

AI BASED SMART FARMING
Submitted in partial fulfillment of the required for the award of

Bachelor of Computer Science

By

KIRUBAKARAN B
(41290029)

&

JAGAN G
(41290025)



DEPARTMENT OF COMPUTER
SCIENCE SCHOOL OF SCIENCE AND
HUMANITIES

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A++" by NAAC

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI 600 199

APRIL 2024

AI BASED SMART FARMING
Submitted in partial fulfillment of the required for the award of

Bachelor of Computer Science

By

KIRUBAKARAN B
(41290029)

&

JAGAN G
(41290025)



DEPARTMENT OF COMPUTER
SCIENCE SCHOOL OF SCIENCE AND
HUMANITIES

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A++" by NAAC

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI 600 199

APRIL 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved b

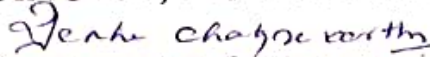
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

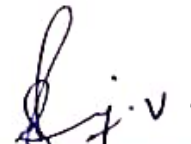
This is to certify that this Project Report is the bonafide work of
KIRUBAKARAN B (41290029) and **JAGAN G (41290025)** who carried out
the project entitled "**AI BASED SMART FARMING**" under our
supervision from NOVEMBER 2023 TO APRIL 2024.


Internal Guide

Ms. ANCY.A, M.Sc. Assistant Professor

DEAN AND HEAD OF THE DEPARTMENT
Dr. REKHA CHAKRAVARTHI M.E., Ph.D.

Submitted for Viva voce examination held on


Internal Examiner


External Examiner

DECLARATION

We, KIRUBAKARAN.B (41290029) and JAGAN.G (41290025) hereby declare that the Project Report entitled "AI BASED SMART FORMING" was done by us under the guidance of Ms. ANCY, A Department of Computer Science at SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, JeppiaarNagar, Rajiv Gandhi Salai, Chennai-600119 is submitted in partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Science.

KIRUBAKARAN.B
JAGAN.G

DATE: 27-04-24
PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to Dr. REKHA CHAKRAVARTHI M.E., Ph.D. Dean of Arts and Science for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our project guide Ms. ANCY A Department of Computer Science for their valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff member of the Department of Computer Science who were helpful in many ways for the completion of the project

JAGAN
KIRUBAKARAN

NO	Title.	Page No.
1	ABSTRACT	8
2	INTRODUCTION	9
3	OBJECTIVE	10
3.1	Requirements	10
4	LITERATURE SURVEY	11
5	EXISTING SYSTEM	14
5.1	Regression	15
5.2	Regression Trainers	16
5.3	Regression Tree Method	17
5.4	Fast Tree Regression Trainer	18
5.5	SOFTWARE ENVIRONMENT	19
6	PROPOSED SYSTEM	20
6.1	FEASIBILITY STUDY	22
6.2	ECONOMICAL FEASIBILITY	24
6.3	TECHNICAL FEASIBILITY	26
6.4	SOCIAL FEASIBILITY	29
7	EXPERIMENTAL OR MATERIALS AND METHODS ALGORITHMS USED	30
7.1	TYPES OF TESTS	32
7.2	Functional testing is centered on the following items	34
7.3	System Test	36
7.3.1	White Box Testing	38
7.3.2	Black Box Testing	40
7.4	Unit Testing	41

7.5	Integration Testing	41
7.6	Acceptance Testing	
8	RESULTS AND DISCUSSIONS	41
9	SCREENSHOTS	42
10	CONCLUSION	42
11	REFERENCES	43

ABSTRACT

Crop yield prediction is critical in agriculture management. To manage our health, we must monitor our cholesterol levels. In this, we use machine learning to predict crop yield in different areas. Machine learning is becoming increasingly important in a variety of fields around the world also very useful in the medical field. For yield management, early prediction is considered one of the most effective ways. To predict the level of crop yield, we use various parameters such as area, item, year, average rain fall per year pesticides tones and average temperature these details are collected by historical data. The data is all gathered from the agricultural sector, and it is given to the machine to train the data. Finally, the machine learning algorithms that forecast the crop yield by using many of features in plant growth.

A supervised machine learning task that is used to predict the value of the label from a set of related features. The label can be of any real value and is not from a finite set of values as in classification tasks. Regression algorithms model the dependency of the label on its related features to determine how the label will change as the values of the features are varied. The input of a regression algorithm is a set of examples with labels of known values. The output of a regression algorithm is a function, which you can use to predict the label value for any new set of input features.

Keywords: Artificial Intelligence, Machine Learning, Mobile Computing, Agriculture.

CHAPTER 1

INTRODUCTION

It is estimated that the world's population will be around 10 billion by 2050. And to meet the food needs of this growing population, there must be growth in the agricultural sector. At the moment, it is true that Artificial Intelligence exceeds human activity. As AI is an emerging technology in the agricultural sector the equipment based on AI, has taken the modern agricultural system to another level. These technologies have improved crop production and real-time monitoring, harvesting, and processing of cultivated crops way much easier. Recent technologies for automated systems, like use of agricultural robots and drones have made a significant contribution to the agricultural sector. The various hi-tech computer-based systems are designed to determine a wide range of important parameters such as weed control, crop detection, crop quality and many other strategies. Currently total area used for crop production is around 37.8%. There is a rapid acquisition of AI in agriculture using a variety of techniques. Managing effective farming practices with the help of new technological developments and solutions, is a requirement of the current situation. The use of new technologies available in Artificial Intelligence will help farmers to produce high quality crop production. Developments in the agricultural sector will also contribute to rural development. Currently there are number of techniques used in the agricultural sector such as disease detection, diagnostics, depending on the soil which fertilizer should be used, and much more. These technologies will be improving crop production and real-time monitoring, harvesting.

Integrating cutting-edge technologies into traditional agriculture, the AI Smart Farming project heralds a new era of precision and efficiency in agricultural practices. Leveraging artificial intelligence, sensor networks, and data analytics, this innovative initiative seeks to revolutionize the way crops are cultivated, monitored, and managed. By employing machine learning algorithms, the system analyzes vast datasets, offering real-time insights into soil health, crop conditions, and weather patterns. Autonomous drones equipped with advanced imaging capabilities contribute to the precision agriculture approach by capturing high-resolution imagery, enabling farmers to identify potential issues such as pests, diseases, or nutrient deficiencies.

The implementation of smart irrigation systems, driven by AI algorithms, optimizes water usage based on specific crop needs, promoting sustainability. Additionally, the project aims to enhance crop yield predictions, streamline supply chain logistics, and empower

farmers with actionable information, ultimately fostering a more resilient and productive agricultural ecosystem. The convergence of AI and farming not only promises increased productivity but also addresses critical challenges like resource conservation, making it a pivotal force in the future of sustainable agriculture.

OBJECTIVE

- Data set collection from various sources.
- Data parsing and cleansing technique is applied to make the raw data into processing data.
- The data collected is subject to machine learning system along with run time analysis makes an efficient crop value updating system.
- Usage of Fast Tree Regression Trainer makes the model more robust and efficient.
- Ranking technique used in the project helps us to make efficient decisions.
- The main objective is to obtain a better variety of crops that can be grown over the season. The proposed system would help to minimize the difficulties faced by farmers in choosing a crop and maximize the yield.
- The model predicts the crop yield by studying factors such as rainfall, temperature, area, season, soil type etc.

3.1 REQUIREMENTS

HARDWARE REQUIREMENTS

System	: Intel i3/i5
Hardware	: 40 GB
Monitor	: 15 Inches
Ram	: 4 GB

SOFTWARE REQUIREMENTS

Operating System	: Windows 7/8/10 (or) MAC OS
Programming Language	: ML.NET for C#
Tool	: Visual studio 2022

Title	Authors	Year
A Review on Data Mining Techniques for Fertilizer Recommendation	Jignasha M. Jethva, Nikhil Gondaliya, Vinita Shah	2018
A Survey on Data Mining Techniques in Agriculture	M.C.S.Geetha	2015
AgroNutri Android Application	S. Srija, R. Geetha Chanda, S.Lavanya, Dr. M. Kalpana Ph.D	2016
Machine Learning: Applications in Indian Agriculture	Karan deep Kaur	2016
Impacts of population growth, economic development, and technical change on global food production and consumption	Uwe A. Schneider a,†, Petr Havlik b, Erwin Schmid c, Hugo Valin b, Aline Mosnier b,c, Michael Obersteiner b, Hannes Bottcher b, Rastislav Skalsky' d, Juraj Balkovic' d, Timm Sauer a, Steffen Fritz b	2011

Brief history of agricultural systems modelling	James W. Jones a, , John M. Antle b, Bruno O. Basso c, Kenneth J. Boote a, Richard T. Conant d, Ian Foster e, H. Charles J. Godfray f, Mario Herrero g, Richard E. Howitt h, Sander Jansseni, Brian A. Keating g, Rafael Munoz-Carpena a, Cheryl H. Porter a, Cynthia Rosenczweig j, Tim R. Wheeler k	2016
A Smart Agricultural Modelby Integrating Iot, Mobile and Cloud-based Big Data Analytics	S.Rajeswari, K.Suthendran, K.Rajkumar.	2017
An Overview of Internet of Things and Data Analytics in Agriculture	Olakunle Elijah, Tharek Abdul Rahman, Igbafe Orikumhi, Chee Yen Leow, Nour Hindia	2018
Circulation Mode Selection Based on Cost Analysis	Xiurong Sun, Jingshan Zhang, Chenglin Wang, Tao Zhang	2017
Support Vector Machine-based Fuzzy Self-learning Control for Induction Machines	Zongkai Shao	2010

CHAPTER 2

LITERATURE SURVEY

[1] Title: A Review on Data Mining Techniques for Fertilizer Recommendation 2018.

Authors : Jignasha M. Jethva, Nikhil Gondaliya, Vinita Shah

To keep up nutrition levels in the soil in case of deficiency, fertilizers are added to soil. The standard issue existing among the Indian agriculturists choose approximate amount of fertilizers and add them manually. Excess or deficient extension of fertilizers can harm the plants life and reduce the yield. This paper gives overview of various data mining frameworks used on cultivating soil dataset for fertilizer recommendation.

[2] Title: A Survey on Data Mining Techniques in Agriculture, 2015.

Authors : M.C.S.Geetha

Agriculture is the most critical application area especially in the developing nations like India. Use of information technology in agriculture can change the situation of decision making and farmers can yield in better way. This paper integrates the work of several authors in a single place so it is valuable for specialists to get data of current situation of data mining systems and applications in context to farming field.

[3] Title : AgroNutri Android Application, 2016.

Authors : S. Srija, R. Geetha Chanda, S. Lavanya, Dr. M. Kalpana Ph.D

This paper communicates the idea regarding the making of AgroNutri an android application that helps in conveying the harvest particular fertilizer amount to be applied. The idea is to calculate the measure of NPK composts to be applied depend on the blanked proposal of the crop of interest. This application works depend on the product chosen by the farmer and that is taken as input, thus providing the farmers. The future scope of the AgroNutri is that GPRS can be included so that according to location nutrients are suggested.

[4] Title: Machine Learning: Applications in Indian Agriculture, 2016.

Authors: Karan deep Kaur

Agriculture is a field that has been lacking from adaption of technologies and their advancements. Indian agriculturists should be up to the mark with the universal procedures. Machine learning is a native concept that can be applied to every field on all inputs and outputs. It has effectively settled its ability over ordinary calculations of

software engineering and measurements. Machine learning calculations have improved the exactness of artificial intelligence machines including sensor based frameworks utilized in accuracy farming. This paper has evaluated the different uses of machine learning in the farming area. It additionally gives a knowledge into the inconveniences looked by Indian farmers and how they can be resolved using these procedures.

[5] Title: Impacts of population growth, economic development, and technical change on global food production and consumption, 2011.

Author: Uwe A. Schneider a, Petr Havlik b, Erwin Schmid c, Hugo Valin b, Aline Mosnier b,c, Michael Obersteiner b, Hannes Bottcher b, Rastislav Skalsky' d, Juraj Balkovic' d, Timm Sauer a, Steffen Fritz b

Throughout the following decades humanity will request more food from less land and water assets. This investigation evaluates the food production effects of four elective advancement situations from the Millennium Ecosystem Assessment and the Special Report on Emission Scenarios. Partially and jointly considered are land and water supply impacts from population development, and specialized change, and forests and agriculture demand request shifts from population development and economic improvement. The income impacts on nourishment request are registered with dynamic flexibilities. Worldwide farming area increments by up to 14% somewhere in the range of 2010 and 2030. Deforestation restrictions strongly impact the price of land and water resources but have little consequences for the global level of food production and food prices. While projected income changes have the highest partial impact on per capita food consumption levels, population growth leads to the highest increase in total food production. The impact of technical change is amplified or mitigated by adaptations of land management intensities

[6] Title: Brief history of agricultural systems modelling, 2016.

Author: James W. Jones a, John M. Antle b, Bruno O. Basso c, Kenneth J. Boote a, Richard T. Conant d, Ian Foster e, H. Charles J. Godfray f, Mario Herrero g, Richard E. Howitt h, Sander Janssen i, Brian A. Keating g, Rafael Munoz-Carpena a, Cheryl H. Porter a, Cynthia Rosenzweig j, Tim R. Wheeler k

Rural frameworks science creates information that enables analysts to consider complex issues or take educated farming choices. The rich history of this science represents the decent variety of frameworks and scales over which they work and have been

contemplated. Demonstrating, a basic apparatus in agrarian frameworks science, has been expert by researchers from an extensive variety of controls, who have contributed ideas and instruments over six decades. As agrarian researchers currently consider the "people to come" models, information, and learning items expected to meet the inexorably mind boggling frameworks issues looked by society, it is vital to check out this history and its exercises to guarantee that we stay away from re-innovation and endeavor to think about all elements of related difficulties. To this end, we outline here the historical backdrop of rural frameworks demonstrating and distinguish exercises discovered that can help control the structure and advancement of up and coming age of farming framework apparatuses and techniques. Various past occasions joined with generally innovative advancement in different fields have unequivocally added to the development of farming framework demonstrating, including improvement of process-based bio-physical models of yields and domesticated animals, factual models dependent on verifiable perceptions, and financial streamlining and reproduction models at family unit and local to worldwide scales. Attributes of rural frameworks models have changed broadly relying upon the frameworks included, their scales, and the extensive variety of purposes that spurred their advancement and use by specialists in various controls. Late patterns in more extensive joint effort crosswise over establishments, crosswise over orders, and between people in general and private segments recommend that the stage is set for the significant advances in rural frameworks science that are required for the up and coming age of models, databases, learning items and choice emotionally supportive networks.

[7] Title: A Smart Agricultural Model by Integrating Iot, Mobile and Cloud-based Big Data Analytics, 2017.

Authors: S.Rajeswari, K.Suthendran, K.Rajkumar.

In the cultivating field, the system models play a significant role to the enhancement of the agro-normal and money related conditions. In the proportions of benefits of the field and farm examinations to give the information and to recognize fitting and fruitful organization practices. It can recognize the organization to arrive managers and transversely over reality as long as the required soil, the board, environment, and money related information. Decision Support Systems (DSSs) use to make the information for the vermin the board, develop the officials. These systems are not using the impelled strategies to process the data. Thusly, use the adroit system thoughts to take the decisions for the issue. It expects a crucial activity in the comprehension of agronomic results, and their use as decision sincerely steady systems for farmers is extending.

[8] Title: An Overview of Internet of Things and Data Analytics in Agriculture: Benefits and Challenges, 2018.

Authors: Olakunle Elijah, Tharek Abdul Rahman, Igbafe Orikumhi, Chee Yen Leow, Nour Hindia.

A blueprint of Iot and DA in agriculture has been shown in this paper. A couple of zones related to the association of Iot in agribusiness have been discussed in detail. The investigation of composing exhibits that there are clusters of work advancing being produced of Iot development that can be used to increase operational efficiency and gainfulness of plant and creatures. The benefits of Iot and DA, and open troubles have been identified and inspected in this paper. Iot is depended upon to offer a couple of benefits to the agribusiness division. Regardless, there are up 'til now different issues to be steered to make it moderate for close to nothing and medium-scale farmers. The key issues are security and cost. It is typical that as contention increases in the cultivating part

[9] Title: Circulation Mode Selection Based on Cost Analysis, 2017.

Authors: Xiurong Sun*, Jingshan Zhang, Chenglin Wang, Tao Zhang

If every farmer and each average production base will join their optimal conditions in making cooperatives, it will accomplish economies of scale. Furthermore, producers will have an all the more favorable position in the plans with downstream firms (shipper or retailer). Second, the main customers of wholesale market are not inhabitants nearby who buy small quantities products but lower distributors or retailers. More redesigned transportation mode respects intensive attempt of new agrarian things, which prompts bolster the movement of new chain joint logistics and strengthen resource utilize and made logistics advantage quality. Refresh everything considered agrarian things spread. By then, regard the examination of gigantic worth control of standard things and achieve the mind blowing control to stream process.

[10] Title: Support Vector Machine-based Fuzzy Self-learning Control for Induction Machines, 2010.

Authors : Zongkai Shao

Using support vector machine (SVM) is to realize the self learning of fuzzy inference system (FIS), based on a fast modified varying metric method (MDFP) and a support

vector machine identifier (SVMI), a SVM-FIS self-learning controller for the three phase induction machine adjustable speed system has been designed. The proposed controller is not only of the advantages that FIS does not depend on the plant model, strong robustness, and adaptive self-learning ability, but also learning ability and generalization performance of SVM. The designed processes of SVM-FIS, MDFP, and SVMI algorithms have been described in details. Simulation results show the feasibility, correctness and effectiveness of the proposed control strategy, such as the excellent static and dynamic performances, and strong anti-interference ability.

[11]Title: Machine Learning Facilitated Rice Prediction in Bangladesh, 2015.

Authors: Mohammad Motiur Rahman, Naheena Haq, Rashedur M Rahman

In this examination, self organizing map (SOM) was utilized to group the information relationship between the information factors. After that chi-square test strategy was utilized to set up the level of reliance between the related variable qualities. It was discovered that the day by day outrageous climate conditions, for example, most extreme and least fluctuation in temperature, precipitation, dampness and wind speed were the principle drivers of product development, yield and wine quality

[12]Title: Support Vector Machine-Based Classification Scheme of Maize Crop, 2017.

Authors: Suhas S Athani, CH Tejeshwar

This paper says about, advancement of a mechanized framework to distinguish and group weeds from the products would be of extraordinary help and we have proposed a set-up that decreases labor. We have considered pictures of maize edits as the informational index. Separating surface highlights of the picture and applying SVM (support vector machine) to arrange whether the given picture is a weed or a yield brought about a precision of 82%. The proposed framework gives a chance to investigate more about element extraction methods.

[13]Title: WITH MACHINE LEARNING ALGORITHMS FOR ESTIMATING WINTER WHEAT AREAS, 2017.

Authors: Y.Z. Pan2

We utilize different kernel functions in the CPPI models to depict the connection between fractional winter wheat area and MODIS EVI time series data. We tried three straight and

non-direct kernel functions, including linear regression, artificial neural system, and support vector machine.. For areas like DT where multiple crop types have comparative phenology cycles, ANN-CPPI is found to play out the best. The two crop types to be specific winter wheat and rapeseed, can be separated well. These tests give elective answers for the uses of CPPI in mixed areas.

CHAPTER 4

PROPOSED SYSTEM

In proposed system, the data analysis technology is used to update the crop yield rate change. The concept of this paper is to implement the crop selection method so that this method helps in solving many agriculture and farmers problems. This improves our Indian economy by maximizing the yield rate of crop production. Different types of land condition. So the quality of the crops are identified using ranking process. By this process the rate of the low quality and high quality crop is also intimated. The usage of AI Smart Farming system represents a groundbreaking integration of cutting-edge technologies to revolutionize traditional agricultural practices. Leveraging the power of artificial intelligence, sensor networks, and data analytics, this innovative project aims to enhance agricultural efficiency, sustainability, and yield. The system incorporates AI algorithms to analyze real-time data from various sensors deployed across the farm, enabling precise monitoring of soil conditions, crop health, and weather patterns. Through intelligent decision-making processes, the system provides timely insights and recommendations to farmers, optimizing resource utilization and minimizing environmental impact. Additionally, the project integrates machine learning models to predict crop diseases, pests, and optimal harvesting times, empowering farmers to take proactive measures and improve overall crop quality. The implementation of automated machinery and robotic systems further streamlines tasks such as planting, irrigation, and harvesting, reducing labor requirements and increasing overall productivity. This comprehensive AI Smart Farming solution not only addresses current challenges in agriculture but also paves the way for a sustainable and technologically advanced future for the global farming community.

CHAPTER 3

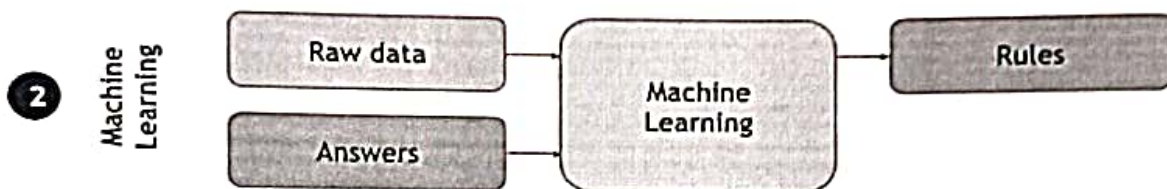
EXISTING SYSTEM

Integrating AI into existing farming systems can revolutionize agriculture, enhancing efficiency, productivity, and sustainability. This smart farming project employs cutting-edge technologies to optimize resource utilization and crop management. By leveraging AI algorithms, sensor networks, and data analytics, the system enables real-time monitoring of soil conditions, weather patterns, and crop health. AI-driven predictive models provide actionable insights, empowering farmers to make informed decisions regarding irrigation schedules, fertilizer application, and pest control. Autonomous drones equipped with AI vision systems contribute to precision agriculture, allowing for the early detection of crop diseases and efficient crop monitoring. Furthermore, the implementation of machine learning algorithms assists in crop yield prediction, helping farmers plan harvests and manage supply chains more effectively. The integration of AI into existing farming infrastructure not only enhances operational efficiency but also promotes sustainable practices by minimizing resource wastage and environmental impact. Overall, this AI smart farming project transforms traditional agriculture into a technologically advanced, data-driven system, fostering increased yields and promoting the long-term resilience of global food production.

CHAPTER 5

EXPERIMENTAL OR MATERIALS AND METHODS; ALGORITHMS USED.

AI based crop recommendation uses supervised machine learning technique to predict the crop suitable for the field based on the pH, Temperature, and Humidity.



5.1 Regression

A supervised machine learning task that is used to predict the value of the label from a set of related features. The label can be of any real value and is not from a finite set of values as in classification tasks. Regression algorithms model the dependency of the label on its related features to determine how the label will change as the values of the features are varied. The input of a regression algorithm is a set of examples with labels of known values. The output of a regression algorithm is a function, which you can use to predict the label value for any new set of input features.

Examples of regression scenarios include:

- Predicting house prices based on house attributes such as number of bedrooms, location, or size.
- Predicting future stock prices based on historical data and current market trends.
- Predicting sales of a product based on advertising budgets.

5.2 Regression Trainers

You can train a regression model using the following algorithms:

- LBFGS Poisson Regression Trainer
- Light GBM Regression Trainer
- SDCA Regression Trainer
- OLS Trainer
- Online Gradient Descent Trainer
- Fast Tree Regression Trainer
- Fast Tree Tweedie Trainer
- Fast Forest Regression Trainer
- Gam Regression Trainer

Regression Inputs and Outputs

The input label column data must be Single/float. The trainers for this task output the following:

Output Name	Type	Description
Score	Single/float	The raw score that was predicted by the model

5.3 Regression Tree Method

Regression trees are a non-parametric regression method that creates a binary tree by recursively splitting the data on the predictor values.

The splits are selected so that the two child nodes have smaller variability around their average value than the parent node. Various options are used to control how deep the tree is grown. Regression predictions for an observation are based on the mean value of all the responses in the terminal node.

5.4 Fast Tree Regression Trainer

Fast Tree is an efficient implementation of the MART gradient boosting algorithm. Gradient boosting is a machine learning technique for regression problems. It builds each regression tree in a step-wise fashion, using a predefined loss function to measure the error for each step and corrects for it in the next.

You Learn How To:

- ✓ Prepare and understand the data
- ✓ Load and transform the data
- ✓ Choose a learning algorithm
- ✓ Train the model
- ✓ Evaluate the model
- ✓ Use the model for predictions

Step 1: Create a console application

- Create a C# Console Application called "Artificial Intelligence _Regression_Crop_Recommendation".
- Choose .NET 6 as the framework to use. Click the Create button.
- Create a directory named Data in your project to store the data set and model files.
- Install the Microsoft.ML and Microsoft.ML.FastTree NuGet Package:

In Solution Explorer, right-click the project and select Manage NuGet Packages. Choose "nuget.org" as the Package source, select the Browse tab, search for Microsoft.ML, select the package in the list, and select the Install button. Select the OK button on the Preview Changes dialog and then select the I Accept button on the License Acceptance dialog if you agree with the license terms for the packages listed. Do the same for the Microsoft.ML.FastTree NuGet package.

Step 2: Prepare and understand the data

- In Solution Explorer, right-click on the project name and select Add → New Folder → Rename the folder as "Data". Once data folder was created, right-click on the Data

folder and select Add → Existing Items → select All files(*.*) → now select the Crop_recommendation.csv file.

In Solution Explorer, right-click the Crop_recommendation.csv file and

- Select Properties. Under Advanced, change the value of Copy to Output Directory to Copy if newer.
- Open the Crop_recommendation.csv data set and look at column headers in the first row. Take a look at each of the columns. Understand the data and decide which columns are features and which one is the label.

The label is the column you want to predict. The identified Features are the inputs you give the model to predict the Label.

The provided data set contains the following columns:

- Temperature: The temperature is a feature.
- Humidity: The humidity is a feature.
- ph: The ph is a feature.
- Label: The label is the label going to be predicted.

Step 3: Program

- Create data classes
- Define data and model paths
- Initialize variables
- Train the model

The Train() method executes the following tasks:

- Loads the data.
- Extracts and transforms the data.
- Trains the model.
- Returns the model.

- Evaluate the model

The Evaluate() method executes the following tasks:

- Loads the test dataset.
- Creates the regression evaluator.

- Evaluates the model and creates metrics.
- Use the model for predictions

The Test Single Prediction() method executes the following tasks:

- Creates a single comment of test data.

```
var CropSample = new CropDetails()
{
    // here you can enter the sample data to be predicted
    temperature = 18.87614998f, humidity
    = 24.54038287f,
    ph = 5.724242065f,
    // Just put 0f label = 0f
};
```

- Predicts label based on test data.
- Combines test data and predictions for reporting.

Displays the predicted results.

5.5 SOFTWARE ENVIRONMENT

C Sharp (C#):

C# (pronounced "C sharp") is a powerful, modern, object-oriented programming language developed by Microsoft as part of its .NET initiative. It was designed to be simple, type-safe, and versatile. Here's an in-depth explanation of various aspects of C# programming:

i. Object-Oriented Programming (OOP):

- C# is deeply rooted in OOP principles. It supports concepts like encapsulation, inheritance, and polymorphism.
- Encapsulation hides the internal state of an object and only exposes the necessary functionalities through methods and properties.
- Inheritance allows classes to inherit behavior and attributes from parent classes, promoting code reusability.
- Polymorphism enables objects to be treated as instances of their parent class, facilitating flexibility and extensibility in code design.

ii. Syntax:

- C# syntax is similar to other C-style languages like C++ and Java.
- Statements end with a semicolon (;), and code blocks are enclosed in curly braces ({ }).
- Main method: Every C# program begins execution with a static method called **Main**, which serves as the entry point for the application.
- Variables: Strongly typed variables are declared using keywords such as **int**, **string**, **bool**, etc.
- Control structures: C# supports various control structures like **if-else**, **switch**, **while**, **for**, **for each**, etc.

iii. Data Types:

- C# supports both primitive data types (**int**, **float**, **double**, **char**, **bool**) and reference types (classes, interfaces, delegates, arrays).
- It also supports null label value types, allowing variables to store null in addition to their usual value.

iv. Memory Management:

- C# incorporates automatic memory management through a process called garbage collection. Developers don't need to explicitly allocate or deallocate memory; the runtime handles memory management automatically.

v. Exception Handling:

- C# provides robust support for exception handling using **try-catch-finally** blocks.

- Exceptions are objects that represent errors that occur during program execution. They can be caught and handled gracefully to prevent application crashes.

vi. Common Language Runtime (CLR):

- C# code is executed by the Common Language Runtime (CLR), which provides various services like memory management, exception handling, and security.
- CLR enables interoperability between different .NET languages like C#, VB.NET, and F#.

vii. Asynchronous Programming:

- C# supports asynchronous programming using `async` and `await` keywords, allowing developers to write non-blocking code for tasks such as I/O operations and web requests.

viii. LINQ (Language Integrated Query):

- LINQ is a powerful feature in C# that provides a uniform way to query data from different sources like collections, databases, XML, etc., using a syntax similar to SQL queries.

ix. .NET Framework and .NET Core/.NET 5+:

- C# is primarily used in the .NET ecosystem, which consists of a vast framework library for building various types of applications, including desktop, web, mobile, and cloud-based applications.
- With the evolution of the platform, .NET Core and later .NET 5+ have emerged, offering cross-platform capabilities and improved performance.

x. Tooling:

- C# development is typically done using integrated development environments (IDEs) like Visual Studio, Visual Studio Code, or JetBrains Rider, which provide robust tools for writing, debugging, and testing C# code.

History of C Sharp:

C# (pronounced "C sharp") was developed by Microsoft and first released in 2000 as part of the .NET framework. Here's a brief history:

i. Origins:

- C# was created by Anders Hejlsberg, who previously worked on Borland's Turbo Pascal and Delphi programming languages.
- It was designed as a modern, object-oriented programming language for building robust and scalable applications on the .NET platform.

ii. Initial Release:

- The first version of C# (C# 1.0) was released in 2000 as part of the initial version of the .NET Framework.
- It introduced key features such as garbage collection, type safety, and simplified syntax compared to languages like C++.

iii. Evolution:

- Subsequent versions of C# introduced significant enhancements and features:
 - C# 2.0 (2005) introduced generics, partial types, and anonymous methods.
 - C# 3.0 (2007) brought language enhancements such as LINQ (Language Integrated Query), lambda expressions, and extension methods.
 - C# 4.0 (2010) introduced dynamic typing, named and optional arguments, and improved COM interoperability.
 - C# 5.0 (2012) introduced asynchronous programming with the **sync** and **await** keywords.
 - C# 6.0 (2015), C# 7.0 (2017), C# 7.1 (2017), C# 7.2 (2018), and C# 7.3 (2018) introduced various language enhancements and features like string interpolation, pattern matching, local functions, and more.
 - C# 8.0 (2019) introduced null label reference types, asynchronous streams, default interface methods, and more.
 - C# 9.0 (2020) introduced record types, pattern matching enhancements, top-level programs, and more.

iv. Open Source:

- In 2014, Microsoft announced plans to open-source the .NET platform, including the C# compiler and runtime.
- This move led to the creation of the .NET Core framework, which allowed C# developers to build cross-platform applications targeting Windows, Linux, and macOS.

v. .NET 5 and Beyond:

- With the release of .NET Core 3.0 in 2019 and subsequent versions, Microsoft unified the .NET platform under the .NET 5 brand.
- .NET 5, released in 2020, was the first step towards a unified .NET platform that supports desktop, web, cloud, mobile, and IoT applications.
- Microsoft continues to evolve C# and the .NET platform, with regular releases and updates to improve performance, developer productivity, and platform capabilities.

Throughout its history, C# has become one of the most popular programming languages worldwide, widely used for building a wide range of applications, including desktop software, web applications, mobile apps, cloud services, and games.

CHAPTER 6

SYSTEM STUDY

6.1 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ **ECONOMICAL FEASIBILITY**
- ◆ **TECHNICAL FEASIBILITY**
- ◆ **SOCIAL FEASIBILITY**

6.2 ECONOMICAL FEASIBILITY:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

6.3 TECHNICAL FEASIBILITY:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

6.4 SOCIAL FEASIBILITY:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not

feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER 7

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

7.1 TYPES OF TESTS:

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business

Process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional Test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

7.2 Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output: identified classes of application outputs must be exercised.
- Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.3 System Test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

7.3.1 White Box Testing:

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

7.3.2 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

7.4 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives:

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested:

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page

7.5 Integration Testing:

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

7.6 Acceptance Testing:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 8

CONCLUSION

This open attitude determines the degree and scope of information sharing. Big data analysis technology can effectively improve the crop yield production is updating. This project proposes a novel intelligent system for agricultural crop price prediction. The key idea is to use ensemble of classifiers for prediction. The usage of fast tree regression trainer paves a path way to make a better decision on predictions due to the usage of multiple classifiers. This system is used to predict the cost of the crop rate for further.

The solution will benefit farmers to maximize productivity in agriculture, reduce soil degradation in cultivated fields, and reduce fertilizer use in crop production by recommending the right crop by considering various attributes.

This would provide a comprehensive prediction on the basis of geographical, environmental and economic aspects.

CHAPTER 9

SOURCE CODE

```
using Microsoft.ML;
using Microsoft.ML.Data;
using System;
using System.IO;
using System.Linq;
namespace AI_Regression_Crop_Recommendation
{
    // Data Class
    public class CropDetails
    {
        [LoadColumn(0)]
        public float temperature;
        [LoadColumn(1)]
        public float humidity;
        [LoadColumn(2)]
        public float ph;
        [LoadColumn(3)]
        public float label;
    }
    // CropPrediction is the result returned prediction by the model
    public class CropPrediction
    {
        [ColumnName("Score")]
        public float Label;
    }
    class Program
    {
        static readonly string _cropDataPath =
            Path.Combine(Environment.CurrentDirectory, "Data", "Crop_recommendation.csv"); static
        void Main(string[] args)
```

```

{
    MLContext mlContext = new MLContext(seed: 0);
    var model = Train(mlContext, _cropDataPath);
    Evaluate(mlContext, model);
    Console.ReadLine();
}

// Train() function
public static ITransformer Train(MLContext mlContext, string cropDataPath)
{
    Console.WriteLine("Training Started.....");
    // IDataView holds the training dataset
    IDataView dataView =
        mlContext.Data.LoadFromTextFile<CropDetails>(cropDataPath, hasHeader: true,
            separatorChar: ',');
    var pipeline = mlContext.Transforms.Conversion.ConvertType("Label",
        "label")
        .Append(mlContext.Transforms.Concatenate("Features",
            "temperature", "humidity", "ph"))
        .Append(mlContext.Regression.Trainers.FastTree());
    //create the model
    Console.WriteLine("Creating Model.....");
    var model = pipeline.Fit(dataView);
    Console.WriteLine("Model created.....");
    // return trained model
    return model;
}

// Evaluate the data
private static void Evaluate(MLContext mlContext, ITransformer model)
{
    Console.WriteLine("Evaluating Data.....");
    IDataView dataView =
        mlContext.Data.LoadFromTextFile<CropDetails>(_cropDataPath, hasHeader: true,
            separatorChar: ',');
    var prediction = model.Transform(dataView);

```

```

var metrics = mlContext.Regression.Evaluate(dataView, "label",
"label");
TestSinglePrediction(mlContext, model);
}
// Test Single Data
private static void TestSinglePrediction(MLContext mlContext,
ITransformer model)
{
var predictionFunction =
mlContext.Model.CreatePredictionEngine<CropDetails, CropPrediction>(model);
var CropSample = new CropDetails()
{
temperature = 18.87614998f,
humidity = 24.54038287f,
ph = 5.724242065f,
label = 0f
};
Console.WriteLine("Predicting data.....");
Console.WriteLine("");
var prediction = predictionFunction.Predict(CropSample);
Console.WriteLine("_____");
Console.WriteLine("Crop Details");
Console.WriteLine("_____");
Console.WriteLine("");
Console.WriteLine($" Temperature: {CropSample.temperature}");
Console.WriteLine($" Humidity: {CropSample.humidity}");
Console.WriteLine($" pH: {CropSample.ph}");
Console.WriteLine("");
Console.WriteLine("_____");
Console.WriteLine(" Predicted Data");
Console.WriteLine("_____");
Console.WriteLine(" ");
Console.WriteLine($" Predicted Value: {prediction.Label}");
Console.WriteLine(" ");

```

```

// crop recommendation
if (prediction.Label > 0 && prediction.Label < 1)
    Console.WriteLine(" Predicted Crop: Rice");
else if(prediction.Label > 1 && prediction.Label < 2)
    Console.WriteLine(" Predicted Crop: Maize");
else if (prediction.Label > 2 && prediction.Label < 3)
    Console.WriteLine(" Predicted Crop: Chickpea");
else if (prediction.Label > 3 && prediction.Label < 4)
    Console.WriteLine(" Predicted Crop: KidneyBeans");
else if (prediction.Label > 4 && prediction.Label < 5)
    Console.WriteLine(" Predicted Crop: Pigeonpeas");
else if (prediction.Label > 5 && prediction.Label < 6)
    Console.WriteLine(" Predicted Crop: Mothbeans");
else if (prediction.Label > 6 && prediction.Label < 7)
    Console.WriteLine(" Predicted Crop: Mungbean");
else if (prediction.Label > 7 && prediction.Label < 8)
    Console.WriteLine(" Predicted Crop: Blackgram");
else if (prediction.Label > 8 && prediction.Label < 9)
    Console.WriteLine(" Predicted Crop: Lentil");
else if (prediction.Label > 9 && prediction.Label < 10)
    Console.WriteLine(" Predicted Crop: Pomegranate");
else if (prediction.Label > 10 && prediction.Label < 11)
    Console.WriteLine(" Predicted Crop: Banana");
else if (prediction.Label > 11 && prediction.Label < 12)
    Console.WriteLine(" Predicted Crop: Mango");
else if (prediction.Label > 12 && prediction.Label < 13)
    Console.WriteLine(" Predicted Crop: Grapes");
else if (prediction.Label > 13 && prediction.Label < 14)
    Console.WriteLine(" Predicted Crop: Watermelon");
else if (prediction.Label > 14 && prediction.Label < 15)
    Console.WriteLine(" Predicted Crop: Muskmelon");
else if (prediction.Label > 15 && prediction.Label < 16)
    Console.WriteLine(" Predicted Crop: Apple");
else if (prediction.Label > 16 && prediction.Label < 17)

```

```
Console.WriteLine(" Predicted Crop: Orange");
else if (prediction.Label > 17 && prediction.Label < 18)
Console.WriteLine(" Predicted Crop: Papaya");
else if (prediction.Label > 18 && prediction.Label < 19)
Console.WriteLine(" Predicted Crop: Coconut");
else if (prediction.Label > 19 && prediction.Label < 20)
Console.WriteLine(" Predicted Crop: Cotton");
else if (prediction.Label > 20 && prediction.Label < 21)
Console.WriteLine(" Predicted Crop: Jute");
else
Console.WriteLine(" Predicted Crop: Coffee");
Console.WriteLine("_____");
}
}
```

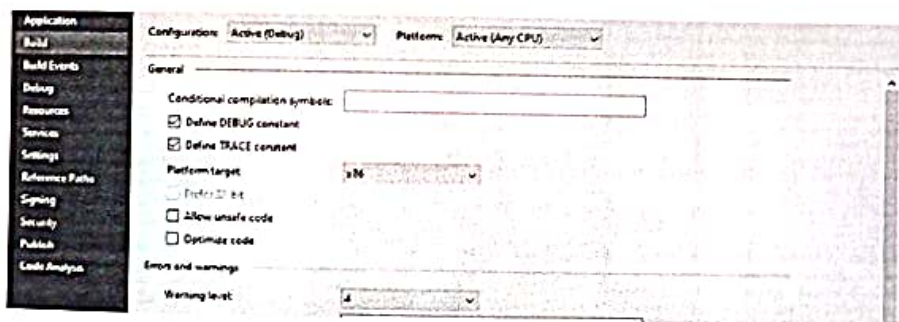
L

RESULTS AND DISCUSSIONS:

SCREENSHOTS

Run the program

In Solution Explorer, Right-click on the project name and select properties.



Click Build → platform target → x86 → Save (press 'Ctrl+S' on the keyboard). Click Start to run the program to see the predicted crop for your test case.

Step 5: Output

