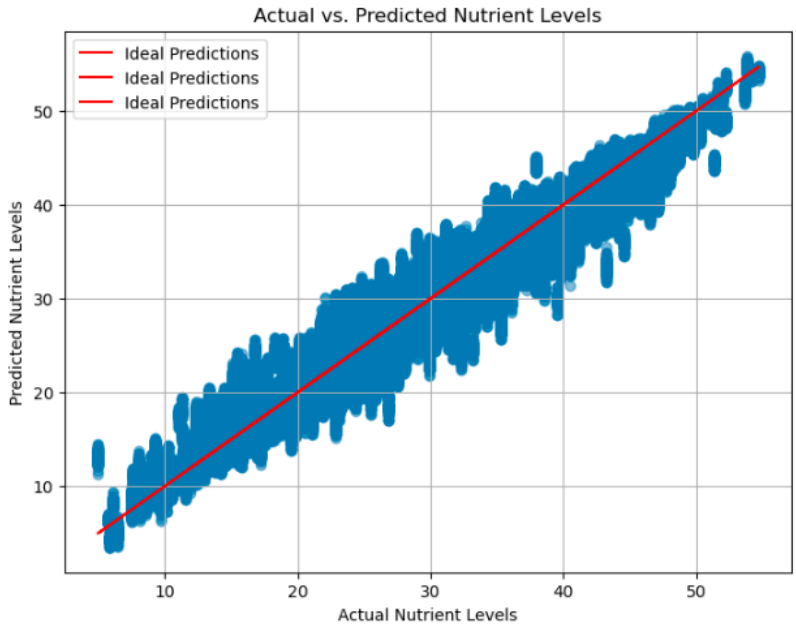

Figure 3.13: Actual versus predicted Chemical Trio

As a conclusion, the XGB model remains the most efficient on the current format of the data, while these other methods show promising results for future cases where they would be more applicable.

Chapter 4

Application building

For this application we built a complete structure, formed from database, server and Frontend, to enable future building on the project. For now, the processing of the input data is done directly into the requests of the server. This can later be improved by separating it and moving it to the Backend.

4.1 Database

4.1.1 Technology used

For building the database, we used the DataGrid IDE with Sqlite, as we have a relativelly simple and small dataset, which didn't necessitate the advanced features provided by more complex database systems like MySQL or PostgreSQL. Sqlite is a is a compact library that integrates directly into the application. This eliminated the need for a separate database server and simplified deployment and maintenance. It's ease of use, minimal configuration and portability made it a great choice for the structure of my project. For a visual representation of the database, we have prepared a schema of the tables.

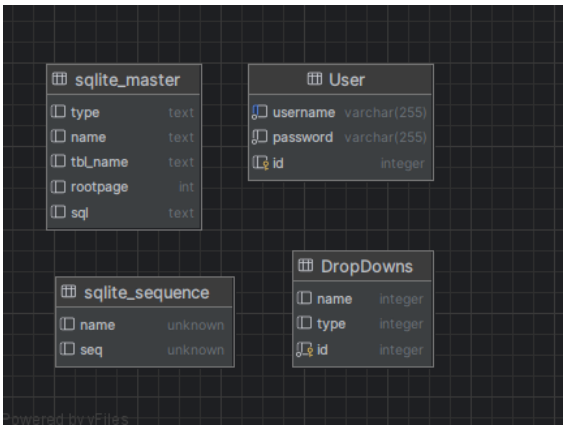

Figure 4.1: Database Schema

4.1.2 Code description

Although we don't have much information to be stocked outside the users data for the login, we decided to create a table for the possible options of the non numeric fields. We combine this with a Dropdown format on the Frontend with the data collected from the database through a request to assure that there are no issues regarding typing correctly and to make it easier for the user to navigate the application.

4.2 Server

4.2.1 Technology used

For the server, we chose to use Pycharm with the programming language Python. Python is renowned for its simplicity and readability, which makes the development process faster, especially for server-side applications and provides powerful tools and pre-built functionalities. FastAPI, in particular, stood out because of it's high performance and ease of use, being extremely intuitive. Pycharm's seamless integration together with the robust debugging capabilities made it a perfect candidate for an IDE on the server application.

```
def load_tensor_model(path):
    custom_objects = {'mse': tf.keras.losses.MeanSquaredError()}
    model = tf.keras.models.load_model(path, custom_objects=custom_objects)
    return model


def load_joblib_model(path):
    model = joblib.load(path)
    return model


cnn_model = load_tensor_model('..\..\MLAlgorithms\Project-CNN\model.h5')
rcnn_model = load_tensor_model('..\..\MLAlgorithms\Project-CNN-Regresion\Rmodel.h5')
xgb_model = load_joblib_model('..\..\MLAlgorithms\Project-XGB\model_xbg.pkl')
encoder = load_joblib_model('..\..\MLAlgorithms\Project-CNN\encoder.pkl')
scaler = load_joblib_model('..\..\MLAlgorithms\Project-CNN\scaler.pkl')
rscaler = load_joblib_model('..\..\MLAlgorithms\Project-CNN-Regresion\scaler.pkl')
```

Figure 4.2: Code of the integration of machine learning algorithms into the server

4.2.2 Code description

The server code is split into different code snippets for a more organised structure. First is the DataAccessLayer where we connect with the database, followed by the Machine Learning loaders, where we prepare two types of extracting the machine learning algorithms, one for the tensor models and one for the joblib models. Aside

from importing the models, we also had to import the Encoders and Scalers and send them to the Service so that we can handle the transformation of the data to the correct format that the machine learning algorithm demands.
The next code snippet, called **Service**, handles the requests for login, register, retrieval of the Dropdown data and the three machine learning algorithms. The login is done with a token so that we can handle the user changing pages and still remaining logged in without sending data about him to the Frontend.

```
@app.post("/login")
def login(user: Models.PUser, db: Session = Depends(get_db)):
    user_db = db.query(Models.User).filter(Models.User.username == user.username).first()
    db.close()
    if user_db and user_db.password == user.password:
        payload = {"username": user_db.username, "user_id": user_db.id}

        token = jwt.encode(payload, "your-secret-key", algorithm="HS256")

        return JSONResponse(content={"token": token}, headers={"Access-Control-Allow-Origin": "*"})
    raise HTTPException(status_code=401, detail="Invalid username or password")


@app.post("/register")
def register(user: Models.PUser, db: Session = Depends(get_db)):
    user_db = db.execute(select(Models.User).filter(Models.User.username == user.username)).first()
    if user_db:
        db.close()
        raise HTTPException(status_code=400, detail="Username already exists")
    db.add(Models.User(**user.dict()))
    db.commit()
    db.close()
    return JSONResponse(content={"message": "Registration successful"}, headers={"Access-Control-Allow-Origin": "*
```

Figure 4.3: Code representation of the Login and Register Request

When handling the machine learning requests, we must take multiple factors into consideration. The first one is the fact that the formatting done in Frontend cannot use spaces in the naming of it's variables, hence the need of column mapping to adjust the data received. The second one is the fact that we have to apply the Encoders and Scalers on the data to format it correctly before sending it to the machine learning algorithm, as we can see in the following example.

4.3 Frontend

4.3.1 Technology used

When developing the Frontend, the technologies we chose to use are Visual Studio with React, for multiple reasons. React, a widely-used JavaScript library for building user interfaces, offers a component-based architecture that promotes reusability and modularity. This structure makes it easier to manage and scale the application as it grows. React's virtual DOM ensures high performance by minimizing direct manipulations of the actual DOM, leading to faster and more efficient rendering.

```python
@app.post('/XGB-predict')
def xgb_predict(input_data: Models.PredictYieldData):
    try:
        input_dict = input_data.dict()
        column_mapping = {
            'Area_hectares': 'Area (hectares)',
            'Soil_Type': 'Soil Type',
            'pH_Level': 'pH Level',
            'Organic_Matter': 'Organic Matter (%)',
            'Nitrogen_Content': 'Nitrogen Content (kg/ha)',
            'Phosphorus_Content': 'Phosphorus Content (kg/ha)',
            'Potassium_Content': 'Potassium Content (kg/ha)',
            'Water_Availability': 'Water Availability (liters/hectare)',
            'Irrigation_Method': 'Irrigation Method'
        }
        input_dict = {column_mapping.get(key, key): value for key, value in input_dict.items()}
        input_df = pd.DataFrame([input_dict])
        cat_cols = ['District', 'Crop', 'Season', 'Soil Type', 'Irrigation Method']
        input_df[cat_cols] = input_df[cat_cols].astype("category")
        input_df.drop(columns=['Production (metric tons)', 'Water Consumption (liters/hectare)'], inplace=True,
                      errors='ignore')
        prediction = xgb_model.predict(input_df)
        return JSONResponse(content={"prediction": prediction.tolist()}, headers={"Access-Control-Allow-Origin":
    except Exception as e:
        raise HTTPException(status_code=500, detail=str(e))
```

Figure 4.4: Code representation of the XGB Request

I opted for Visual Studio Code as my integrated development environment (IDE) due to its powerful features, flexibility, and community support. VS Code provides excellent support for JavaScript and React through extensions such as the React extension pack. Furthermore, VS Code's customization options, including themes and extensions, enable a tailored development environment that meets specific project needs.

4.3.2 Code description

The code is structured in pages, namely the Home page, Crop Yield page and Chemical Components page, splitting the functionalities of the app. The Home page holds the login form and serves as a navigation page after the login was completed, as users can return here from any other page and can navigate to any of them using the buttons with an intuitive visual representation.

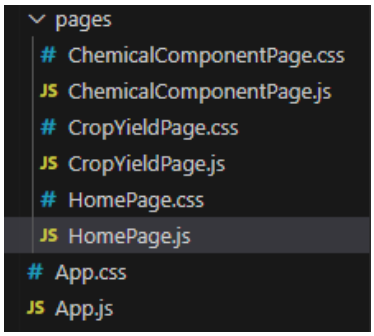

Figure 4.5: Visual representation of the frontend structure

In the App component we define the navigation through the React Router DOM, so we can navigate to all pages.

In the **Home page**, we handle the login and register of the user. With a press of a button, the user can change between register and login features, without changing pages. We have a form submission handling for login/register functionality. The app has

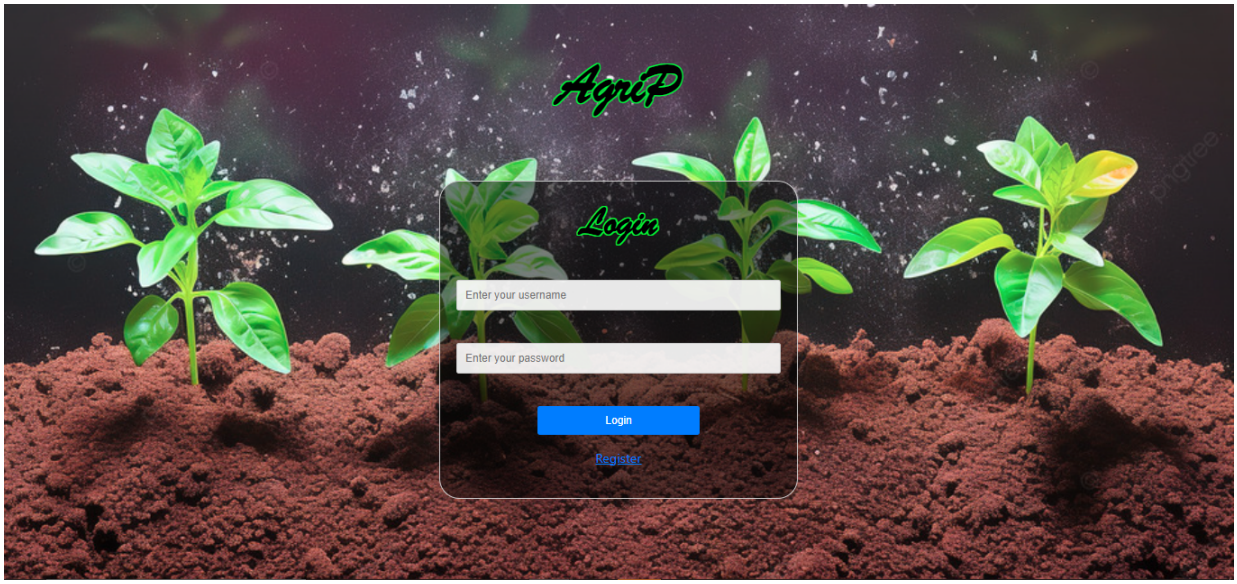

Figure 4.6: Visual representation of login page

After the login has been completed and the user receives a token, the form disappears with a fade out animation and the buttons for navigating to the other pages appear at the bottom left and right of the screen.

Figure 4.7: Visual representation of login page after receiving the login token

Through the button at the bottom right of the page, we can navigate to the **Crop Yield Prediction page**. Here, we have a form covering the entire page for the input necessary. The input fields containing names have been transformed into Dropdowns, to ease the completion process for the user and avoid mistakes like typos or case sensitive situations. The rest of the fields are coded to only receive numbers as input, to avoid invalid submissions for the numerical fields.
After the completion of the form, the user presses the submit button and the result will appear in the last box of the form, called Result. This field is unmodifiable in order to prevent the user from writing something in it.

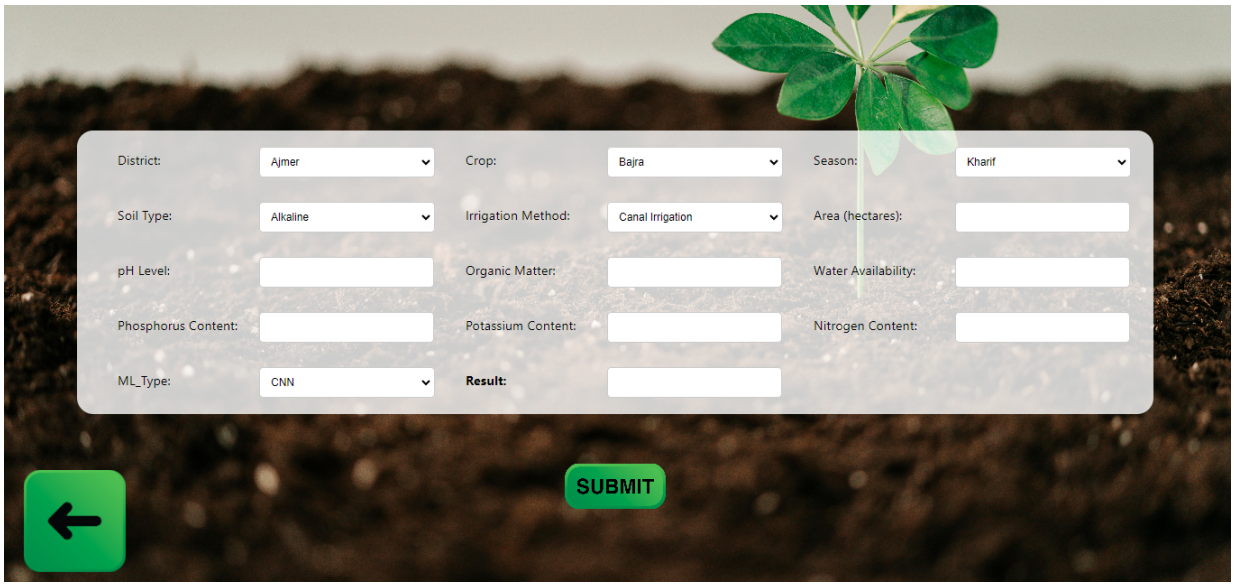

Figure 4.8: Visual representation of crop prediction page

In order to change the algorithm used, the user has one last Dropdown to the left of the result box, where he can choose if he wants to use the CNN algorithm or the XGB one.

After returning to the home page through the back arrow button, the user can navigate to the **Chemical Components Prediction Page** through the button at the bottom left of the screen. Here, he will be met with a similar form that needs to be completed before sending the data to the FCN algorithm and receiving the resulted predictions, which in this case are the three chemical components. Similar validation is applied through the Dropdown method and the numerical input fields receving only numbers as input.

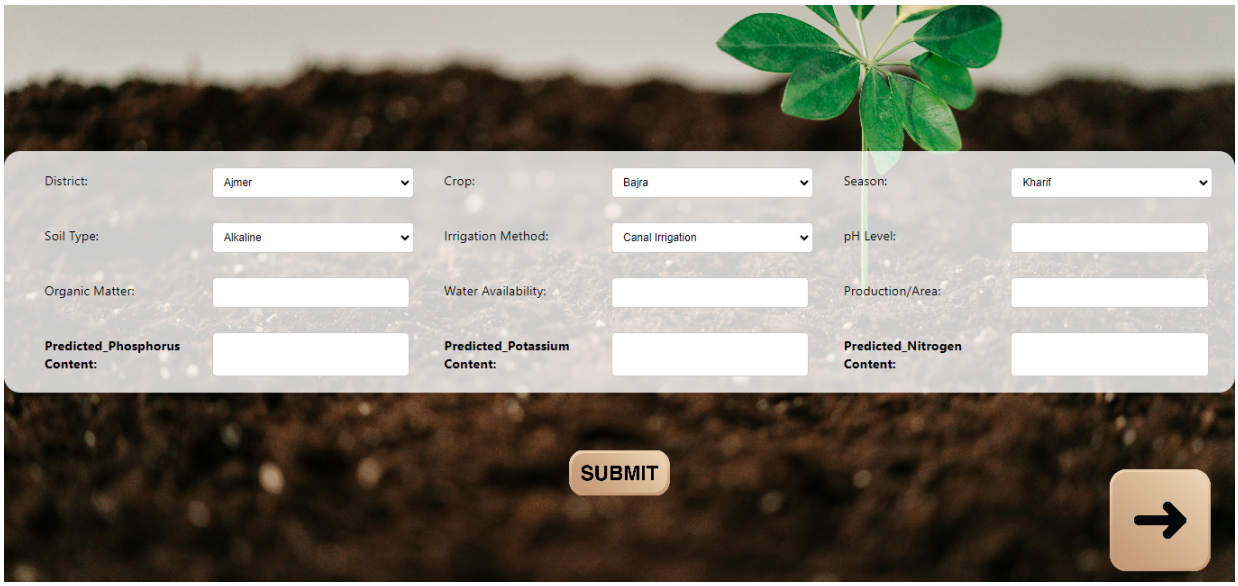

Figure 4.9: Visual representation of crop prediction page

After the completion of the form and the click of the submit button, the user can receive the optimal values of the three chemical components for the desired plant, in the highlighted unmodifiable boxes.

4.4 Similar aproaches

As previously stated, most of the papers on this subject revolve around AI predicting crop production using climate data. We wanted to focus on the impact of soil information, be it harder to get such data. These two can later be combined for the most accurate results, but for now, our focus was on soil and irrigation.
Even though most of other similar approaches focus on meteorological data, there are projects similar to ours, focusing on soil, for example the one we talked about in chapter 3.1, [AC18]. In this case, the focus is mainly on nitrogen and it's impact on the plantation. But there are three major chemicals, all with great impact on the crops. Not only are they a deciding factor in the productivity of the plant, but in some cases they even decide what type of plant should be used, as others would not survive at all. On top of this approach of integrating all chemical factors of the soil and taking irrigation into consideration, the CNN and FCN model we built with the purpose carving the path for the future of the project also touches a new idea in the field, which is monitoring these values throughout time and preparing for additional data like the fertilizer, which would greatly impact the nutrients in the soil

Chapter 5

Conclusions and future developments

5.1 Conclusions

The aim of this research project was first and foremost to encourage further development for integrating AI into agriculture, especially in the field of data gathering, which opens a lot of possibilities, by showing the potential results and accuracy of AI models.

By exploring different approaches, we were able to find the most efficient one for the structure of our data, namely the Extreme Gradient Boost algorithm, which has an accuracy of 99.75%. This proves the impact that soil and irrigation data has on the production of crops. This provides the farmers with statistical data over their productivity of the field in regards to the plant they are using, which helps them prepare for the harvest.

We provided an intuitive and lively application by combining Sqlite, Python and React for the Database, Server and Frontend for the user to easily navigate the functionalities provided by the machine learning algorithms.

5.2 Further improvements and developments

In terms of Further improvements and developments of the project, there could be small additions that would better the project like deploying the application or adding the Backend layer for a better structure and better security provided by the addition of another layer.

The major changes that would radically modify the functionality of this project would be on the data provided and the machine learning algorithms. By gathering information about the fertilizer used and it's quantity and also adding the temporal structure of the data, which would require us to bring the same data in different times of the year to help the algorithm better understand the variation and impact everything has, we would expand the project from a simple crop yield prediction model to an extremely useful project that would recommend the farmer the amount of fertilizer he should use in different periods of the year for the best productivity of the field. WE would do this with the help of the CNN or the FCN models, which have been tested and showcased their potential for growth in this paper.

The reason this project requires an AI for it's applicability in the real world stays in the high variation of data and it's impact on crops. For the same plant, different values of chemicals in the soil are needed depending on the climate, the region, the altitude and other factors. The machine learning algorithm could solve all these by learning specifically on that land and providing recommendations targeted to the specifics of the owned field.

Acknowledgments

We are deeply grateful to Nicu Sima for their invaluable guidance and mentorship in the field of agriculture. Their expertise and insights have been instrumental in deepening my understanding of the agricultural domain and its unique challenges. Their support has played a pivotal role in shaping my approach and finding a meaningful solution. Thank you for your unwavering support and encouragement.

We would like to also thank Angheloiu Dan, for his graphical design support provided for the frontend part of the application, as he contributed with all of the visual designs of the buttons used.

Bibliography

[AC18] Brett Whelan Anna Chlingaryan, Salah Sukkarieh. Statistical and machine learning methods for crop yield prediction in the context of precision agriculture. *ScienceDirect*, 2018. https://www.sciencedirect.com/science/article/pii/S0168169917314710?casa$_{t}ok$ $Zqo2xtZnfMcAAAAA:-Bf8i8KEjhdHFv0ARJLzrqCHn2t7mTl9TnOQyXlP$

[Ana] Analytic Steps. 6 Major Branches of Artificial Intelligence. `https://www.analyticssteps.com/blogs/6-major-branches-artificial-intelligence-ai`. Online; accessed 24 May 2024.

[Bur22] Wellen C. Burdett, H. Statistical and machine learning methods for crop yield prediction in the context of precision agriculture. *Springer Link*, 2022. https://doi.org/10.1007/s11119-022-09897-0.

[cyc13] The Phosphorus cycle. The phosphorus cycle. *Science learning hub*, 2013. https://www.sciencelearn.org.nz/resources/961-the-phosphorus-cycle.

[GEF] GEF. Land Degradation. `https://www.thegef.org/what-we-do/topics/land-degradation`. Online; accessed 21 April 2024.

[GRF] GRFC. 2023 Global report on food crises: Joint analysis for better decisions. `https://www.fsinplatform.org/sites/default/files/resources/files/GRFC2023-brief-EN.pdf`. Online; accessed 12 April 2024.

[Ian24] Flaviu Iancu. Romania se desertifica iar ministrul mediului constata: „am vazut cu ochii mei. daca nu actionam rapid, vom avea o clima foarte similara cu cea a greciei". *Pagina verde*, 2024. https://www.aktual24.ro/romania-se-desertifica-iar-ministrul-mediului-constata-am-vazut-cu-ochii-mei-daca-nu-actionam-rapid-vom-avea-o-clima-foarte-similara-cu-cea-a-greciei/: :text=a

[Ibe] Iberdrola. History of artificial intelligence. `https://www.iberdrola.com/innovation/history-artificial-intelligence#:~:text=American%20computer%20scientist%20John%20McCarthy,replace%20humans%20in%20assembly%20tasks`. Online; accessed 23 May 2024.

[ID20] Pattabiraman Venkatasubbu Parvathi Ramasubramanian Ishita Dasgupta, Jayit Saha. Ai crop predictor and weed detector using wireless technologies: A smart application for farmers. *Springer Link*, 2020. https://link.springer.com/article/10.1007/s13369-020-04928-2.

[MHAA22] Theyazn H.H. Aldhyani Mosleh Hmoud Al-Adhaileh. Artificial intelligence framework for modeling and predicting crop yield to enhance food security in saudi arabia. *Peerj*, 2022. https://peerj.com/articles/cs-1104/.

[NBVJB11] F. Nachtergaele, J. Bruinsma, John Valbo-Jørgensen, and Devin Bartley. *Anticipated trends in the use of global land and water resources*, page 12. 01 2011.

[oAF05] Ontario Ministry of Agriculture and Food. Environmental impacts of nitrogen use in agriculture. Technical report, Ontario Ministry of Agriculture, Food and Rural Affairs, November 2005. ISSN 1198-712X.

[oIiawTTBS20] The Fertilizer Association of Ireland in association with Teagasc Technical Bulletin Series. The importance of potassium (k) in agricultural soils. *Technical Bulletin Series*, 2020. https://www.fertilizer-assoc.ie/wp-content/uploads/2022/11/Technical-Bulletin-No.-5-Potassium.pdf.

[Pau] Paul Blocchi. The 6 Branches Of Artificial Intelligence. `https://www.linkedin.com/pulse/6-branches-artificial-intelligence-paul-blocchi/`. Online; accessed 23 May 2024.

[RRGM+23] Hannah Ritchie, Lucas Rodés-Guirao, Edouard Mathieu, Marcel Gerber, Esteban Ortiz-Ospina, Joe Hasell, and Max Roser. Population growth. *Our World in Data*, 2023. https://ourworldindata.org/population-growth.

[Sur19] Suraj(suraj520). Agricultural data for rajasthan, india (2018-2019), 2019. 05.06.2024.

[Tru24] Valentin Trufasu. România se transformă într-un deșert. ce spune un cunoscut cercetător despre cauzele fenomenului și cum îl putem combate. *Stiri interne*, 2024. https://adevarul.ro/stiri-interne/societate/romania-se-transforma-intr-un-desert-ce-spune-un-2316453.html.

[Wora] World bank. Agriculture and food. `https://www.worldbank.org/en/topic/agriculture/overview#1`. Online; accessed 12 April 2024.

[worb] worldometer. Romania Demographics. `https://www.worldometers.info/demographics/romania-demographics/`. Online; accessed 23 April 2024.

[YC19] David Lobell Andries B. Potgieter Shaowen Wang Jian Peng g Tianfang Xu Senthold Asseng Yongguang Zhang Liangzhi You Bin Peng Yaping Cai,

Kaiyu Guan. Integrating satellite and climate data to predict wheat yield in australia using machine learning approaches. *ScienceDirect*, 2019. https://www.sciencedirect.com/science/article/pii/S0168192319301224?casa$_{t}oken = Wspebf6B1OoAAAAA : UVyWYg6ZnyzJmhzuQUwSGlblijoNk6q6yxeD5Ot5TNGJx-V - MToUI8hG2rIwnqwySfG_{v}w - WGQ.$