

**College of Engineering, Environment, and Computing School of Science**

**MSc Data Science**

EECT 109

7150 CEM Data Science Project

## Project Report

**Time Series Forecasting of Yield Based on Fertilizer and Nutrient Inputs**

By

**Tharun Chilikeshwaram**

**SID: 15537232**

1st Supervisor: **Dr. Furrkh Aslam**

2nd Supervisor: **Dr. Haipeng Liu**

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Data Science

Academic Year: 2024/25

**Declaration of Originality**

I declare that this project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

**Statement of copyright**

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialize products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see [www.coventry.ac.uk/ipr](http://www.coventry.ac.uk/ipr) or contact [ipr@coventry.ac.uk](mailto:ipr@coventry.ac.uk).

**Statement of ethical engagement**

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking).

**Signed:** Tharun Chilikeshwaram Date:15thAugust 2025

Please complete all fields.

| First Name: | **Tharun** |
| --- | --- |
| Last Name: | **Chilikeshwaram** |
| Student ID number | **15537232** |
| Ethics Application Number | **P187825** |
| 1st Supervisor Name | **Dr.** **Furrkh Aslam** |
| 2nd Supervisor Name | **Dr. Haipeng Liu** |

**Certificate of Ethical Approval****

***Abstract***

This dissertation performs detailed investigation of the application of multivariate time series forecasting techniques for maize yield prediction based on fertilizer and nutrient input data. The dataset used was sourced from the Rothamsted Broadbalk experiment. For predictive modeling three models were created which included SARIMA, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). These models were compared under identical data settings. The dataset was preprocessed extensively which included handling missing values, encoding categorical variables, and generating time-dependent sequences.

For measuring the model performance, the evaluation metrics like Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error(MAPE), and the Akaike Information Criterion (AIC) were used. The results showed that both deep learning models consistently outperformed SARIMA model, with GRU achieving the lowest RMSE (2.03) and MAPE (~20%), demonstrating superior accuracy. The regression analysis and feature importance analysis revealed that nitrogen application timing, farmyard manure use, and dry matter percentage were significant yield determinants, while excessive phosphorus application could reduce productivity.

The study concludes that GRU provides an optimal balance between the predictive performance and the resource efficiency for multivariate agricultural forecasting. The methodology used in this dissertation follows a well-established structured approach and is thus, transferable to other corps and geographies, providing potential for integration into decision support systems for optimizing the fertilizer management, improving the yield of crops and also supporting sustainable agriculture.

**Keywords:** Crop Yield Forecasting, Multivariate Time Series, Fertilizer Management, LSTM, GRU

# 

[**1. INTRODUCTION 1**](#_jis4yibriyfx)

[1.1. Background 1](#_ra8rtzdvvp8)

[1.2. Statement of Problem 2](#_vuadxi3uq8q3)

[1.3. Research Question 3](#_dq74of4orc57)

[1.4 Aims and Objectives 3](#_hdlpjnw9494)

[1.5. Project Scope 3](#_eskhxgrt1fwq)

[1.6. Structure of the Report 4](#_5vpojfnpog2k)

[**2. LITERATURE REVIEW 5**](#_82riswwbn59o)

[2.1. Introduction to Crop Yield Forecasting 5](#_7n2xe06nhta7)

[2.2. Time Series Forecasting in Agriculture 5](#_xnjitju9u5bm)

[2.3. Statistical Model for Time Series Forecasting 6](#_ogszzgecp10w)

[2.4. Machine Learning Approaches in Crop Yield Prediction 7](#_pghwdha7xdy9)

[2.5. Deep Learning Models for Time Series Forecasting 8](#_22e5ei7d9jo2)

[2.6. Multivariate Time Series Forecasting Techniques 8](#_anrrtij6w8xn)

[2.7. Evaluation Metrics for Crop Yield Forecasting Models 9](#_l3p94jvxl1la)

[2.8. Gaps and Challenges in Current Research 10](#_m6shvp2v1fn8)

[**Chapter 3: Research Methods 11**](#_a9rdwl98wmzx)

[3.1. Introduction 11](#_bx2bm26sxz4a)

[3.2. Summary of Research: Saunders Research Onion 11](#_n4o9aj21u93p)

[3.2.1. Research Philosophy - Positivism 11](#_5jqz8h7vv37g)

[3.2.2. Research Approach 11](#_7fos9q5sjdfs)

[3.2.3. Research Strategy 11](#_i035lxyjizc0)

[3.2.4. Methodological Choice 12](#_u53q3cp8yrt0)

[3.2.5. Time Horizon 12](#_z2btdsb9eooo)

[3.2.6: Techniques and Procedures 12](#_84dpqjwhmwen)

[3.3. Research Trajectory Followed with Justification 13](#_4nmugvu7h1qx)

[3.3.1. Data Collection and Description 13](#_iqx83oio9m9o)

[3.3.2. Data preprocessing and Preparation 14](#_txv457vbc7ns)

[3.3.3. Exploratory Data Analysis (EDA) and Visualizations 14](#_e9tydo6vdb9)

[3.3.4. Model Development 18](#_qwa74qsv4xsx)

[3.3.4. Model Evaluation 19](#_oi5j62rco6ub)

[**CHAPTER 4: RESULTS 20**](#_uj69n51t1t2f)

[4.1. Introduction 20](#_5pz2mkcdde41)

[4.2. SARIMA Model Output Analysis 20](#_695p54mjb0vz)

[4.3. LSTM Model Performance 21](#_4xikxogg5uft)

[4.4. GRU Model Performance 21](#_oey0b3fluukh)

[4.5. Model Performance Comparison 22](#_vfccdkuk7q5i)

[**CHAPTER 5: CRITICAL DISCUSSION OF RESULTS AND RECOMMENDATIONS 23**](#_p65gdg79rc5i)

[5.1. Introduction 23](#_4yzexp2asf73)

[5.2. Discussion 23](#_zeue7sg0kpxy)

[5.3. Recommendations 24](#_qmkuilt5gzp4)

[**CHAPTER 6: CONCLUSION, FUTURE ENHANCEMENTS, AND CRITICAL EVALAUTION 26**](#_34hbnzv2dhq9)

[6.1. Introduction 26](#_nws2rrilw680)

[6.2. Mapping of Objectives to Completion 26](#_thnfh7a2m7ao)

[6.3. Conclusion of Work Carried Out 27](#_yqrw9z7r8o1a)

[6.4.Future Enhancements 28](#_h5o96yk1kf2i)

[6.5. Critical Evaluation 29](#_sll60jhmera4)

[6.6 Final Remarks 30](#_i4n63pogudp1)

[**CHAPTER 7: SELF-REFLECTION 31**](#_9dhbnvlomp8)

[7.1. Introduction 31](#_pblrv7kk5xzl)

[7.2. Description and Feelings 31](#_jkd6375ai0p5)

[7.3. Skills Developed 31](#_i3vs6eselatv)

[7.4. Challenges and Response 32](#_kkr1xyovv2wx)

[7.5. Lessons Learned from Dissertation 32](#_96od36c07cff)

[7.6 Future Application 33](#_n37m6fxuo1xb)

[7.7 Conclusion 33](#_e4259m6151hw)

[**CHAPTER 8: PROJECT MANAGEMENT 34**](#_m7wkflri5vp0)

[8.1. Introduction 34](#_y20t5s4642ep)

[8.2. Work Breakdown Structure (WBS) 34](#_qqrfbzlcguqg)

[8.3. Task Dependencies 35](#_aj3dm0xhklql)

[8.4. Baseline Gantt Chart 36](#_cxqjwyjx962b)

[8.5. Milestones 36](#_2axstcj06srf)

[8.6. Risk Management 37](#_svklzrcwgb1r)

[8.7 Monitoring and Control 37](#_jtfw3lbhcdw)

[8.8 Lessons Learned in Project Management 38](#_n8o2l1jqaavo)

[8.9. Conclusion 38](#_nx4vbu2vuok2)

[**REFERENCE 39**](#_6ibaxqql29jx)

[**APPENDIX I: DATA LINK 44**](#_ktqq1b68hofw)

[**APPENDIX II: CODE SNIPPETS 45**](#_ag2xatfipckc)

# 

# INTRODUCTION

## 1.1. Background

Agricultural productivity is one of the important parts of global food security, economic stability and sustainable development (Arora, 2018). In most of the developing countries with emerging economies, it is not only a means of odd production but also the main source of their livelihood. Crop yield is one of the most important measures of agricultural performance as it shows how productive and efficient farming methods are. There are a wide range of the factors which influence the crop yield. These factors include climate, type of soil, irrigation availability, pest control measures and most importantly, type and quantity of the fertilizer used (Balasubramanian et al., 2024; Liliane & Charles, 2020). Fertilizers used in the field provide micronutrients like phosphorus (P), potassium (K), nitrogen (N) etc which are essential for the growth of the crop and improving their quality (ALnaass et al., 2021). In addition to this, fertilizers also are the source of micronutrients like zinc, boron, iron etc which are very important for the healthy growth of the crops (Shuman, 2017). These micronutrients and macronutrients help to maintain the quality of the crops. The right balance of these nutrients helps to make sure the optimal crop growth, while its imbalance might cause reduced yields, causes damage to the environment, and also increases the production costs.

In recent times, the use of data in agriculture has grown substantially. Large datasets from the agricultural surveys performed by the government, data collected by the weather monitoring systems, solid analyses, and field trails are available to the farmers and academics (Coble et al., 2018; Ramirez-Vilegas & Challinor, 2012). The main challenge today is not the absence of data, but converting these data into actionable insights. Among the different types of analytical techniques available today, time series forecasting has come out as one of the most powerful techniques for performing the forecasts of future yields in agriculture (Reddy & Sureshbabu, 2019). Time series forecasting can capture both the temporal patterns and also the long-term trends which are present in the agricultural systems.

There are numerous forecasting tools from statistical to deep learning algorithms which can make effective predictions. The popular statistical tools contain algorithms like ARIMA, SARIMA etc while the state-of-art forecasting technique includes neural network based architectures like Long-Short Term Memory (LSTM), Gated Recurrent Unit (GRU) network (Kurumatani, 2020). These tools allow for the integration of historical yield records with input variables like fertilizer used, nutrient type, quantity of fertilizer, application timing etc. The ability to accurately forecast future yields and modify agricultural techniques in response is provided by this integration. Planning resources, managing market supply, and implementing sustainable agricultural methods all depend on these estimations.

## 1.2. Statement of Problem

Despite the critical role that fertilizers and nutrients play in crop output, their proper application remains very challenging in modern agriculture. Farmers mostly depend on empirical information, traditional standards, or generalized suggestions which fail to account for site-specific variables, seasonal variations, and climate-related changes. The overapplication of the fertilizers can cause degradation of soil, cause water pollution from nutrient runoff, and also increase the financial expense (Ramamoorthy et al., 2024). On the other hand, underapplication of the fertilizers can result in shortages of nutritional content in the soil and thus affect the growth of the crop, resulting in lower yield.

Most of the yield prediction methods usually fail to take into account for the time-dependent relationships between application of fertilizers and yield results Most of the current techniques regard agricultural output as a static process and employs cross-sectional or regression models which do not take into considerations for the historical patterns, seasonally cycles, or the delayed impacts of fertilizer application (Mendelsohn & Massetti, 2017; Lobell & Burke, 2010). As a result, decision-makers may not have access to accurate, dynamic projections which can adapt to changing circumstances.

This gap in the existing research highlights the need for a forecasting approach which can:

1. Incorporate the historical patterns of fertilizer and nutrient application
2. Adjust forecasts to reflect changing environmental and agricultural circumstances.
3. Provide accurate yield forecasts to help make prompt and precise fertilizer management decisions.

It is necessary to address this problem because it helps to improve farm-level productivity and also helps to ensure food security, reduce environmental impact, and optimize the agricultural resource allocation on a larger scale.

## 1.3. Research Question

The research question which drives this dissertation is:

*“How can time series forecasting be effectively applied to forecast crop yield based on fertilizer and nutrient inputs, in order to optimize agricultural productivity?”*

## 1.4 Aims and Objectives

Aim: The aim of the project is to develop and perform a comprehensive analysis of a time series forecasting model using statistical techniques such as ARIMA and deep learning techniques such as LSTM and GRU, based on fertilizer and nutrient input data, with a focus on improving decision-making in agricultural management.

Objectives: The objectives are listed below:

1. To perform comprehensive review and synthesis of existing literature on yield prediction, fertilizer optimization and time series forecasting methods in agriculture.
2. To collect and preprocess historical datasets on crop yield, fertilizer usage, and nutrient composition from credible data sources.
3. To implement ARIMA, LSTM and GRU models and train the models on the preprocessed dataset and validate the performance of each model using evaluation metrics like Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and MAPE (Mean Absolute Percentage Error) etc.
4. To perform analysis of the relationship between various fertilizer/nutrient application levels and predicted yields.
5. Based on the findings of the evaluation, formulate some recommendations for optimized fertilizer and nutrient management strategies.

## 1.5. Project Scope

This study focuses on forecasting maize (Zea mays L.) yield using time series models that integrate fertilizer and nutrient input data.

1. Geographic Scope: The crop data originates from the Broadbalk experimental fields in Harpenden, UK. While the research is site-specific, the modeling technique is intended to be applicable to similar maize-growing locations.
2. Crop Scope: The study is limited to forage maize grown in approved Broadbalk portions as part of the five-year cycle of wheat, wheat, oats, and maize.
3. Data Scope: Historical yield data containing information of the fertilizer and micronutrient and macronutrient information.
4. Technical Scope: Includes comparative evaluation of time-series forecasting methods like ARIMA, LSTM and GRU to find the model that is most accurate.
5. Results Scope: Delivery of the validated maize yield forecasting models and recommendations for fertilizer management.

## 1.6. Structure of the Report

This first chapter is the introduction chapter which provides the background and scope of the project. After the introduction chapter is the literature review chapter in which the review of previous relevant literature has been performed and gaps in existing research have been identified. The third chapter is the methodology chapter which outlines all the steps followed to achieve the aim of the dissertation. The methodology for the experiment has been provided in detail ensuring its reproducibility. After methodology the results of the evaluation of different time series forecasting models have been presented. Following the results chapter is the project management chapter which discusses the strategies and standards followed and project management tools used to manage this project. The sixth chapter provides critical discussion of the results which is then followed by the conclusion and future enhancements for the project in the seventh chapter. The final chapter is the student self reflection chapter in which the knowledge and experience acquired from this project has been reflected on.

# 2. LITERATURE REVIEW

## 2.1. Introduction to Crop Yield Forecasting

It is important to forecast the crop yield accurately for ensuring food security and economic stability (Fischer et al., 2014) It allows farmers and governments to prepare plans effectively by anticipating the future yield levels. However, projecting agricultural yields is difficult because of the effect of several dynamic elements like fluctuations in climate, solid quality, irrigation, pest management and fertilizer use (Aggarwal et al., 2006). These elements interact nonlinearly which makes the accurate prediction challenging.

Traditional forecasting methods which include statistical models like ARIMA and SARIMA have been widely used for forecasting the crop yields. These models can capture the seasonal trends and temporal patterns (Mwanga et al., 2017). However, these techniques often struggle when it comes to handling the nonlinear and multivariate nature of agricultural data (Thota, 2025). In recent years, the field of machine learning and deep learning has advanced significantly. The deep learning algorithms, especially recurrent neural network variants like LSTM and GRU have been proven to show good ability in capturing the complex dependencies and improving the forecast accuracy (Shiri et al., 2023). Integrating these advanced models with the multivariate data like fertilizer inputs and weather variables can provide a very robust approach to enhance crop yield predictions and support sustainable agricultural decision-making (Jabed & Murad, 2024).

## 2.2. Time Series Forecasting in Agriculture

Time series data consists of the sequential observations which are collected over a certain period of time (Chatfield, 2013). In agricultural settings, these patterns represent underlying phenomena such as seasonal crop cycles, long-term yield shifts, and sudden fluctuations which are driven by environmental factors (Rogachev & Simonov, 2022; Jeba et al., 2024) .

Time series forecasting can be classified into univariate and multivariate techniques (Liu et al., 2021). In univariate forecasting, there is use of the single historical variable for making the forecasting of future values. It is a simple approach and might overlook important external factors which may have a significant influence (Petropoulos & Spiliotis, 2021). In comparison to this, multivariate forecasting techniques incorporate more than one variable in making forecasts. This inclusion of multiple variables can capture complex interactions and dependencies among these factors (Mendis et al., 2024).

Since, the crop production is highly dependent on a number of interrelated factors, including weather (temperate, rainfall), solid nutrient levels, irrigation, and fertilizer application, multivariate techniques are particularly relevant in forecasting the crop yield (Bharadiya et al., 2023). Multivariate models may better comprehend and estimate production changes by including these factors in the forecast. This makes the predictor model more reliable and accurate and promotes better agricultural management and planning.

## 2.3. Statistical Model for Time Series Forecasting

Various traditional statistical models have been used for time series forecasting in agriculture. The popular algorithms are Autoregressive Integrated Moving Average (ARIMA) and its seasonal extensions, Seasonal ARIMA (SARIMA). ARIMA models work by combining the autoregression, differencing and moving averages. These components of ARIMA help the model to capture the temporal dependencies and trends. The SARIMA model extends on the working mechanism of the ARIMA. This model extends the ARIMA model to account for seasonal patterns commonly present in the agricultural data (Majka, 2024).

SARIMA is suitable for modeling seasonal agricultural datasets in which yield patterns repeat annually due to planting and harvesting cycles. This model is helpful for predicting agricultural yields under cyclical climatic conditions because it can clearly represent seasonality, trend and residual noise present in the historical yield data (Zevallos-Aquije et al., 2025; Kmytiuk et al., 2024). SARIMA works on the assumption of linear correlations and stationarity (Ayo et al., 2024). This may limit its capacity to capture complex nonlinear interactions across a wide range of variables like weather, and soil nutrients. The applicability of the SARIMA model might also be limited in multivariate agricultural applications because it was initially designed for univariate forecasting (Othman et al., 2024; Nokeri, 2021).

In the domain of forecasting crop yield, the study, Parreño, (2023) used SARIMA successfully in predicting the yields of crops like rice, corn and other staples by making use of the historical yield series and seasonal climate variables. There are also studies like Selvakumar & Kasthuri, (2022) which have used ARIMA/SARIMA variants as benchmarks for Wheat production prediction tasks. While these statistical models are good for stable seasonal trends, there are several recent studies which have highlighted the need for more advanced deep learning methods which are good at modeling the non-linear, multivariate and long-term dependencies better than the statistical models (Mahmoud & Mohammed, 2024; Moazzen & Hossain, 2024).

## 2.4. Machine Learning Approaches in Crop Yield Prediction

Machine learning (ML) techniques are increasingly being adopted in agriculture to improve crop output prediction. The techniques are well known for their ability to capture complicated, non-linear interactions between various environmental factors. Some of the most popular ML algorithms includes Random Forests (RF), which make use of the ensemble decision trees for reducing the variance and improving the robustness; Support Vector Machine (SVM) which show good performance in high-dimensional feature spaces; and Gradient Boosting Machines (GBM), such as XGBoost and LightGBM, which performs multiple interactions to improve accuracy of prediction via sequential learning (Halabaku & Bytyci, 2024; Kazemi, 2024; Airlangga, 2024) .

The ML techniques provide various strengths for the prediction of crop yield. It includes capacity to handle diverse datasets, incorporate interactions which are of non-linear nature and leverage multivariate input variables like solid qualities, fertilizer application rates, and remote sensing indices (Jabed & Murad, 2024). However, traditional ML models usually do not account for temporal dependencies which are present in the time series dataset unless specifically designed through techniques like lag features, rolling windows, or other preprocessing techniques (Masini et al., 2023. Amor et al., 2016; Bontempi et al., 2012)

There are various studies which have demonstrated the usefulness of multivariate ML models in agriculture. For example, the study Jeong et al. (2016) made use of the Random Forests with climate, soil, and management factors for estimating maize yields in the US over a thirty-year period. The study found that a Random Forests model accurately predicted maize yields with a strong correlation (R=0.89) and a low error (RMSE = 1.1 Mg ha⁻¹), outperforming the traditional methods. This implies that crop yield estimation can benefit greatly from the use of a machine learning framework that incorporates both climate and management aspects (Jeong et al., 2016). Another study, Shahhosseini et al. (2021), made use of the hybrid framework which combined APSIM crop model with Extreme Gradient Boosting (XGBoost) and Light Gradient Boosting Machine (LightGBM) with the weather and soil data to forecast the corn yields. This hybrid ML framework was able to improve the performance, reducing the RMSE of corn yield prediction by 7% to 20% as compared to using standalone ML models. These results show the potential of ML techniques in multivariate yield prediction, while sophisticated temporal models which include different architectures of neural networks like LSTM and GRU may outperform them in capturing the sequential relationships (Shahhosseini et al., 2021).

## 2.5. Deep Learning Models for Time Series Forecasting

Recurrent Neural Networks (RNNs) and their enhanced variants such as LSTM and GRU, have been created to capture sequential dependencies that are present in the sequential data like time series. In contrast to the traditional RNN model, LSTM and GRU are effective in capturing the long-term dependencies and patterns in the sequential data. The LSTM and GRU are able to achieve this by overcoming the vanishing gradient problem which is very common in the vanilla RNN models.

LSTM makes use of the memory cells and gating mechanisms. Through these mechanisms this model is able to retain the relevant information over long sequences. Apart from this, the GRU models are the simplified architecture which has fewer number of gates due to which, GRU models train faster than the LSTM models but at the same time offers comparable accuracy. Due to the features of LSTM and GRU, both models are good for agricultural forecasting, where crop yield depends on numerous complex, non-linear temporal factors such as weather trends, fertilizer application timing, soil nutrients etc.

There are various studies which have used LSTM and GRU models for predicting the crop yields and showed improved performance over traditional statistical methods like ARIMA. For example, a study Qiu, (2023), used LSTM and GRU for predicting the rice yield. As per the results, the LSTM and GRU showed strong performance. In comparison to the original state, the mean squared error (MSE) loss was decreased by 96–98%. The final MSE values varied between 0.2358 and 0.4256 for each model configuration. One important finding of this study is that simpler models with fewer layers converged more quickly, indicating that they are a more effective option for issues with small sample sets, even if model depth and direction had no significant impact on the final prediction performance (Qui, 2023). Another Study, ArunKumar et al., (2022) made a comparative assessment of the LSTM, GRU and ARIMA model for time series forecasting of COVID-19 trends. The study found out that the LSTM and GRU models consistently outperform ARIMA with RMSE values up to 40 times lower for the deep learning models for most cases (ArunKumar et al., 2022).

These studies highlight the capability of deep learning models in modeling complex dependencies and long-term patterns and are thus suitable for the crop yield forecasting task.

## 2.6. Multivariate Time Series Forecasting Techniques

Multivariate time series forecasting includes multiple interdependent variables. This allows models to capture both temporal dependencies and cross-variable correlations (Mendis et al., 2024). This is very crucial in agriculture as soil, fertilizer, climate etc. all affect crop yields (Liliane & Charles, 2020). Although this method increases the accuracy of predictions over univariate models, it too has certain limitations like high complexity, missing values, and issues with data alignment (Koch, 2013).

Modeling long-term patterns and non-linear relationships is a good fit for deep learning architectures like multivariate LSTM and GRU. For example, Shook et al., (2021), presents a study on soybean yield prediction using LSTM-RNN models on the multivariate data which included genotype, weather variable and pedigree data. One of the main contributions of this paper is the development of temporal attention mechanisms for the LSTM models, which allow for better interpretability via the identification of the specific environmental conditions which influence the yield prediction (Shook et al., 2021).

In agriculture and other fields, hybrid systems that combine deep learning with statistical models (like SARIMA) have also proven effective. In agriculture and other fields, hybrid systems that combine deep learning with statistical models (like SARIMA) have also proven effective (Khashei & Bijari, 2011). These hybrid models offer a balance between interpretability and modeling capacity (Khashei & Bijari, 2011).

## 2.7. Evaluation Metrics for Crop Yield Forecasting Models

In machine learning or statistical tasks, choosing the right metric for evaluation of the model is very important as it provides the idea of how well the model performs. Combining multiple metrics can offer a comprehensive evaluation, especially in forecasting tasks where accuracy and interpretability are important and directly impacts the decision making process by the management team. Some of the evaluation metrics that are suitable for the crop yield forecasting task are provided below:

**Root Mean Square Error (RMSE):** RMSE is a metric which gives the measure of the square root of the mean error measured between the predicted and observed values. The objective of taking the square root is that, this will penalize larger value errors more. This makes the metric sensitive to outliers (Hodson, 2022). RMSE is largely used in tasks related to regression and time series forecasting and thus, is suitable for the evaluation of models in crop yield forecasting (Jadon et al., 2024).

**Mean Absolute Error (MAE):** This metrics measures the mean of the absolute difference between the predicted value and the actual or real values and provides a very simple interpretation of average prediction errors. It doesn’t put emphasis on larger errors unlike RMSE and provides a balanced perspective on model performance (Hodson, 2022).

**Mean Absolute Percentage Error (MAPE):** MAPE is another important metric which expresses errors as a percentage of observed values. It allows for relative assessment of error. It is helpful in understanding prediction accuracy in terms of proportional deviations. However, it can be distorted when observed values are near zero (Chicco et al., 2021).

**Coefficient of Determination (R²):** This metric is used for describing how much variance is explained by the model. This metric is the indication of goodness of fit. The value of 1 represents better fit. However, this metric does not directly measure the magnitude of the prediction error (Chicco et al., 2021).

## 2.8. Gaps and Challenges in Current Research

There are various advances made in the prediction of crop yield using machine learning and deep learning techniques. However, most of these studies rely on the univariate models or limited variables. Due to this, there is a lack of comprehensive multivariate approaches which can reflect the complex environmental and management interactions. Other issues involve data scarcity, inconsistency, and quality issues. This affects the generalizability of the models. Thus, this dissertation study aims to bridge this gap by making use of the multivariate data for crop yield forecasting. The study also addresses the issues of data inconsistency and quality issues through extensive preprocessing and data validation techniques. This study integrates LSTM, GRU and SARIMA techniques to predict yields based on fertilizer and nutrient data. Through the use of the capacity of deep learning to model non-linearity and long-term dependencies alongside SARIMA’s strength in modeling seasonality, this dissertation aims to enhance the accuracy of forecasting yield and support data driven decision making in agricultural management.

# Chapter 3: Research Methods

## 3.1. Introduction

The details of the methodology adopted for investigating multivariate time series forecasting in the context of the maize yield prediction using three distinct modeling approaches like SARIMA, LSTM networks, and GRU networks has been presented in this chapter. A thorough methodological framework is important for ensuring that the study is reproducible, valid and credible. The research methodology makes use of the Saunders et al’s (2019) Research Onion as the guiding structure. This helps to ensure the philosophical assumptions, strategy, approach, collection of data and analysis are logically aligned with the aim and objectives of the research (Seuring et al., 2021).

## 3.2. Summary of Research: Saunders Research Onion

The Research Onion presents a methodical decision-making framework. Each layer is examined as follows for this study:

### 3.2.1. Research Philosophy - Positivism

The study is based on the positivist ideology which holds that objective reality can be measured and quantified by means of statistical analysis and methodological data collecting (Ali, 2024). Crop yield forecasting has been viewed as an objective challenge which involves identifying quantifiable trends in the agricultural datasets.

### 3.2.2. Research Approach

The research follows a deductive approach. The study builds upon established forecasting theories and performs testing of the different statistical and deep learning models in the agricultural domain. The main hypothesis here is that deep learning models like LSTM and GRU will easily outperform statistical models like SARIMA in capturing the trends and non-linear dependencies and multivariate relationships in the crop yield data.

### 3.2.3. Research Strategy

For this study, empirical research has been chosen which contains the experimental strategy. Experiments for the study have been done in a controlled environment which included implementation and evaluation of multiple forecasting algorithms under the same dataset and preprocessing conditions. The approach made it possible to ensure that variations in model performance could be ascribed to the models themselves rather than any discrepancies in the data preparation.

### 3.2.4. Methodological Choice

The research is of quantitative nature which includes building of the various deep learning and statistical models using the numerical agricultural dataset on maize yield. For the evaluation of the models, different performance metrics such as RMSE, MAE, MAPE, and R-squared have been used.

### 3.2.5. Time Horizon

Since the information covers several years of data, a longitudinal approach is used to capture long-term agricultural patterns and seasonal cycles. For models like SARIMA, which rely significantly on seasonal trends, this is essential.

### 3.2.6: Techniques and Procedures

The experiment conducted in this study follows the following structured approach of the typical machine learning project:

1. Data collection from the open database.
2. Data preprocessing which includes handling missing data, normalization and feature engineering for multivariate inputs.
3. Exploratory data analysis which consisted of building different visualization plots to gain insights from the data.
4. Model development which includes implementing models like SARIMA, LSTM and GRU architectures.
5. Evaluation of the developed models using multiple statistical metrics for robust performance comparison.

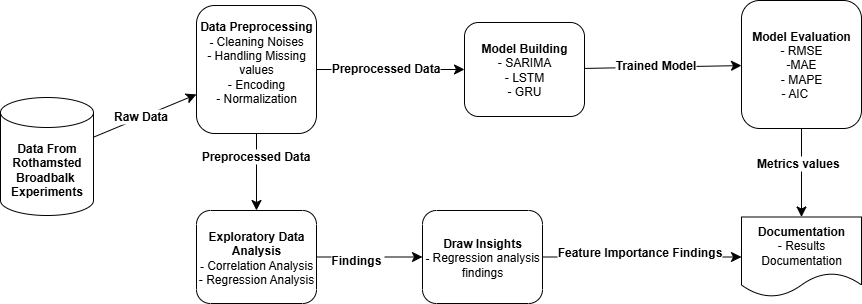


Figure 1: Diagram showing the stages in the pipeline for empirical research

## 3.3. Research Trajectory Followed with Justification

### 3.3.1. Data Collection and Description

The dataset for this research was acquired from the Electronic Rothamsted Archive (ERA), specifically the Broadbalk Winter Wheat Experiment-Grain Yield dataset (Rothamsted Research, 2024). This dataset represents one of the world's longest-running agricultural studies, including high-quality, long-term data on winter wheat yields. Since the mid-19th century, yearly grain production data have been gathered under a variety of fertiliser and management treatments (Rothamsted Research, 2024). The dataset used for this experiment was the maize yield data which provides detailed annual records of maize yields alongside comprehensive agronomic management variables from year 1997 onwards.

Each dataset record contains information on harvest year, fertilizer used, nutrient application levels (N, P, K, Na, Mg), application dates, previous crop, cultivar, sowing and harvest dates, and dry matter percentage. The yield in the data has been reported as 100% dry matter (tonnes per hectare). This ensures consistency across years. Fertilizer treatments are coded (e.g., “FYM N4 PK”) to represent combinations of farmyard manure, nitrogen levels, and other nutrient applications.

This dataset is suitable for multivariate time series forecasting as it includes the temporal yield data with multiple influencing factors like soil nutrient levels, crop rotation history, and management practices. Its high degree of information and regular yearly measurements make it appropriate for use with both statistical models (SARIMA) and deep learning architectures (LSTM, GRU).

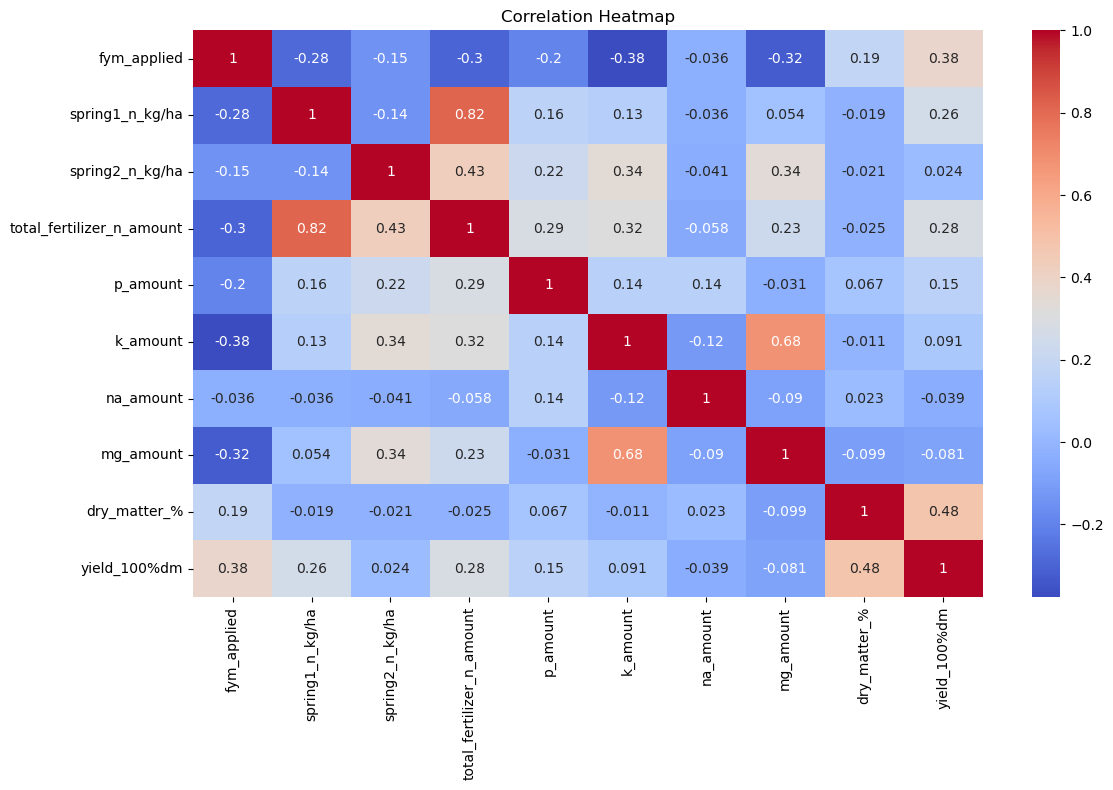
### 3.3.2. Data preprocessing and Preparation

The raw maize yield dataset collected had a lot of issues like missing values, inconsistencies and also contained noise. Thus, to make it suitable for the model training process, the data was preprocessed using following techniques:

1. **Data cleaning and handling missing values:** Initial inspection of the dataset identified non-essential metadata fields and inconsistent entries. These variables were removed from the dataset. There were also numerous missing values in numerical fields which were handled partially using mean imputation where columns were of numerical and continuous nature. For imputing the missing values in categorical columns, mode imputation was used.
2. **Feature Engineering:** Only relevant variables were identified and retained. The relevant variables included Yield (100% dry matter, t/ha) as the target variable, Fertiliser nutrient amounts (N, P, K, Na, Mg), application dates,FYM (Farmyard Manure) levels, sowing and harvest dates and fertilizer code as the independent variables.
3. **Data Encoding:** There were multiple variables in the dataset which were of categorical nature like *fertilizer\_code* and *cultivar*. Thus, these variables were encoded into vectors using one-hot encoding techniques (Poslavskaya & Korolev, 2023).
4. **Normalization:** The features were of different ranges. So, to make them consistent, the normalization of the data was carried out to make them consistent so that each of the variables contribute equally to the prediction. The min-max scaling technique was used for normalizing the data (Henderi et al., 2021).
5. **Sequence Generation:** Sliding window sequences with a specified look-back duration of 3 years were developed for LSTM and GRU models. Every sequence included the relevant yield objective for the anticipated year along with other input characteristics from previous years.
6. **Train-Test Split:** Data was split into training and testing set in the ratio of 8:2. The test set consisted of the most recent years to evaluate real-world forecasting performance.

### 3.3.3. Exploratory Data Analysis (EDA) and Visualizations

To get better understanding and extract valuable insights from the dataset, extensive EDA and visualizations were done. To understand the relationship between various variables, the correlation heatmap was generated which can be seen below (see Figure 1):

Figure 2: Correlation heatmap showing the strength of linear relationship between various variable pairs.

The correlation heatmap shows strong correlations between spring1\_n\_kg/ha and total\_fertilizer\_n\_amount (r = 0.82) and between k\_amount and mg\_amount (r = 0.68), suggesting redundancy. Moderate yield correlations were found for fym\_applied, total\_fertilizer\_n\_amount, spring1\_n\_kg/ha, and dry\_matter\_%, while spring2\_n\_kg/ha, na\_amount, and mg\_amount showed little to none.

To see variation of yield across different fertilizer code a box plot was plotted. The plot can be seen below (see Figure 2):

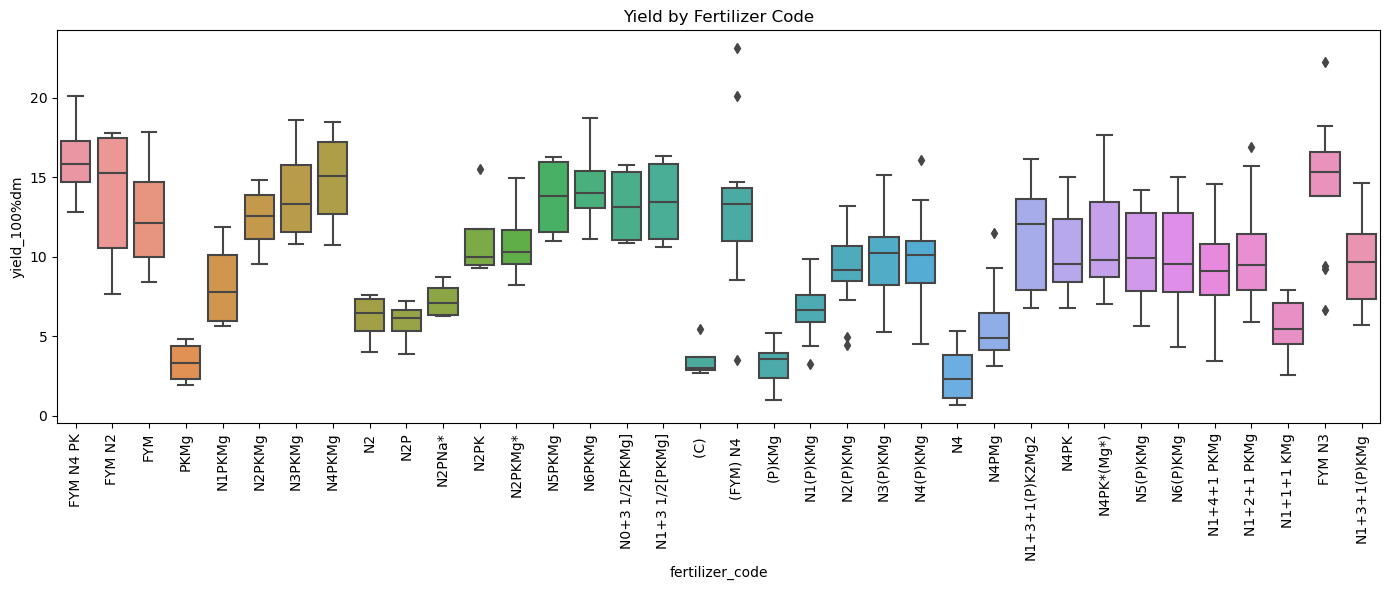


Figure 3: Box plot for different columns of the dataset shows presence of outliers in several columns

The box plot shows variation across yield by fertilizer code. Some combinations like FYM N4 PK and N1+3+1(P)KMg have higher median yields with larger interquartile ranges, suggesting higher effectiveness but also larger variability. Treatments like (P)KMg and N4(P)KMg have lower median yield and lower range, suggesting little impact on productivity. Outliers are observed in certain groups, such as those containing FYM and multi-nutrient combinations, suggestive of potential response heterogeneity. In general, the plot suggests that balanced and organic fertilizer treatments tend to be associated with increased yields.

Further to see the trend of maize yield over the years, a time series plot was plotted. The plot can be seen below (see Figure 3):

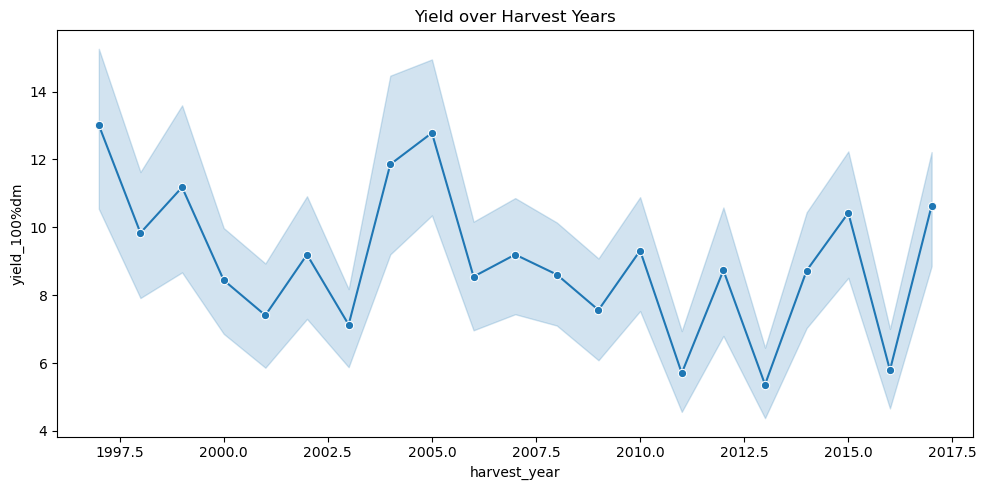


Figure 4: Time series plot of the yield over the years

From the plot it can be seen that the trend of maize yield decreased from 1997 to 2002 significantly. Then after the year 2002 till 2005 there was a significant rise in the yield. However after 2005, the crop yield is in decreasing trend again.

In addition to these visualizations, regression analysis was also done as a part of EDA to understand which factor or variable contributes most to the yield. The multivariate regression analysis revealed various interesting insights about the data. The analysis performed proper analysis of how factors like fertilizer type, application timing and crop characteristics affect yield, focusing on statistically significant results to ensure findings reflect genuine effects rather than chance.. The regression analysis also showed a slight decline in yield over time, while the application of farmyard manure (FYM) moderately increased yield. In addition to this, nitrogen applied in spring produces significant boost, and higher total nitrogen also increases yield. In contrast to this, increased phosphorus content in the soil was associated with a small reduction in yield. Higher dry matter content and longer days to harvest were both linked to higher yields.

The correct combinations of the Fertilizer had significant effects. For example, combinations such as N1+1+1 KMg, N1+2+1 PKMg, and nitrogen timing in mid-spring led to very large yield gains.However, some treatments, including N4, N6(P)KMg, N5(P)KMg, and certain FYM blends like FYM N3 or FYM N4 PK, resulted in sharp yield declines. PKMg, N2PKMg, and N1(P)KMg were among the combinations that showed moderate yield gains. Cultivar Severus continuously performed better than the baseline variety in terms of crop variety.

In practice, these results show that applying nitrogen at the right time, especially in mid spring, and choosing effective fertilizer combinations can greatly improve yields. In contrast, using too much of certain nutrients like phosphorus or unsuitable fertilizer mixes can lower productivity. Selecting the right crop variety and managing nutrients carefully is very important for producing higher crop yield.

Furthermore, to rank the various variables based on their contribution to crop yield, the feature importance analysis was performed using Random Forest. The top 10 variables which have significant impact on the yield can be seen in the plot below (see Figure 4):

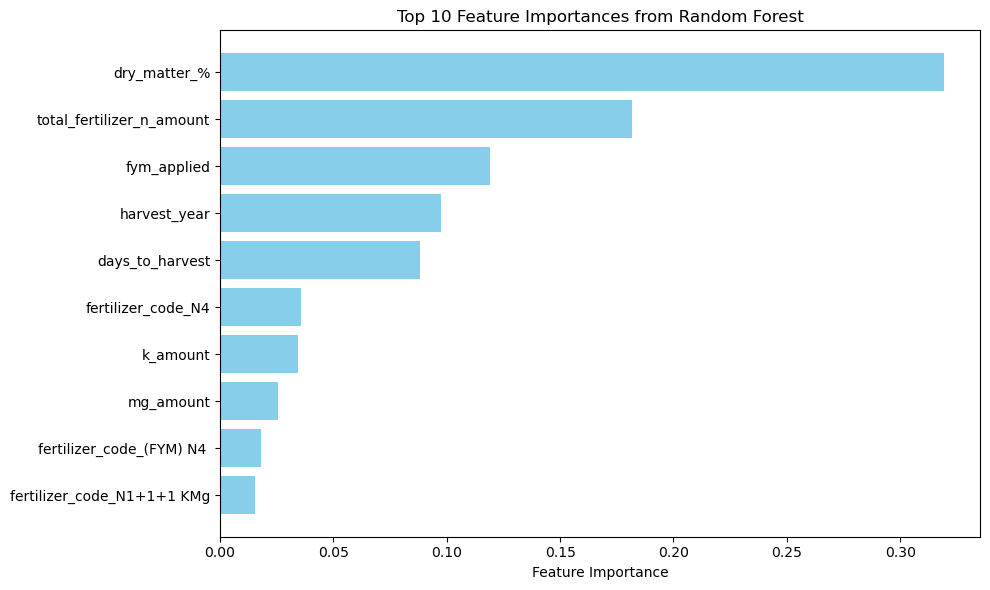


Figure 5: Top 10 factors affecting the crop yield

The dry matter was the most important factor which determined the quantity of yield followed by total fertilizer used, FYM applied, harvest year and so on.

### 3.3.4. Model Development

In this dissertation three models were selected which are described below:

1. SARIMA Model: This model was chosen for the dissertation because of its capability to model both trend and seasonality present in the agricultural yield data. Using autocorrelation (ACF) and partial autocorrelation (PACF) plots, together with grid search to decrease the Akaike Information Criterion (AIC), seasonal and non-seasonal parameters (p,d,q,P,D,Q,s) were chosen. The model was then trained on the yield series with exogenous variables such as nutrient application amounts, previous crop type, etc. These exogenous variables were included as predictors. The fitted model generated forecasts for the test set and the model was evaluated using the standard metrics like RMSE, MAE and R-squared.
2. LSTM Model: For implementing this model, the Tensorflow/Keras library was used. The architecture of the LSTM created consists of:

* One LSTM layer with 50 units.
* Dropout layer(0.2) to reduce overfitting
* Dense output layer with linear activation for regression.

For compiling the model, Adam optimizer was used and to monitor the performance of the model, the MSE was used. The model was trained for around 100 epochs with the batch size of 16.. The early stopping was used to break the training process if the loss didn’t increase for 10 epochs.

1. GRU Model: The GRU model was also built using Tensorflow/Keras library, following a similar structure to the LSTM. The architecture of the GRU consisted of following:

* One GRU layer with 50 units.
* Dropout layer (0.2) to prevent overfitting issue.
* Dense output layer with linear activation.

For compiling the model, the optimizer used was similar to the LSTM model, i.e. Adam optimizer. The loss monitored was MSE. The model was also trained for around 100 epochs with batch size of 16 and with early stopping functionality if loss doesn’t increase for 10 consecutive epochs.

### 3.3.4. Model Evaluation

All the three models were evaluated on the same test dataset using evaluation metrics like RMSE, MAE, MAPE , MSE and AIC to ensure a fair comparison. While SARIMA provided a robust statistical baseline for seasonal trends, the deep learning models (LSTM, GRU) were specifically evaluated for their capacity to grasp intricate, nonlinear correlations between yield and multivariate inputs.

# CHAPTER 4: RESULTS

## 4.1. Introduction

This chapter presents the results of the maize yield forecasting experiments that were conducted during the course of this dissertation. The results obtained from each of the models including SARIMA, LSTM and GRU has been presented in the respective subsection. The models were trained and tested on the preprocessed dataset from 1997-2013 containing yield data and multiple agronomic variables. The evaluation metrics like MSE, MAE, MAPE and RMSE recorded during the evaluation process have been presented in this chapter. Besides measuring the complexity of the model, the AIC scores were also calculated.

## 4.2. SARIMA Model Output Analysis

The SARIMA model was able to achieve an RMSE of 2.4649 and an AIC of 77.81. The AIC of this model was the lowest which signifies that the statistical models are far more simple than the deep learning models. However, the MSE, MAE, and MAPE were significantly higher than those of the deep learning models. The MSE obtained was around 6.0759, while the MAE, MAPE were around 2.2906 and 29.81 respectively. Coefficients for exogenous variables such as nutrient amounts and FYM application were mostly statistically insignificant, suggesting that SARIMA was not able to capture the complex, nonlinear interactions present in the dataset. The results from the SARIMA model are provided below:

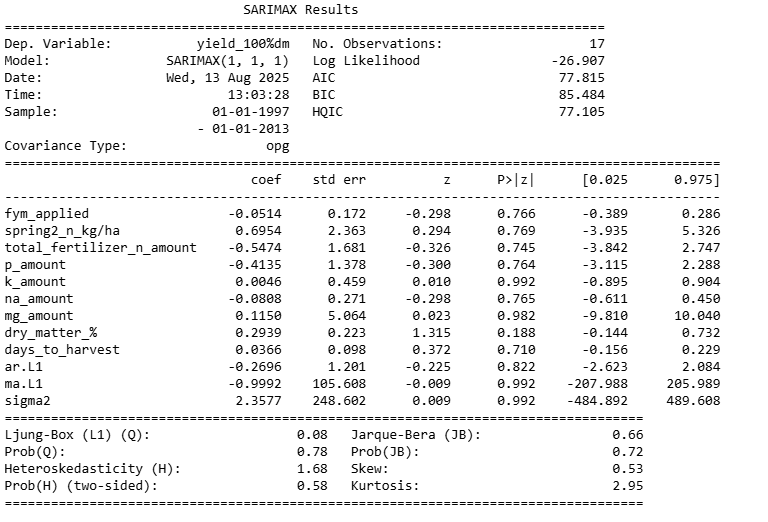


Figure 6: Results of the SARIMA Model

## 4.3. LSTM Model Performance

The performance of the LSTM model was a significant improvement over the SARIMA model. This model was able to achieve the RMSE of around 2.0753 which is around 15% reduction in RMSE as compared to SARIMA. The MAPE obtained was around 21.83%. These results indicated a better alignment between predicted and actual yields. The AIC value was comparatively very high (38023.84). This indicates the large number of trainable parameters in the network, which increases model complexity. Despite this, LSTM was effective in capturing the temporal dependencies in the multivariate data.

## 4.4. GRU Model Performance

The GRU model was the best performing model as it was able to outperform the other models, with the lowest RMSE of around 1.95 and MAPE of only 19.52. The MSE was also lowest (3.8168) among all models, which suggests that the model has strong predictive power. The AIC value was around 28935.36 which was significantly lower than LSTM indicating that its more compact architecture with fewer parameters.

## 4.5. Model Performance Comparison

Table 1: Summary of the evaluation metrics for each model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **MSE** | **MAE** | **MAPE (%)** | **RMSE** | **AIC** |
| SARIMAX | 6.0759 | 2.2906 | 29.81 | 2.4649 | 77.81 |
| LSTM | 4.6240 | 1.9930 | 22.39 | 2.1503 | 38023.84 |
| GRU | 4.1300 | 1.8114 | 20.35 | 2.0322 | 28935.36 |

From the results it is clear that both deep learning models are able to outperform the SARIMA model across all error matrics, with the GRU achieving the lowest MSE, MAE, and RMSE. The MAPE values indicate that GRU predictions deviated from actual yields by approximately 20%, compared to 22% for LSTM and nearly 30% for SARIMAX. Thus, the deep learning models were better particularly in handling nonlinearities and multivariate inputs. The best model identified was therefore the GRU model.

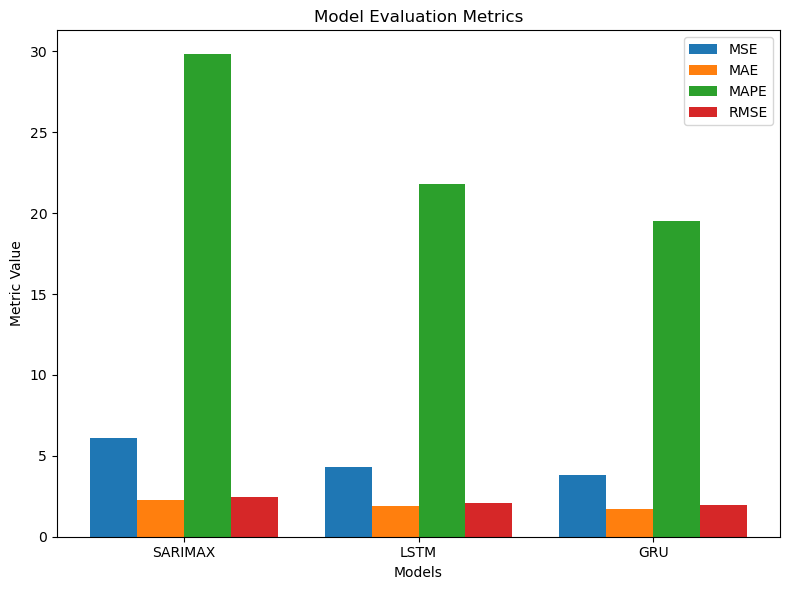


Figure 7: Grouped bar chart showing evaluation metrics for different models

# CHAPTER 5: CRITICAL DISCUSSION OF RESULTS AND RECOMMENDATIONS

## 5.1. Introduction

This study compared the performance of SARIMA, LSTM and GRU models for multivariate time series forecasting of maize yield using agronomic data from the Rothamsted Broadbalk experiments. From the results it was found that the best performing model was the GRU model which was able to achieve the best performance across all error metrics, followed by LSTM, while SARIMA produced the weakest forecasts. This section critically discusses these findings in relation to existing literature.

## 5.2. Discussion

The comparative analysis revealed that deep learning-based models, i.e., GRU showed better performance in multivariate crop yield forecasting than the traditional SARIMA approach. This is in line with earlier research (Othman et al., 2024; Mahmoud & Mohammed, 2024) revealing the strength of SARIMA in extracting seasonality but shorter capacity in extracting complex, nonlinear, and multivariate data. This is contrasted with other research such as Parreño (2023) and Selvakumar & Kasthuri (2022), which also achieved satisfactory SARIMA performance, and therefore that SARIMA performance is context-specific, more so where the univariate seasonality-driven underlying the forecasting exercises are.

The good performance of the deep learning model in this study directly supports the research by Shiri et al (2023), ArunKumar et al. (2022), and Qiu (2023). These are the research works that have pointed towards the ability of deep learning models to detect complex temporal relationships between diverse agricultural variables. Between the two deep learning models, the GRU model has the advantage of being more efficient and accurate. This concurs with Qiu (2023) and ArunKumar et al. (2022), who indicated that GRU's smaller architecture encourages quicker convergence and better generalisation in reduced datasets.

These results also reinforce the conclusions of *Bharadiya et al. (2023)*, *Mendis et al. (2024)*, and *Shook et al. (2021)* that multivariate approaches incorporating environmental, agronomic, and management variables significantly improve forecasting reliability. The differences between the findings of this study and the studies which support SARIMA, demonstrate that the nature of the dataset, including whether it contains strong nonlinear relationships, is important in deciding which modelling approach is most suitable.

## 5.3. Recommendations

Based on the exploratory analysis, results and critical discussion of this study, the following are some of the recommendations proposed for the researchers, agricultural practitioners, and policy makers:

1. Use Deep Learning Models for Complex Multivariate Forecasting

The results indicate that GRU and LSTM provide superior accuracy when predicting crop yields from multivariate inputs. These models are better in handling the complex and non-linear relationships, as shown by their lower RMSE and MAPE values. Agricultural decision-support systems should make use of the deep learning approaches instead of statistical ones in a similar context.

1. Prefer GRU for Efficiency

The GRU model performed better and yielded higher accuracy with fewer parameters than LSTM, resulting in lower AIC scores and faster training process. Thus, this efficiency of the GRU is very suitable for real-time forecasting and for applications where computational resources are limited.

1. Focus on Significant Agronomic Factors

The regression coefficients from the SARIMAX model indicated that various nutrient related variables like total nitrogen and phosphorus amounts, did not show strong statistical significance in the current dataset. This suggests that yield variation may be more influenced by interactions between multiple variables rather than single nutrient effects. So, it is recommended that interactions need to be captured in the data more precisely.

1. Improve Data Quality and Variable Tracking

The limited statistical significance was observed for many variables or predictors in the regression analysis. This highlights the need for more accurate and consistent measurement of agronomic variables. Accuracy and interpretability of the model can be enhanced by careful monitoring of management processes, environmental factors, and the timing of fertilizer applications.

1. Adopt Multivariate Data Collection Practice

The improved performance of deep learning models in this study clearly shows the importance of including multiple predictors like nutrient inputs, and crop characteristics. Thus, while building the predictive models for the crop yield prediction, it is recommended that the researchers or developers should integrate the multivariate data.

# CHAPTER 6: CONCLUSION, FUTURE ENHANCEMENTS, AND CRITICAL EVALAUTION

## 6.1. Introduction

In this chapter of the dissertation, the mapping of the aims and objectives of the project against the outcomes achieved has been provided. In addition to this, the summary of the core contributions and findings has been presented. Also this dissertation outlines the potential avenues for future work and critically evaluates the strengths and limitations of the study.

## 6.2. Mapping of Objectives to Completion

The main aim of this research was to answer the question: “How can time series forecasting be effectively applied to forecast crop yield based on fertilizer and nutrient inputs, in order to optimize agricultural productivity?”. This was broken down into specific objectives:

Table 2: Mapping of research objectives to completion status and supporting evidence

|  |  |  |
| --- | --- | --- |
| **Objective** | **Completion Status** | **Evidence from Study** |
| Review and synthesize literature on yield prediction, fertilizer optimization, and time series forecasting. | Completed | The comprehensive review which covers the works on statistical, machine learning, and deep learning methods have been presented in chapter 2 along with the key gaps identified in multivariate agricultural forecasting. |
| Collect and preprocess historical datasets on crop yield, fertilizer usage, and nutrient composition. | Completed | Data from Rothamsted Broadbalk was cleaned, normalized, encoded, and sequenced for multivariate model input (Chapter 3). |
| Implement ARIMA/SARIMA, LSTM, and GRU models and train them using the properly preprocessed crop yield data | Completed | All three models were implemented using the relevant libraries in python and they were trained and evaluated under consistent data conditions (Chapters 3 and 4). |
| Validate and compare model performance using RMSE, MAE, MAPE, and AIC | Completed | GRU achieved the lowest RMSE (2.03) and MAPE (~20%), outperforming SARIMA and LSTM (Chapter 4). |
| Analyse relationship between fertilizer/nutrient levels and predicted yields. | Completed | Feature importance, correlation analysis, and regression analysis were done in Chapter 3 which revealed the influence of nitrogen timing, farmyard manure, and dry matter content (Chapter 3). |
| Formulate recommendations for fertilizer and nutrient management strategies. | Completed | Based on the findings of the analysis, recommendations were listed in Chapter 5 which highlighted optimal nitrogen application timing and combinations to maximise yields while avoiding overapplication (Chapter 5). |

## 6.3. Conclusion of Work Carried Out

The research performed during the course of this dissertation demonstrated that:

1. Deep learning models consistently outperforms statistical baseline - Both LSTM and GRU outperformed the SARIMA in predictive accuracy across all the metrics that were used in the evaluation process. This validates the prior findings that deep learning architectures are better suited for implementing predictive tasks on multivariate agricultural datasets that have complex and non-linear dependencies.
2. GRU offers an optimal balance of accuracy and efficiency: The GRU model achieved the lowest RMSE and MAPE. This model also maintained a smaller architecture and faster convergence than LSTM. This clearly suggests GRU as the strong and possibly the best candidate for deployment where computation efficiency needs to be prioritized.
3. SARIMA Models has advantage of interpretability but lacks complexity handling: While SARIMA model had the smallest AIC of all the models tested in the experiment, it struggled to capture the nonlinear trends and multivariate dependencies in the data. Thus, it is concluded that the statistical models are better suited for the univariate or seasonality-domintated contexts.
4. Agronomic insights for supporting the actionable decisions: The regression analysis and feature important analysis revealed that correct nitrogen timing, balanced nutrient rations, and the use of organic matter (FYM) significantly enhance yields. On the other hand, the excessive use of the phosphorus and certain fertilizer blends can reduce the yields.
5. Site-specific forecasting is feasible but generalizable: Despite the fact that this study trained models on Rothamsted maize data, the methodology follows very structured and standard procedures. And because of this, the methodology is transferable to other crops and regions as well, given similar multivariate datasets are available.

Thus, it is concluded that, this dissertation study contributes both a validated forecasting workflow as well as the practical agronomic recommendations. Through this, the study bridges the gap between predictive modeling and farm-level decision-making.

## 6.4.Future Enhancements

While the dissertation was able to accomplish the stated objectives, there are several areas where the improvements can be made in the future. Some of the future enhancements are listed below:

1. Incorporation of the weather and remote sensing data: In the future version of the project, the high-resolution meteorological variables related to weather (such as temperature, precipitation, solar radiation) and vegetation indices from satellite imagery can be incorporated in model training (Haque et al., 2024; Schwalbert et al., 2020). This could possibly improve the model accuracy and make the results more robust and reliable.
2. Expand the scope of the project to multi-crop forecasting: This dissertation only focuses on the maize yield prediction. In the future, the study could adapt its methodology to other types of crops like wheat, rice etc which would further validate the generalizability of this study. This could involve making use of the transfer learning from the GRU model trained on maize data to other crop datasets.
3. Implementation of hybrid models: The future studies could also combine the statistical and deep learning approaches like SARIMA-GRU hybrids which could make use of the interpretability of statistical models and the accuracy of the deep learning models, especially for long-horizon forecasting.
4. Interpretability-focused deep learning: The future studies could also explore the explainable AI technology to make the results obtained from the deep learning models more interpretable. For this attention mechanisms or SHAP (SHapley Additive exPlanations) could be used for improving the transparency in deep learning models and help stakeholders better understand why certain predictions are made (Gholami et al., 2023).
5. Deployment as a decision support system (DSS): A web or mobile-based platform which incorporates the best model from the study could be developed which could provide farmers and agricultural planners with accessible, actionable yield predictions and fertilizer recommendations.

## 6.5. Critical Evaluation

Strengths

1. Robust Methodology: The research followed a very structured and well-established methodology which included steps like preprocessing, training and evaluation using well-established methods across all the models. This ensured fairness in comparison.
2. Use of the high quality dataset: The Rothamsted Broadbalk dataset is one of the reliable, long-term ergonomic records with rich multivariate detail.
3. Novel application: While there have been many studies which applied deep learning in agricultural forecasting, the direct comparison between the SARIMA, LSTM, and GRU on multivariate maize yield prediction using nutrient data tackles a specific and practical gap.
4. Practical results: The study on the one hand made comparative assessment of the model accuracy and on the other hand also extracted ergonomic insights for real-world decision-making.

Limitations

1. Single-site dataset: Despite the fact that the dataset is detailed, the dataset represents only a specific geography and controlled experimental setting. Due to this, the model may not generalize well in commercial farm conditions in diverse climates as it remains untested.
2. Limited external variables: The study doesn’t include weather related variables in the data, thus this potentially limits forecasting accuracy, as these factors have significant impact on yield.
3. Computational resource requirements: Deep learning models, especially the LSTM model had high number of parameter counts as indicated by the high AIC value. Thus, it required more computation power. This might limit its adoption in low-resource settings.
4. Temporal resolution: Annual yield data restricted the ability to model intra-seasonal dynamics. The higher frequency data could have revealed more granular trends.

Reliability and Viability

1. The study makes use of the multiple error metrics like RMSE, MAE, MAPE, AIC etc. The use of these numerous metrics provides the good view of the model performance from different perspectives, thereby improving the reliability of the findings.
2. The dissertation methodology and results have been well documented in detail. All the steps like preprocessing, model architectures and evaluation process has been documented properly which enhances the reproducibility of the project.
3. The external validity of this study is limited by the scope of the dataset. Wider testing might be necessary for broader application.

Contribution to knowledge

This dissertation makes a significant contribution by empirically validating that GRU outperforms both LSTM and SARIMA in a multivariate nutrient-based maize yield forecasting. It also showed the practical benefits of feature-level ergonomic analysis and also showed the effectiveness of predictive modeling.

## 6.6 Final Remarks

The study confirms that advanced deep learning techniques, particularly GRU, can yield high performance and significant improvements in agricultural yield forecasting as compared to the traditional statistical approach. This dissertation emphasizes the importance of integrating ergonomic expertise with machine learning to make sure that the predictions are not only accurate but also are actionable. The world today is facing food security pressures, climate variability due to global warming, and resource limitations. In such scenarios the use of the predictive systems have the potential for empowering farmers, policymakers, and researchers in making effective data driven decisions. With further development via integration of additional data sources, and deployment in user-friendly platforms, this dissertation project could act as the foundation for building scalable, sustainable agricultural decision support systems.

# CHAPTER 7: SELF-REFLECTION

## 7.1. Introduction

This chapter provides a reflection on my experience which I acquired during the course of this dissertation. This chapter reflects upon the skills that I have gained, challenges encountered and lessons learned. Since the project was related to data science and machine learning, the process demanded a combination of technical skills, ability to think critically and effective project management.

## 7.2. Description and Feelings

At the start, I felt both excited and a bit nervous. The field of agriculture was a bit new for me, and I knew that I had a lot to learn about agronomy and fertilizer management. However, I was still confident that my programming and machine learning skills would help me close this knowledge gap.

The initial literature review was very interesting. But at the same time also, I felt a bit overwhelmed because of the wide range of related studies. Building the first SARIMA model was a big milestone for me. The success of building the SARIMA model showed me that the goals were realistic and achievable. However, building the deep learning models were comparatively more challenging and it brought few issues with convergence and hyperparameter tuning. But in the end, these came out to be valuable learning experiences.

## 7.3. Skills Developed

**Technical Skills**

* Data Preprocessing & Feature Engineering: I gained skills in managing missing values, encoding categorical variables, normalisation, and preparing sequential data for time series forecasting.
* Model Development: I learned to use python and its third party libraries. I developed skills of using TensorFlow, Keras and statsmodels and learned the skills to implement SARIMA, LSTM and GRU.
* Model Evaluation: I also learned the importance of the different evaluation metrics and learned to apply RMSE, MAE, MAPE, and AIC metrics, and linking performance to practical agricultural decision-making.

**Research Skills**

* Learned to conduct a critical literature review and identify the gaps in the studies
* Developed a skill of designing machine learning related experiments that ensure fair model comparison.
* Last but not least, I also developed skills of structuring a coherent academic narrative with evidence-based conclusions.

**Professional Skills**

* Time management skills, problem-solving when encountered with unexpected dataset or model issues and communicating technical findings were some of the professional skills which I acquired during the course of this dissertation project.

## 7.4. Challenges and Response

**Data Complexity:** The dataset obtained was not in proper form which could be directly used. It had missing values, inconsistent variable formats and coded categorical variables. Thus, careful cleaning and encoding were important to ensure accuracy.

**Balancing Accuracy and Interpretability:** While deep learning models were able to yield higher accuracy, its “black box” nature limits their interpretability. To address this, regression analysis and feature importance analysis were done which partially address this challenge.

**Computational Constraints:** Training LSTM and GRU models were heavy on the PC. The use of efficient batch sizes, early stopping, and the more compact GRU architecture helped to manage this.

**Scope Management:** Initially, there was a temptation to include weather and remote sensing data. Maintaining focus on the original objectives ensured project feasibility.

## 7.5. Lessons Learned from Dissertation

This dissertation helped me understand the value of combining domain knowledge with machine learning expertise. From the dissertation I learned that agricultural forecasting is not only a technical challenge but it is also necessary to understand crop cycles, nutrient roles, and farming practices to produce the results that are meaningful.

The experience also highlighted the importance of reproducibility and documentation for the project. The detailed documentation and recording of the preprocessing steps, model training, evaluation etc helped to make sure that results could be replicated and validated.

## 7.6 Future Application

I believe that the skills which I acquired from this dissertation project are directly transferable to data science roles in agriculture and beyond. I am planning to explore integrating weather and remote sensing data into future models and apply explainable AI techniques like SHAP values to improve the transparency of models.

## 7.7 Conclusion

The journey through this dissertation not only helped me strengthen my technical and professional capabilities but it also helped me progress from uncertainty in a new domain to deliver a reliable forecasting system with actional insights. This project developed a sense of appreciation inside me for the interdisciplinary approaches. I learned to maintain balance between accuracy, interpretability and practical impact in a machine learning based project. I believe that these skills which I acquired through this dissertation will be very valuable for me in my future career as a data scientist.

# CHAPTER 8: PROJECT MANAGEMENT

## 8.1. Introduction

For the completion of the project, effective project management was crucial. This project was an interdisciplinary project which included agricultural science, statistical modeling, deep learning, and extensive data preprocessing. It was very important to adopt a structured and methodological approach to plan, monitor and control the scope, time, risk and quality of the project.

The project management methodology used in the instance of this dissertation project was based on PMBOK standards and PRINCE2 methodology (Sobieraj et al., 2021).

The process began with scope and objective definition. This was then followed by planning the project schedule using a base Gantt chart, identifying the project dependencies, creating milestones, assigning the required resources, and delivering outputs on a scheduled timeline. In this chapter the Work Breakdown Structure (WBS), Dependencies between tasks, Baseline Gantt Chart, Project milestones and Key deliverables has been outlined.

## 8.2. Work Breakdown Structure (WBS)

The WBS was used for breaking the project into small manageable sprints and sub-tasks. Each of these tasks were created in such a way that they had clearly defined milestones. The WBS for the project is provided below:

**Dissertation Project – Maize Yield Forecasting**

**Project Initiation**

* 1.1 Prepare scope, aims and objectives for the project
* 1.2 Identification of the stakeholders
* 1.3 Identification of possible project risks and their mitigation planning

**Literature Review**

* 2.1 collect the literatures
* 2.2 Perform comprehensive review of collected literatures
* 2.3 Prepare the literature review chapter with identified gaps

**Data Collection & Preprocessing**

* 3.1 collect dataset from Rothamsted Broadbalk archive
* 3.2 Perform extensive data cleaning
* 3.3 Feature engineering (nutrient levels, dates, FYM usage)
* 3.4 Performing data encoding and normalization to make them consistent
* 3.5 Sequence generation for deep learning models

**Exploratory Data Analysis**

* 4.1 Perform correlation analysis
* 4.2 Visualize data (heatmaps, time series plots, boxplots) and get insights
* 4.3 Perform regression and feature importance analysis

**Model Development**

* 5.1 Build SARIMA model
* 5.2 Build LSTM model
* 5.3 Build GRU model

**Model Evaluation**

* 6.1 Finalize the evaluation metrics (RMSE, MAE, MAPE, AIC)
* 6.2 Do comparative analysis of all models based on these metrics

**Results & Discussion**

* 7.1 Document the results
* 7.2 Critical discussion linked to literature

**Recommendations & Conclusion**

* 8.1 Practical recommendations for fertilizer management
* 8.2 Document the conclusions and implications of the research

**Documentation & Submission**

* 9.1 Prepare the draft of final report
* 9.2 Proofreading and formatting
* 9.3 Submission

## 8.3. Task Dependencies

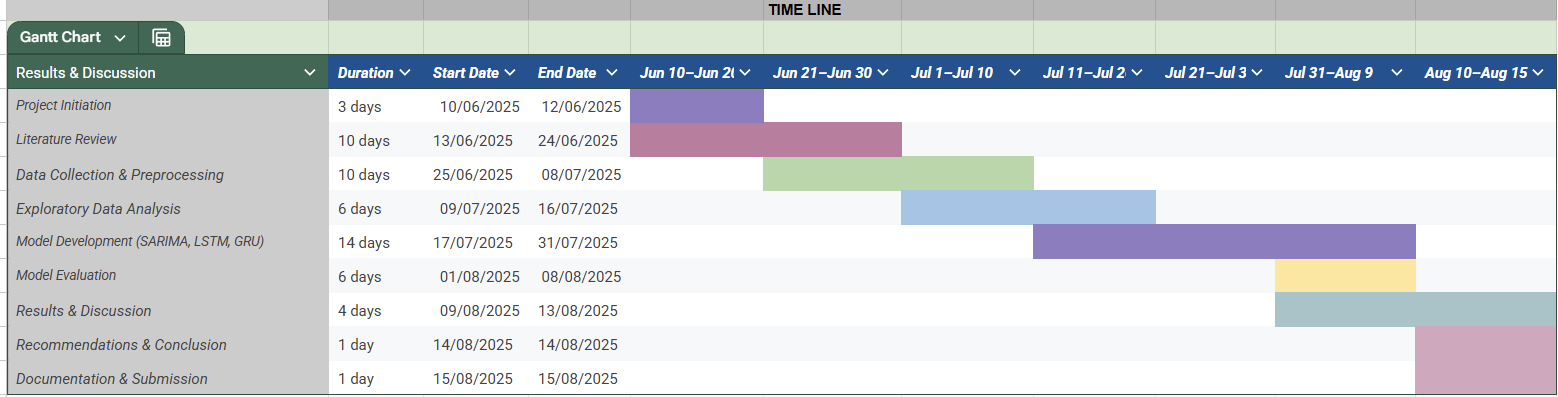
In this project, certain tasks were sequential and hugely dependent on the completion of earlier tasks before proceeding.

* **Data preprocessing** (3.2–3.5) depended on **data collection** (3.1).
* **Model development** (5.1–5.3) depended on both **EDA** (4.1–4.3) and **preprocessed datasets**.
* **Model evaluation** (6.1–6.2) could only begin after all three models were developed.
* **Results & discussion** (7.1–7.2) required completion of evaluation.
* **Final recommendations** (8.1) depended on the insights generated from discussion.

These dependencies helped to make sure the project flowed in a logical manner and minimized rework.

## 8.4. Baseline Gantt Chart

The Gantt chart prepared during the initial planning phase is provided below:

Figure 8: Gantt chart created for the project

## 8.5. Milestones

Proper project milestones were established which helped to track progress and ensure deliverables met deadlines:

1. Project Approval (12/06/2025): Got approval for the project scope, objectives, and plan.
2. Literature Review Completed (24/06/2025): Prepared the full draft of the literature review chapter.
3. Data Ready for Modelling (08/07/2025): Performed preprocessing of data and got the cleaned and proper data.
4. EDA Completed (16/07/2025): Performed EDA and other analysis to get insights from data.
5. All Models Developed (31/07/2025): SARIMA, LSTM, and GRU implemented using the respective libraries in Python.
6. Evaluation Completed (08/08/2025): Completed comparative analysis and final comparison table created and documented.
7. Results & Discussion Draft Completed (13/08/2025): Fully documented the results and discussion in the draft dissertation report.
8. Final Report Completed (14/08/2025): Dissertation report completed and finalized formatting.
9. Dissertation Submission (15/08/2025): Final dissertation submitted.

## 8.6. Risk Management

To manage the possible risks which may arise during the course of the dissertation project, a risk register was created. The risk register developed for this project is provided below:

Table 3: Risk Register

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Likelihood** | **Impact** | **Mitigation** |
| Poor data quality and inconsistencies | Medium | High | Implement the comprehensive preprocessing and proper validation techniques |
| High computation requirements | Medium | Medium | Make use of the techniques like early stopping and also simplify the architecture by removing unnecessary complexities. |
| Scope creep | Low | High | Limit the scope and strictly adhere to WBS |
| Delays due to illness or technical issues | Medium | Medium | Include buffer time in the project timeline during the project planning phase. |

## 8.7 Monitoring and Control

For tracking the progress of the project, a combination of structured monitoring tools were used. A predefined checklist which aligned with the WBS was created in the Trello application and the task completion was reviewed against this checklist. It helped to make sure that all the planned activities were completed in sequence and the standards were properly maintained. The regular update was made to the baseline Gantt Chart to reflect upon the actual progress made against the planned schedule. For properly managing the different versions of the code, version controlling was done via the use of GitHub. This helped to ensure both secure backup and traceability of changes. In addition to this, regular meetings were conducted with the project supervisor for reviewing the progress, addressing issues, and adjusting priorities where necessary to make sure the project remained on track and quality was maintained.

## 8.8 Lessons Learned in Project Management

This project benefitted from following management practices:

1. The decomposition of the project pipeline into the small, manageable sprints in the WBS made it easy to manage the tasks and made scheduling more accurate.
2. The analysis and building of the dependencies into the plan helped to prevent premature starting of any task which helped to reduce the rework.
3. Making use of the baseline Gantt chart helped to identify task schedules and track progress of each task.
4. Including buffer time proved very crucial when model building, issue debugging and training took longer than expected.

## 8.9. Conclusion

Thus, managing the project into a structured approach helped to ensure that the research progressed in a logical and timely manner. The use of different project management tools and techniques like baseline Gantt chart, Trello, milestone-base tracking provided a very clear blueprint for tracking the progress of the dissertation from project initiation to submission. The project was completed on schedule and to high standards due to the implementation of rigorous monitoring and the inclusion of flexibility for unanticipated situations.

.

# REFERENCE

Aggarwal, P. K., Kalra, N., Chander, S., & Pathak, H. (2006). InfoCrop: a dynamic simulation model for the assessment of crop yields, losses due to pests, and environmental impact of agro-ecosystems in tropical environments. I. Model description. *Agricultural systems*, *89*(1), 1-25.

Ali, I. M. (2024). A guide for positivist research paradigm: From philosophy to methodology. *Idealogy Journal*, *9*(2).

Airlangga, G. (2024). Enhancing medical diagnostics with ensemble machine learning: a comparative study of gradient boosting, XGBoost, LightGBM, and blended models. *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, *9*(2), 1117-1132.

ALnaass, N. S., Agil, H. K., & Ibrahim, H. K. (2021). Use of fertilizers or importance of fertilizers in agriculture. *International Journal of Advanced Academic Studies*, *3*(2), 52-57.

Amor, L. B., Lahyani, I., & Jmaiel, M. (2016, August). Recursive and rolling windows for medical time series forecasting: a comparative study. In *2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)* (pp. 106-113). IEEE.

Arora, N. K. (2018). Agricultural sustainability and food security. *Environmental Sustainability*, *1*(3), 217-219.

ArunKumar, K. E., Kalaga, D. V., Kumar, C. M. S., Kawaji, M., & Brenza, T. M. (2022). Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alexandria engineering journal*, *61*(10), 7585-7603.

Ayo, A. O., Owolawi, P. A., & Ojo, J. S. (2024). Time series forecasting of rain-induced attenuation using coupled ARIMA-SARIMA models for radio propagation applications over subtropical locations. *Edelweiss Appl. Sci. Technol.*, *8*(6).

Balasubramanian, V., Alves, B., Aulakh, M., Bekunda, M., Cai, Z., Drinkwater, L., ... & Oenema, O. (2004). Crop, environmental, and management factors affecting nitrogen use efficiency. *Agriculture and the nitrogen cycle*, *65*, 19-33.

Bharadiya, J. P., Tzenios, N. T., & Reddy, M. (2023). Forecasting of crop yield using remote sensing data, agrarian factors and machine learning approaches. *Journal of Engineering Research and Reports*, *24*(12), 29-44.

Bontempi, G., Ben Taieb, S., & Le Borgne, Y. A. (2012). Machine learning strategies for time series forecasting. In *European Big Data Management and Analytics Summer School* (pp. 62-77). Berlin, Heidelberg: Springer Berlin Heidelberg.

Chatfield, C. (2013). *The analysis of time series: theory and practice*. Springer

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *Peerj computer science*, *7*, e623.

Chlingaryan, A., Sukkarieh, S., & Whelan, B. (2018). Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. *Computers and electronics in agriculture*, *151*, 61-69.

Coble, K. H., Mishra, A. K., Ferrell, S., & Griffin, T. (2018). Big data in agriculture: A challenge for the future. *Applied Economic Perspectives and Policy*, *40*(1), 79-96.

Fischer, R. A., Byerlee, D., & Edmeades, G. (2014). Crop yields and global food security. *ACIAR: Canberra, ACT*, 8-11.

Gholami, H., Mohammadifar, A., Golzari, S., Song, Y., & Pradhan, B. (2023). Interpretability of simple RNN and GRU deep learning models used to map land susceptibility to gully erosion. *Science of the Total Environment*, *904*, 166960.

Halabaku, E., & Bytyçi, E. (2024). Overfitting in Machine Learning: A Comparative Analysis of Decision Trees and Random Forests. *Intelligent Automation & Soft Computing*, *39*(6).

Haque, M. A., Reza, M. N., Ali, M., Karim, M. R., Ahmed, S., Lee, K. D., ... & Chung, S. O. (2024). Effects of environmental conditions on vegetation indices from multispectral images: A review. *Korean Journal of Remote Sensing*, *40*(4), 319-341.

Henderi, H., Wahyuningsih, T., & Rahwanto, E. (2021). Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer. *International Journal of Informatics and Information Systems*, *4*(1), 13-20.

Hodson, T. O. (2022). Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development Discussions*, *2022*, 1-10.

Jabed, M. A., & Murad, M. A. A. (2024). Crop yield prediction in agriculture: A comprehensive review of machine learning and deep learning approaches, with insights for future research and sustainability. *Heliyon*, *10*(24).

Jadon, A., Patil, A., & Jadon, S. (2024, January). A comprehensive survey of regression-based loss functions for time series forecasting. In *International Conference on Data Management, Analytics & Innovation* (pp. 117-147). Singapore: Springer Nature Singapore.

Jeba, R. P., Kirthiga, S. M., Issac, A. M., Bindhu, V. M., Srinivasan, R., & Narasimhan, B. (2024). An improved framework for mapping and assessment of dynamics in cropping pattern and crop calendar from NDVI time series across a heterogeneous agro-climatic region. *Environmental Monitoring and Assessment*, *196*(11), 1141.

Jeong, J. H., Resop, J. P., Mueller, N. D., Fleisher, D. H., Yun, K., Butler, E. E., ... & Kim, S. H. (2016). Random forests for global and regional crop yield predictions. *PloS one*, *11*(6), e0156571.

Kazemi, M. (2024). Support vector machine in ultrahigh-dimensional feature space. *Journal of Statistical Computation and Simulation*, *94*(3), 517-535.

Khashei, M., & Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing, 11*(2), 2664–2675.<https://doi.org/10.1016/j.asoc.2010.10.015>

Koch, I. (2013). *Analysis of multivariate and high-dimensional data* (Vol. 32). Cambridge University Press.

Kurumatani, K. (2020). Time series forecasting of agricultural product prices based on recurrent neural networks and its evaluation method. *SN Applied Sciences*, *2*(8), 1434.

Kmytiuk, T., Majore, G., & Bilyk, T. (2024). Time series forecasting of price of the agricultural products using data science. *Agricultural and Resource Economics: International Scientific E-Journal*, *10*(3), 5-33.

Liliane, T. N., & Charles, M. S. (2020). Factors affecting yield of crops. *Agronomy-climate change & food security*, *9*, 9-24.

Liu, Z., Zhu, Z., Gao, J., & Xu, C. (2021). Forecast methods for time series data: A survey. *Ieee Access*, *9*, 91896-91912.

Lobell, D. B., & Burke, M. B. (2010). On the use of statistical models to predict crop yield responses to climate change. *Agricultural and forest meteorology*, *150*(11), 1443-1452.

Majka, M. (2024). Seasonal time series analysis: Why sarima outshines arima. *ResearchGate. Published September*, *19*, 2024.

Masini, R. P., Medeiros, M. C., & Mendes, E. F. (2023). Machine learning advances for time series forecasting. *Journal of economic surveys*, *37*(1), 76-111.

Mendelsohn, R. O., & Massetti, E. (2017). The use of cross-sectional analysis to measure climate impacts on agriculture: theory and evidence. *Review of Environmental Economics and Policy*.

Mendis, K., Wickramasinghe, M., & Marasinghe, P. (2024, April). Multivariate time series forecasting: A review. In *Proceedings of the 2024 2nd Asia Conference on Computer Vision, Image Processing and Pattern Recognition* (pp. 1-9).

Mahmoud, A., & Mohammed, A. (2024). Leveraging hybrid deep learning models for enhanced multivariate time series forecasting. *Neural Processing Letters*, *56*(5), 223.

Moazzen, F., & Hossain, M. J. (2024). Multivariate deep learning long short-term memory-based forecasting for microgrid energy management systems. *Energies*, *17*(17), 4360.

Mwanga, D., Ong’ala, J., & Orwa, G. (2017). Modeling sugarcane yields in the Kenya sugar industry: A SARIMA model forecasting approach. *International Journal of Statistics and Applications*, *7*(6), 280-288.

Nokeri, T. C. (2021). Forecasting using ARIMA, SARIMA, and the additive model. In *Implementing Machine Learning for Finance: A Systematic Approach to Predictive Risk and Performance Analysis for Investment Portfolios* (pp. 21-50). Berkeley, CA: Apress.

Othman, S. A., Jameel, H. H., & Abdulazeez, S. T. (2024). Comparative Analysis of Univariate SARIMA and Multivariate VAR Models for Time Series Forecasting: A Case Study of Climate Variables in Ninahvah City, Iraq. *Mathematical Problems of Computer Science*, *61*, 27-49.

Parreño, S. J. E. (2023). Forecasting quarterly rice and corn production in the Philippines: A comparative study of seasonal ARIMA and Holt-Winters models. *ICTACT Journal on Soft Computing, 14*(2)

Petropoulos, F., & Spiliotis, E. (2021). The wisdom of the data: Getting the most out of univariate time series forecasting. *Forecasting*, *3*(3), 478-497.

Poslavskaya, E., & Korolev, A. (2023). Encoding categorical data: Is there yet anything'hotter'than one-hot encoding?. *arXiv preprint arXiv:2312.16930*.

Qiu, Y. (2023, May). Rice yield prediction based on LSTM and GRU. In *International Conference on Computer Graphics, Artificial Intelligence, and Data Processing (ICCAID 2022)* (Vol. 12604, pp. 487-493). SPIE.

Ramamoorthy, P., Sathiyamurthi, S., Pavithra, A., Sivasakthi, M., & Praveen Kumar, S. (2024). Impact of Major Nutrients Fertilizer Application on Soil Pollution and Management Measures. In *Soil, Water Pollution and Mitigation Strategies: A Spatial Approach* (pp. 301-314). Cham: Springer Nature Switzerland.

Ramirez-Villegas, J., & Challinor, A. (2012). Assessing relevant climate data for agricultural applications. *Agricultural and forest meteorology*, *161*, 26-45.

Rogachev, A. F., & Simonov, A. B. (2022, August). Systematic analysis of retrospective crop yields time series based on their structure identification. In *IOP Conference Series: Earth and Environmental Science* (Vol. 1069, No. 1, p. 012014). IOP Publishing.

Rothamsted Research. (2024). *Broadbalk Winter Wheat Experiment: Grain Yield (FMYIELD)* [Dataset]. Electronic Rothamsted Archive (e-RA).<https://www.era.rothamsted.ac.uk/dataset/rbk1/01-FMYIELD>

Schwalbert, R. A., Amado, T., Corassa, G., Pott, L. P., Prasad, P. V., & Ciampitti, I. A. (2020). Satellite-based soybean yield forecast: Integrating machine learning and weather data for improving crop yield prediction in southern Brazil. *Agricultural and Forest Meteorology*, *284*, 107886

Selvakumar, S., & Kasthuri, V. (2022). Wheat Production Prediction in India using ARIMA, Neural Network and Fuzzy Time Series.

Seuring, S., Stella, T., & Stella, M. (2021). Developing and publishing strong empirical research in sustainability management—Addressing the intersection of theory, method, and empirical field. *Frontiers in Sustainability*, *1*, 617870.

Shahhosseini, M., Hu, G., Huber, I., & Archontoulis, S. V. (2021). Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt. *Scientific reports*, *11*(1), 1606.

<https://doi.org/10.1038/s41598-020-80820-1>

Shook, J., Gangopadhyay, T., Wu, L., Ganapathysubramanian, B., Sarkar, S., & Singh, A. K. (2021). Crop yield prediction integrating genotype and weather variables using deep learning. *Plos one*, *16*(6), e0252402.

Shiri, F. M., Perumal, T., Mustapha, N., & Mohamed, R. (2023). A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU. *arXiv preprint arXiv:2305.17473*.

Shuman, L. M. (2017). Micronutrient fertilizers. In *Nutrient use in crop production* (pp. 165-195). CRC Press.

Sobieraj, J., Metelski, D., & Nowak, P. (2021). PMBoK vs. PRINCE2 in the context of Polish construction projects: Structural Equation Modelling approach. *Archives of Civil Engineering*, *67*(2).

Thota, V. R. (2025). *Comparative Study of Time Series Forecasting for Trucking Shipments: Evaluating Arima, LSTM, and Hybrid Arima-LSTM Models* (Master's thesis, State University of New York at Binghamton).

Zevallos-Aquije, A., Zevallos-Aquije, A., Palomino-Salcedo, K., & Salas-Bolaños, R. A. (2025). Improving Accuracy in Agricultural Forecasts: Evaluation of SARIMA and Forecast Models for Artichoke Prices. *Research on World Agricultural Economy*, 273-290.

:

# APPENDIX I: DATA LINK

Dataset: <https://www.era.rothamsted.ac.uk/dataset/rbk1/01-FMYIELD>

|  |
| --- |
|  |

# APPENDIX II: CODE SNIPPETS

|  |
| --- |
| import pandas as pd # for reading the data file  import numpy as np  import matplotlib.pyplot as plt # for plotting graphs and all  import seaborn as sns # also for plotting graphs  from sklearn.preprocessing import LabelEncoder  import statsmodels.api as sm  from sklearn.ensemble import RandomForestRegressor  from sklearn.model\_selection import train\_test\_split  from sklearn.metrics import mean\_squared\_error, r2\_score, mean\_absolute\_error  from sklearn.preprocessing import OneHotEncoder  import pandas as pd  import matplotlib.pyplot as plt  from statsmodels.tsa.api import VAR  import math  import tensorflow as tf  from tensorflow.keras.models import Sequential  from tensorflow.keras.layers import Dense, LSTM, GRU, Dropout  from tensorflow.keras.callbacks import EarlyStopping  import statsmodels.api as sm  from sklearn.preprocessing import StandardScaler, MinMaxScaler  import warnings  warnings.filterwarnings("ignore")  yield\_data = pd.read\_csv('01-FMYIELD/01-BKfmyield9717/yield\_data.csv', na\_values=['Nil','nil','\*', "MISSING", 'Missing', "missing"]) # using pandas to read the dataset  yield\_data['mg\_factor\_level'].unique()  yield\_data.head() # printing the first few rows of the dataset to perform manual inspection  # looking into the dimension of the dataset  yield\_data.shape  yield\_data.dtypes  yield\_data.isna().sum()  # Convert numeric columns  numeric\_cols = [  'fym\_applied', 'spring1\_n\_kg/ha', 'spring2\_n\_kg/ha', 'total\_fertilizer\_n\_amount',  'p\_amount', 'k\_amount', 'na\_amount', 'mg\_amount', 'dry\_matter\_%', 'yield\_100%dm'  ]  for col in numeric\_cols:  yield\_data[col] = pd.to\_numeric(yield\_data[col])  yield\_data['mg\_factor\_level'].unique()  yield\_data[numeric\_cols].describe()  # Correlation heatmap  plt.figure(figsize=(12, 8))  sns.heatmap(yield\_data[numeric\_cols].corr(), annot=True, cmap='coolwarm')  plt.title("Correlation Heatmap")  plt.tight\_layout()  plt.show()  plt.figure(figsize=(14, 6))  sns.boxplot(x='fertilizer\_code', y='yield\_100%dm', data=yield\_data)  plt.xticks(rotation=90)  plt.title("Yield by Fertilizer Code")  plt.tight\_layout()  plt.show()  plt.figure(figsize=(10, 5))  sns.lineplot(x='harvest\_year', y='yield\_100%dm', data=yield\_data, marker='o')  plt.title("Yield over Harvest Years")  plt.tight\_layout()  plt.show()  # Grouped summary by fertilizer treatment  group\_summary = yield\_data.groupby("fertilizer\_code")["yield\_100%dm"].agg(["count", "mean", "std", "min", "max"])  print("\nGrouped Yield Summary by Fertilizer Code:\n", group\_summary)  drop\_threshold = 0.80 \* len(yield\_data)  yield\_data = yield\_data.dropna(axis=1, thresh = int(yield\_data.shape[0] - drop\_threshold))  # Mode for categorical variables with low missingness  for col in ['fertilizer\_code', 'n\_factor\_level', 'n\_timing', 'spring1\_n\_date', 'p\_factor\_level',  'k\_factor\_level', 'na\_factor\_level', 'mg\_factor\_level']:  yield\_data[col] = yield\_data[col].fillna(yield\_data[col].mode()[0])  # Impute high-missing categorical date fields with "Unknown" or placeholder date  for col in ['p\_date', 'k\_date', 'mg\_date']:  yield\_data[col] = yield\_data[col].fillna("Unknown")  # Impute numeric fields (though none seem missing here, just in case)  # Example: use median or mean if needed  for col in ['p\_amount', 'k\_amount', 'na\_amount', 'mg\_amount']:  if yield\_data[col].isna().sum() > 0:  yield\_data[col] = yield\_data[col].fillna(yield\_data[col].median())  # 'note' column: probably text notes — fill with empty string or drop  yield\_data['note'] = yield\_data['note'].fillna("")  # 'spring1\_n\_kg/ha' and 'total\_fertilizer\_n\_amount' are highly correlated (0.82)  # Drop one of them to reduce multicollinearity  yield\_data = yield\_data.drop(columns=['spring1\_n\_kg/ha'])  cols\_step1 = ['strip', 'section', 'plot', 'section\_1926-67', 'note']  yield\_data = yield\_data.drop(columns=cols\_step1, errors='ignore')  cols\_step2 = ['fym\_date', 'spring1\_n\_date', 'spring2\_n\_date', 'p\_date', 'k\_date', 'na\_date', 'mg\_date']  yield\_data = yield\_data.drop(columns=cols\_step2, errors='ignore')  cols\_step3 = ['fym\_factor\_level', 'n\_factor\_level', 'p\_factor\_level', 'k\_factor\_level', 'na\_factor\_level', 'mg\_factor\_level']  yield\_data = yield\_data.drop(columns=cols\_step3, errors='ignore')  yield\_data = yield\_data.drop(columns=['crop', 'previous\_crop'], errors='ignore')  # Convert date columns to datetime and create 'days\_to\_harvest'  yield\_data['sow\_date'] = pd.to\_datetime(yield\_data['sow\_date'])  yield\_data['harvest\_date'] = pd.to\_datetime(yield\_data['harvest\_date'])  yield\_data['days\_to\_harvest'] = (yield\_data['harvest\_date'] - yield\_data['sow\_date']).dt.days  # Drop original date columns  yield\_data = yield\_data.drop(columns=['sow\_date', 'harvest\_date'])  # Separate features and target  X = yield\_data.drop(columns=['yield\_100%dm'])  y = yield\_data['yield\_100%dm']  categorical\_cols = ['fertilizer\_code', 'n\_timing', 'cultivar']  # Initialize OneHotEncoder with drop='first' to avoid dummy trap  encoder = OneHotEncoder(drop='first', sparse\_output=False)  encoded\_dfs = []  for col in categorical\_cols:  col\_reshaped = X[[col]] # DataFrame slice  encoded\_array = encoder.fit\_transform(col\_reshaped)  encoded\_col\_names = encoder.get\_feature\_names\_out([col])  encoded\_df = pd.DataFrame(encoded\_array, columns=encoded\_col\_names, index=X.index)  encoded\_dfs.append(encoded\_df)  # Concatenate all encoded categorical columns  X\_cat\_df = pd.concat(encoded\_dfs, axis=1)  # Drop original categorical columns and concatenate encoded ones  X\_numeric = X.drop(columns=categorical\_cols)  X\_final = pd.concat([X\_numeric, X\_cat\_df], axis=1)  # Add constant for intercept  X\_final = sm.add\_constant(X\_final)  # Fit OLS model  model = sm.OLS(y, X\_final)  results = model.fit()  print(results.summary())  # Drop the constant column since RF doesn't need it  X\_rf = X\_final.drop(columns=['const'])  # Split into train and test sets  X\_train, X\_test, y\_train, y\_test = train\_test\_split(  X\_rf, y, test\_size=0.2, random\_state=42  )  # Initialize Random Forest regressor  rf = RandomForestRegressor(n\_estimators=100, random\_state=42)  # Fit model  rf.fit(X\_train, y\_train)  # Predict on test set  y\_pred = rf.predict(X\_test)  # Evaluate  mse = mean\_squared\_error(y\_test, y\_pred)  r2 = r2\_score(y\_test, y\_pred)  print(f"Random Forest Regression Results:")  print(f"Mean Squared Error: {mse:.2f}")  print(f"R-squared: {r2:.2f}")  # After fitting Random Forest and getting feature importances (same as before)  importances = rf.feature\_importances\_  features = X\_rf.columns  feat\_imp\_df = pd.DataFrame({  'feature': features,  'importance': importances  }).sort\_values(by='importance', ascending=False)  # Plot top 10 important features  top\_features = feat\_imp\_df.head(10)  plt.figure(figsize=(10, 6))  plt.barh(top\_features['feature'][::-1], top\_features['importance'][::-1], color='skyblue')  plt.xlabel('Feature Importance')  plt.title('Top 10 Feature Importances from Random Forest')  plt.tight\_layout()  plt.show()  def calculate\_metrics(y\_true, y\_pred):  mse = mean\_squared\_error(y\_true, y\_pred)  mae = mean\_absolute\_error(y\_true, y\_pred)  mape = np.mean(np.abs((y\_true - y\_pred) / y\_true)) \* 100  rmse = math.sqrt(mse)  r2 = r2\_score(y\_true, y\_pred)  return mse, mae, mape, rmse, r2  # Aggregate to yearly series (mean) - adjust if you prefer median or grouping by fertilizer etc.  # We take yearly mean of 'yield\_100%dm' and also build yearly exogenous features as means.  df = yield\_data.copy()  # Ensure numeric columns are numeric  numeric\_cols = ['fym\_applied', 'spring2\_n\_kg/ha', 'total\_fertilizer\_n\_amount',  'p\_amount', 'k\_amount', 'na\_amount', 'mg\_amount', 'dry\_matter\_%',  'yield\_100%dm', 'days\_to\_harvest']  for c in numeric\_cols:  df[c] = pd.to\_numeric(df[c], errors='coerce')  # Aggregate by harvest\_year (mean). You can change aggregation as needed.  yearly = df.groupby('harvest\_year')[numeric\_cols].mean().reset\_index()  yearly = yearly.sort\_values('harvest\_year').reset\_index(drop=True)  # set index as datetime-like for plotting convenience (using year)  yearly['date'] = pd.to\_datetime(yearly['harvest\_year'].astype(int).astype(str) + '-01-01')  yearly = yearly.set\_index('date')  print("Years available:", yearly.index.year.min(), "-", yearly.index.year.max())  yearly.head()  # Prepare series and exogenous features  TARGET = 'yield\_100%dm'  # choose exogenous predictors (drop the target)  exog\_features = ['fym\_applied', 'spring2\_n\_kg/ha', 'total\_fertilizer\_n\_amount',  'p\_amount', 'k\_amount', 'na\_amount', 'mg\_amount', 'dry\_matter\_%', 'days\_to\_harvest']  # drop rows with NaN in target  yearly = yearly.dropna(subset=[TARGET])  yearly = yearly.dropna(subset=exog\_features) # ensure exog are present  y = yearly[TARGET]  X = yearly[exog\_features]  print(f"Data points: {len(y)}")  # Train-test split by time (e.g., last 20% years for test)  n = len(y)  test\_size = max(1, int(n \* 0.2))  train\_size = n - test\_size  y\_train, y\_test = y.iloc[:train\_size], y.iloc[train\_size:]  X\_train, X\_test = X.iloc[:train\_size], X.iloc[train\_size:]  print("Train years:", y\_train.index.year.min(), "-", y\_train.index.year.max())  print("Test years:", y\_test.index.year.min(), "-", y\_test.index.year.max())  # We'll fit a simple SARIMAX(p,d,q) with exog. Choose small orders as a baseline (e.g., p=1,d=1,q=1).  # For a production model you should grid-search p,d,q using AIC/BIC or cross-validation.  order = (1, 1, 1)  print("Fitting SARIMAX ... (may take a moment)")  sarimax\_model = sm.tsa.SARIMAX(y\_train, exog=X\_train, order=order, enforce\_stationarity=False, enforce\_invertibility=False)  sarimax\_res = sarimax\_model.fit(disp=False)  print(sarimax\_res.summary())  # Forecast  sarimax\_forecast = sarimax\_res.get\_forecast(steps=test\_size, exog=X\_test)  y\_pred\_sarimax = sarimax\_forecast.predicted\_mean  rmse\_sarimax = math.sqrt(mean\_squared\_error(y\_test, y\_pred\_sarimax))  print(f"SARIMAX RMSE: {rmse\_sarimax:.4f}")  # Normalize features. We'll scale X and y separately.  scaler\_X = StandardScaler()  scaler\_y = StandardScaler()  X\_all = pd.concat([X\_train, X\_test])  scaler\_X.fit(X\_all)  X\_train\_scaled = scaler\_X.transform(X\_train)  X\_test\_scaled = scaler\_X.transform(X\_test)  scaler\_y.fit(y\_train.values.reshape(-1,1))  y\_train\_scaled = scaler\_y.transform(y\_train.values.reshape(-1,1)).flatten()  # Create sequences: window\_length years -> predict next year yield  window = 3 # you can change this (number of past years used to predict next year)  def make\_sequences(X\_scaled, y\_scaled, window):  Xs, ys = [], []  for i in range(len(X\_scaled) - window):  Xs.append(X\_scaled[i:(i+window), :])  ys.append(y\_scaled[i+window])  return np.array(Xs), np.array(ys)  # Build sequences from the training portion only  X\_train\_seq, y\_train\_seq = make\_sequences(X\_train\_scaled, scaler\_y.transform(y\_train.values.reshape(-1,1)).flatten(), window)  # For test sequences we need to produce sequences that may include the end of train data.  # Construct a combined scaled X for sequence building  X\_combined\_scaled = np.vstack([X\_train\_scaled, X\_test\_scaled])  y\_combined\_scaled = np.vstack([scaler\_y.transform(y\_train.values.reshape(-1,1)), scaler\_y.transform(y\_test.values.reshape(-1,1))]).flatten()  X\_all\_seq, y\_all\_seq = make\_sequences(X\_combined\_scaled, y\_combined\_scaled, window)  # Find the start index of test sequences within X\_all\_seq  # The first test prediction corresponds to index = len(X\_train\_scaled) - window  start\_idx = len(X\_train\_scaled) - window  X\_test\_seq = X\_all\_seq[start\_idx : start\_idx + len(y\_test)]  y\_test\_seq = y\_all\_seq[start\_idx : start\_idx + len(y\_test)]  print("Train seq shape:", X\_train\_seq.shape, y\_train\_seq.shape)  print("Test seq shape:", X\_test\_seq.shape, y\_test\_seq.shape)  # Build and train LSTM  def build\_lstm(input\_shape, units=64, dropout=0.2):  model = Sequential()  model.add(LSTM(units, input\_shape=input\_shape))  model.add(Dropout(dropout))  model.add(Dense(1))  model.compile(optimizer='adam', loss='mse')  return model  input\_shape = (X\_train\_seq.shape[1], X\_train\_seq.shape[2])  lstm\_model = build\_lstm(input\_shape, units=64, dropout=0.2)  es = EarlyStopping(monitor='val\_loss', patience=20, restore\_best\_weights=True, verbose=0)  history = lstm\_model.fit(X\_train\_seq, y\_train\_seq,  epochs=300,  batch\_size=8,  validation\_split=0.1,  callbacks=[es],  verbose=0)  # Predict and inverse-scale  y\_pred\_lstm\_scaled = lstm\_model.predict(X\_test\_seq).flatten()  y\_pred\_lstm = scaler\_y.inverse\_transform(y\_pred\_lstm\_scaled.reshape(-1,1)).flatten()  rmse\_lstm = math.sqrt(mean\_squared\_error(y\_test.values, y\_pred\_lstm))  print(f"LSTM RMSE: {rmse\_lstm:.4f}")  # Build and train GRU (same approach)  def build\_gru(input\_shape, units=64, dropout=0.2):  model = Sequential()  model.add(GRU(units, input\_shape=input\_shape))  model.add(Dropout(dropout))  model.add(Dense(1))  model.compile(optimizer='adam', loss='mse')  return model  gru\_model = build\_gru(input\_shape, units=64, dropout=0.2)  es2 = EarlyStopping(monitor='val\_loss', patience=20, restore\_best\_weights=True, verbose=0)  history\_gru = gru\_model.fit(X\_train\_seq, y\_train\_seq,  epochs=300,  batch\_size=8,  validation\_split=0.1,  callbacks=[es2],  verbose=0)  y\_pred\_gru\_scaled = gru\_model.predict(X\_test\_seq).flatten()  y\_pred\_gru = scaler\_y.inverse\_transform(y\_pred\_gru\_scaled.reshape(-1,1)).flatten()  rmse\_gru = math.sqrt(mean\_squared\_error(y\_test.values, y\_pred\_gru))  print(f"GRU RMSE: {rmse\_gru:.4f}")  import numpy as np  # SARIMAX metrics  mse\_sarimax, mae\_sarimax, mape\_sarimax, rmse\_sarimax, rsq\_sarimax = calculate\_metrics(y\_test.values, y\_pred\_sarimax.values)  aic\_sarimax = sarimax\_res.aic  # LSTM metrics  mse\_lstm, mae\_lstm, mape\_lstm, rmse\_lstm, rsq\_lstm = calculate\_metrics(y\_test.values, y\_pred\_lstm)  n\_lstm = len(y\_test)  k\_lstm = lstm\_model.count\_params()  aic\_lstm = n\_lstm \* np.log(mse\_lstm) + 2 \* k\_lstm  # GRU metrics  mse\_gru, mae\_gru, mape\_gru, rmse\_gru, rsq\_gru = calculate\_metrics(y\_test.values, y\_pred\_gru)  n\_gru = len(y\_test)  k\_gru = gru\_model.count\_params()  aic\_gru = n\_gru \* np.log(mse\_gru) + 2 \* k\_gru  # Display results  print("=== Model Evaluation Metrics ===")  print(f"SARIMAX - MSE: {mse\_sarimax:.4f}, MAE: {mae\_sarimax:.4f}, MAPE: {mape\_sarimax:.2f}%, RMSE: {rmse\_sarimax:.4f}, AIC: {aic\_sarimax:.2f}")  print(f"LSTM - MSE: {mse\_lstm:.4f}, MAE: {mae\_lstm:.4f}, MAPE: {mape\_lstm:.2f}%, RMSE: {rmse\_lstm:.4f}, AIC: {aic\_lstm:.2f}")  print(f"GRU - MSE: {mse\_gru:.4f}, MAE: {mae\_gru:.4f}, MAPE: {mape\_gru:.2f}%, RMSE: {rmse\_gru:.4f}, AIC: {aic\_gru:.2f}")  # Data  models = ["SARIMAX", "LSTM", "GRU"]  metrics = ["MSE", "MAE", "MAPE", "RMSE"]  values = [  [6.0759, 2.2906, 29.81, 2.4649], # SARIMAX  [4.3069, 1.9171, 21.83, 2.0753], # LSTM  [3.8168, 1.7329, 19.52, 1.9537] # GRU  ]  # Bar settings  x = np.arange(len(models)) # positions for models  width = 0.2 # width of each metric bar  fig, ax = plt.subplots(figsize=(8, 6))  # Plot each metric as an offset bar  for i, metric in enumerate(metrics):  ax.bar(x + i \* width - (width \* (len(metrics)-1) / 2),  [v[i] for v in values],  width,  label=metric)  # Labels and title  ax.set\_ylabel('Metric Value')  ax.set\_xlabel('Models')  ax.set\_title('Model Evaluation Metrics')  ax.set\_xticks(x)  ax.set\_xticklabels(models)  ax.legend()  plt.tight\_layout()  plt.show() |

# APPENDIX III: GitHub Link and One Drive Link

**GitHub Link**

[**https://github.com/Tharunch9866/Dissertation**](https://github.com/Tharunch9866/Dissertation)

**One Drive Link**