A Model for Predicting Tea Yield for Enhanced Food Security in Kenya

Masai Joan Jemutai

Student ID : 101816

A Dissertation submitted in partial fulfillment of the requirements for the award of a Degree of Master of Science in Information Technology at Strathmore University

School of Computing and Engineering Sciences

Strathmore University

Nairobi, Kenya

March 2024

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the research proposal itself.

Signature……jmasai…… **Date**……March 29, 2024……...

Joan Jemutai Masai

Student ID: 101816

Approval

This proposal has been submitted to the Faculty of Information Technology for examination with my approval as the university supervisor.

Signature…… **Date**……29th March 2024……...

Dr. Victor Rop

Abstract

The increasing uncertainty caused by climate change and its effects on crop yields have made it essential to develop accurate predictive models for crop yield in Kenya. By accurately predicting crop yields, stakeholders can effectively plan and manage crop production, ensuring food security and preventing potential food emergencies. This study aims to address this need by utilizing artificial intelligence techniques to develop a predictive model specifically for tea crop yield.

The developed model leverages on machine learning algorithms to analyze historical data on tea yield, rainfall, temperature, soil water deficit, and hail damage. These variables are crucial factors influencing tea crop production in Kenya. By training the model with this data, it was able to make predictions about future tea crop yields. The performance and accuracy of the model was evaluated using the Root Mean Squared Error (RMSE) metric, which measures the differences between the predicted and actual values.

The outcomes of this research underscore the potential of artificial intelligence techniques in accurately predicting tea crop yield. Leveraging machine learning algorithms and historical data on crucial variables such as tea yield, rainfall, temperature, soil water deficit, and hail damage, the developed model shows promising predictive prowess. This research augments agricultural planning and management practices, bolstering food security and resilience amidst the uncertainties posed by climate change.

Key words: artificial intelligence, climate change, crop yield, food security, Kenya, machine learning, predictive modeling.

Table of Contents

List of Figures

List of Tables

Abbreviations and Symbols

AI	-	Artificial Intelligence
ANNs	-	Artificial Neural Networks
FAO	-	Food and Agriculture Organization
GDP	-	Gross Domestic Product
KES	-	Kenyan Shilling
KNN	-	K-Nearest Neighbor
LS-SVM	-	Least Squared Support Vector Machine
MAE	-	Mean Absolute Error
MAPE	-	Mean Absolute Percentage Error
ML	-	Machine Learning
R&D	-	Research and Development
RMSE	-	Root Mean Squared Error
SDGs	-	Sustainable Development Goals
SVM	-	Support Vector Machine
UN	-	United Nations

Definition of Terms

Artificial Intelligence (AI): AI is a branch of computer science that focuses on developing intelligent machines capable of performing tasks that typically require human intelligence, such as visual perception, decision-making, and problem-solving (Russell & Norvig, 2016).

Climate Change: Climate change refers to long-term shifts in weather patterns and average temperatures, primarily resulting from human activities, such as burning fossil fuels and deforestation, leading to changes in the Earth's climate system (IPCC, 2014).

Crop Yield: Crop yield refers to the measure of agricultural output or production per unit area of cultivated land, usually expressed as a weight or quantity of the harvested crop (FAO, 2016).

Food Security: Food security exists when all people, at all times, have physical, social, and economic access to sufficient, safe, and nutritious food that meets their dietary needs and preferences for an active and healthy life (FAO, 2003).

Greenleaf Tea: Greenleaf tea refers to the young, tender leaves and buds of the Camellia sinensis plant that are harvested for tea production before they undergo significant oxidation. Unlike black tea, which is fully oxidized, greenleaf tea is minimally processed and retains its natural green color. It is known for its delicate flavor, light aroma, and high levels of antioxidants. Greenleaf tea is widely consumed and valued for its potential health benefits, including boosting metabolism, aiding in weight loss, and improving cardiovascular health (Chacko et al., 2010).

Machine Learning: Machine learning is a subset of AI that involves the development of algorithms and models that enable computers to learn and make predictions or decisions based on patterns and data without explicit programming (Mitchell, 1997).

Machine Learning Algorithms: Machine learning algorithms are mathematical models and techniques that enable computers to learn from data, identify patterns, and make predictions or decisions. Examples include regression, decision trees, random forests, and support vector machines (Bishop, 2006).

Acknowledgments

I extend my deepest gratitude to Eastern Produce Kenya Ltd for their generous provision of the dataset that played a crucial role in the success of this study. The collaboration with the company significantly enriched the research, allowing for practical insights and a real-world application of the developed model.

My heartfelt thanks go to my supervisor, Dr. Victor Rop, for his unwavering support and guidance throughout this research journey. His expertise and constructive feedback have been invaluable, shaping the direction of this thesis and contributing to its successful completion. I am profoundly grateful for the time and effort he invested in mentoring me.

I am also indebted to the faculty members in the School of Computing and Engineering Sciences. Their support and guidance during my graduate studies have been instrumental in my academic growth. Special thanks to Dr. Butime and Dr. Omwenga for their encouragement to complete the thesis.

Lastly, I express my deepest appreciation to my family and friends for their unwavering support, encouragement, and love. Their faith in me has been a constant source of inspiration and motivation, propelling me to achieve my academic goals.

For Nyla, my shining light.

Chapter 1: Introduction

1.1 Background of the study

Kenya is a country heavily reliant on agriculture, with more than 75% of the population engaged in agricultural activities (FAO, 2021). According to the Food and Agriculture Organization (FAO), Kenya is experiencing a "serious food security crisis" due to a combination of factors, including drought, conflict, and economic challenges (FAO, 2021). The report indicates that over 2.8 million Kenyans are facing acute food insecurity, while an additional 3.5 million are experiencing chronic food insecurity. In addition, climate change and extreme weather events, such as droughts and floods, are having a significant impact on food security in Kenya (FAO, 2019). These events are leading to low agricultural productivity and limited food availability, resulting in food insecurity and malnutrition, particularly in rural areas.

To address these challenges, there is a need to develop innovative and sustainable solutions that can enhance agricultural productivity and ensure food security in Kenya. One promising approach is the use of artificial intelligence (AI) in agriculture, which can provide accurate predictions of crop yields, improve decision-making for farmers, and guide policymakers in formulating effective policies.

Previous research has shown the potential of AI in agriculture, including predicting crop yield, monitoring crop health, optimizing irrigation, and improving pest management (Kamilaris & Prenafeta-Boldú, 2018). However, there is need for more research on the use of AI in agriculture in Kenya. Therefore, this study aims to develop an AI model for predicting tea yield in Kenya to enhance food security. The process involves collecting the relevant data including historical crop yields, selecting relevant features, and building a predictive model using machine learning algorithms. The study provides insights into the potential of AI in enhancing food security in Kenya and serves as a guide for researchers and practitioners working in the agricultural sector.

1.2 Problem Statement

Previous research on crop yield prediction has mainly focused on statistical and traditional machine learning techniques, which have shown limitations in accurately predicting crop yield due to their inability to account for the complex and dynamic nature of the agricultural system (Kamilaris et al., 2018; Zhao et al., 2020). These techniques often fail to capture the intricate relationships between various factors influencing crop yield, such as weather patterns, soil

characteristics, and agricultural practices. Moreover, some studies use small sample sizes, which may not accurately represent the population and lead to biased predictions. In addition, most of the existing models have been developed and tested in other countries, and their performance may not be directly applicable to the Kenyan context due to differences in climate, soil, and agricultural practices. Kenya has a unique agricultural landscape characterized by diverse climates, soil types, and crop varieties. Therefore, there is a critical need to develop a more robust and accurate model for predicting crop yield in Kenya that takes into account the unique characteristics of the local agricultural system.

Furthermore, traditional approaches to crop yield prediction often lack scalability and efficiency, making it challenging to handle large volumes of data and provide timely insights to farmers and policymakers. With the increasing availability of data from various sources such as remote sensing, weather stations, and farm management systems, there is an opportunity to leverage advanced data analytics techniques, such as machine learning and artificial intelligence, to develop more accurate and scalable models for crop yield prediction in Kenya. Therefore, the problem statement revolves around the need to address the limitations of existing crop yield prediction models by developing a more robust, accurate, and scalable model specifically tailored to the Kenyan agricultural context. This model should leverage advanced data analytics techniques, incorporate diverse data sources, and consider the unique characteristics of the local agricultural system to provide timely and accurate predictions of crop yields.

1.3 Aim

The aim of this research is to develop a model for predicting tea yield for enhanced food security in Kenya.

1.4 Research Objectives

The research objectives are:

i. To analyze the existing approaches used for crop yield prediction.
ii. To review the current approaches used in forecasting crop yield.
iii. To develop a model for predicting tea yield.
iv. To validate the developed model.

1.5 Research questions

The research questions that will guide the study are:

i. What are the current methods used to predict crop yield?
ii. What are the challenges facing current approaches to crop yield forecasting?
iii. How will the model for tea yield prediction be developed?
iv. How can the model performance be validated?

1.6 Justification

According to FAO (2021), agriculture is the mainstay of Kenya's economy, providing employment and livelihoods for over 75% of the population, particularly in rural areas. However, the agricultural sector in Kenya faces numerous challenges, including climate change, soil degradation, pests and diseases, and limited access to finance and markets. These challenges have contributed to low agricultural productivity, limited food availability, and food insecurity. The use of artificial intelligence (AI) in agriculture has the potential to address some of these challenges by providing accurate predictions of crop yields and guiding decision-making for farmers and policymakers. Predicting crop yield, especially in tea production in Kenya, is crucial for mitigating the impacts of climate change and ensuring food security. Accurate predictions can help in formulating effective measures for crop production management, enabling farmers and industries to plan their activities efficiently (Kihoro, 2019).

Crop yield prediction using AI can support farmers and policymakers in making informed decisions related to agricultural practices, resource allocation, and crop management strategies. The study developed a predictive model using artificial intelligence and machine learning algorithms to provide valuable insights and improve food security strategies in Kenya. Tea production is a significant contributor to Kenya's economy and livelihoods. Accurate predictions of tea crop yield can lead to better production planning, increased food security, and reduced risks of food emergencies in the region (Kinyua et al., 2021). The study's findings will contribute to the existing knowledge in crop yield prediction using artificial intelligence and machine learning techniques. The research will provide context-specific results and recommendations tailored to the challenges and opportunities in tea production in Kenya.

The outcomes of this study can have practical implications for tea farmers, industries, and policymakers, aiding in optimizing crop production, improving food security, and boosting Kenya's economy. This research aligns with the global effort to address food security

challenges in the face of climate change and supports the sustainable development of the agriculture sector in Kenya.

1.7 Scope of the study

The objective of this study was to develop an artificial intelligence (AI) model for predicting crop yield in Kenya, with a specific focus on the tea crop. The study utilized relevant data on soil water deficit, rainfall, temperature, hail damage figures, and historical tea yields from the period of 2012 to 2022. Data collection and preprocessing were conducted using data from Eastern Produce Kenya Limited (EPK), located in Nandi County, Kenya. Machine learning algorithms were employed for feature selection, model development, and performance evaluation. The performance evaluation of the model was based on RMSE and R-squared (R2) score. It is important to note that the study's scope was limited to the tea crop in Kenya, which is a significant contributor to the country's economy and a major source of employment. The findings of this study serve as a proof of concept for the application of AI in modeling and predicting tea yield, providing valuable insights for future research and practical applications in the tea industry. Additionally, the results can inform decision-making processes for tea farmers and policymakers, contributing to sustainable tea production and enhancing food security in Kenya.

However, it is essential to consider that the study's findings and recommendations may not be directly applicable to other crops or regions without further investigation and consideration of contextual factors. Furthermore, since the dataset used to train the model was from Nandi County, the findings and recommendations may not be directly applicable to other tea-growing regions without further investigations.

1.8 Significance of the study

The significance of this study lies in its potential to enhance food security in Kenya by developing an artificial intelligence (AI) framework for modeling and predicting crop yield. The use of AI in agriculture has the potential to increase crop yield, improve resource efficiency, and reduce the impact of climate change on crop production. The study's focus on tea, an important cash crop in Kenya, and the use of publicly available data sources increase the study's relevance and potential impact on the agricultural sector in Kenya. The use of AI in modeling and predicting tea yield can provide accurate predictions of tea yields, guide decision-making for tea farmers and policymakers, and contribute to sustainable tea

production, ultimately benefiting the country's economy, society, and environment. Additionally, the study can provide insights into the impact of climate change on tea yield and guide the development of strategies to mitigate the impact of climate change on tea production. The study's findings and recommendations can also inform decision-making for farmers, policymakers, and other stakeholders, contributing to sustainable crop production and food security in Kenya. Moreover, the study's approach to developing an AI framework for crop yield prediction can be adapted and extended to other crops and regions, contributing to global efforts to enhance sustainable agriculture and food security. Overall, the study's significance lies in its potential to guide future research and applications of AI in agriculture, ultimately contributing to sustainable and resilient food systems in Kenya and beyond.

Chapter 2: Literature Review

2.1 Introduction

Crop yield prediction involves estimating the amount of crop production from a specific land area by considering various factors such as weather conditions, soil type, crop variety, and management practices. This estimation is valuable for farmers, agricultural researchers, and policymakers as it enables them to plan and make informed decisions regarding crop management, market prices, and food security (Kamilaris & Kartakoullis, 2017). This chapter provides a comprehensive overview of the existing literature on crop yield prediction, focusing on both traditional approaches and recent advancements utilizing artificial intelligence and machine learning techniques. The literature review is followed by a summary and discussion of the contributions made by this thesis.

2.2 Theoretical Literature

Theoretical literature in the field of crop yield prediction has explored various concepts and models that form the foundation for understanding and developing predictive models. One notable theoretical framework is the use of machine learning algorithms, such as artificial neural networks (ANNs), in crop yield prediction. ANNs are inspired by the structure and function of the human brain and have been widely used in different domains, including agriculture (Kamilaris & Prenafeta-Boldú, 2018). These algorithms have the ability to learn from data, identify patterns, and make predictions based on the learned patterns. The concept of data-driven modeling has gained prominence in crop yield prediction. Data-driven models rely on large-scale datasets to uncover hidden patterns and relationships between variables. These models, such as random forests and support vector machines, have shown promising results in accurately predicting crop yields based on diverse input data (Sahoo et al., 2021). Furthermore, spatial analysis and Geographic Information Systems (GIS) have been integrated into crop yield prediction models. These frameworks enable the incorporation of spatially explicit data, such as land cover, elevation, and soil types, to enhance the accuracy of predictions (Xiao, Zhang, & Wang, 2020).

Theoretical literature in crop yield prediction provides a conceptual understanding of different models, algorithms, and approaches that can be employed to predict crop yields. These

frameworks serve as a guide for researchers and practitioners in developing and implementing effective prediction models to support decision-making in agriculture.

2.2.1 Traditional Approaches for Crop Yield Prediction

The traditional methods employed in Kenya for predicting crop yield primarily involve statistical approaches such as regression analysis and time-series analysis. These methods rely on historical data related to weather, soil, and crop production to develop predictive models. However, these traditional models face certain limitations. One of the key challenges is the inability to effectively account for the impact of climate change and the scarcity of accurate and timely data. According to Mureithi et al. (2020), the traditional approach to crop yield prediction in Kenya is hindered by insufficient and unreliable data concerning weather and soil conditions, which are crucial factors in accurate yield prediction. Outdated and unreliable data makes it challenging to develop reliable predictive models. Furthermore, these traditional models fail to consider the significant influence of climate change, posing a significant threat to crop production in Kenya. Chepkwony et al. (2021) also highlighted the limitations of the traditional approach, emphasizing the lack of accurate and up-to-date data on weather and soil conditions. The authors proposed that the integration of remote sensing technology and machine learning algorithms could enhance the accuracy of crop yield prediction by providing real-time data on weather and soil conditions. Therefore, it is evident that the traditional approach to crop yield prediction in Kenya encounters limitations related to data availability and the consideration of climate change. To address these challenges and improve the accuracy and reliability of crop yield prediction, novel approaches incorporating advanced technologies such as remote sensing and machine learning need to be explored, contributing to enhanced food security in the country.

2.2.2 Models for Crop Yield Prediction

Crop yield prediction is a critical aspect of agricultural decision-making, and various models have been developed to forecast crop yields based on environmental factors, management practices, and crop genetics. These models play a crucial role in improving agricultural productivity, optimizing resource utilization, and mitigating risks associated with crop production. One of the prominent models used for crop yield prediction is the Agricultural Production Systems Simulator (APSIM). APSIM is a comprehensive modeling framework that simulates various aspects of crop growth, development, and yield under different environmental and management scenarios. It incorporates detailed representations of soil

processes, plant physiology, and crop management practices to accurately predict crop yields (Holzworth et al., 2014).

Another widely utilized model in crop yield prediction is the Decision Support System for Agrotechnology Transfer (DSSAT). DSSAT comprises a suite of crop simulation models that simulate the growth, development, and yield of a wide range of crops under diverse environmental conditions and management practices. These models integrate empirical relationships and physiological processes to capture the complex interactions between crops and their environment (Jones et al., 2017). In recent years, machine learning techniques have gained popularity in crop yield prediction due to their ability to handle large datasets and capture nonlinear relationships between input variables and crop yields. Support Vector Machine (SVM) regression, Random Forests, and Gradient Boosting Machines are among the machine learning algorithms that have been applied to crop yield prediction with promising results (Rasheed et al., 2019; Khaled et al., 2020). Ensemble modeling approaches, which combine multiple models to improve prediction accuracy, have emerged as effective tools for crop yield prediction. Ensemble methods such as model averaging, stacking, and bagging have been used to integrate the strengths of different models and mitigate their individual weaknesses, leading to more robust predictions (Montesinos-López et al., 2021). Overall, the development and application of advanced modeling techniques in crop yield prediction have the potential to revolutionize agricultural decision-making and contribute to sustainable food production.

2.2.3 Crop Yield Prediction Using Remote Sensing

Remote sensing techniques have emerged as a valuable approach for crop yield prediction, offering timely and precise information on crop health and growth. These techniques involve the utilization of satellite and airborne sensors to capture data on various crop characteristics such as chlorophyll content, leaf area index, and biomass. Advanced algorithms are then employed to process this data and estimate crop yield. Several studies have demonstrated the effectiveness of remote sensing techniques in accurately predicting crop yield for various crops, including maize, wheat, and rice (Gao et al., 2018; Liu et al., 2020; Wang et al., 2018). One of the key advantages of remote sensing techniques is their non-destructive nature, enabling rapid coverage of large agricultural areas. This makes them particularly suitable for regional or national crop monitoring. However, the use of remote sensing techniques for crop yield prediction also presents certain limitations. These include the high cost associated with

acquiring and processing satellite imagery, the requirement for specialized expertise to interpret the data, and the constraints in spatial and temporal resolution of the imagery (Apan et al., 2017; Torres-Rua et al., 2020). Although remote sensing techniques hold great promise in enhancing crop yield prediction and improving food security, further research is necessary to address these limitations and develop more accurate and cost-effective methods.

2.2.4 Crop Yield Prediction Using Machine Learning

Crop yield prediction using machine learning techniques involves the utilization of various algorithms to forecast crop yields based on historical data, weather conditions, soil information, and other relevant factors. Machine learning algorithms offer the capability to construct predictive models that can accurately estimate crop yields. Random forests, support vector machines, artificial neural networks, and deep learning models are among the commonly employed machine learning algorithms for crop yield prediction. For instance, Huang et al. (2018) developed a machine learning-based model to predict maize yield in China by incorporating climate, soil, and management data, resulting in high prediction accuracy. Similarly, Moshou et al. (2019) utilized machine learning algorithms to forecast wheat yield in Europe, combining remote sensing data with field measurements to achieve accurate field-level predictions. Mureithi et al. (2020) developed a machine learning model for maize yield prediction in Kenya, employing weather, soil, and crop management data, and achieving high prediction accuracy. Chepkwony et al. (2021) also employed machine learning techniques to predict tea yield in Kenya, utilizing weather, soil, and management data to achieve accurate predictions. These studies underscore the potential of machine learning in crop yield prediction, highlighting its significance in enhancing food security and agricultural productivity.

2.3 Empirical Literature

Empirical literature on crop yield prediction has provided valuable insights into the effectiveness of various models and algorithms in accurately forecasting crop yields. Zhang, Wei, and Zhang (2019) conducted a study using a deep learning model to predict maize yield based on meteorological and remote sensing data. Their results demonstrated the potential of deep learning in achieving high prediction accuracy. Gao, Chen, Huang, and Zhang (2020) employed a random forest model to predict rice yield using weather and soil data. Their findings highlighted the effectiveness of the random forest algorithm in accurately predicting rice yields. Another study by Karthikeya, Sudarshan, and Shetty (2020) focused on crop yield prediction using the K-Nearest Neighbor algorithm. Their research showcased the potential of

K-Nearest Neighbor in accurately predicting crop yields. Dhiman, Singh, and Prakash (2021) utilized the support vector machine algorithm to predict rice yield in India based on climate variables. Their study demonstrated the effectiveness of the support vector machine in accurately predicting rice yields.

2.3.1 Machine Learning Algorithms for Crop Yield Prediction

Machine learning, as defined by Goodfellow, Bengio, and Courville (2016), is a branch of computer science that focuses on developing algorithms and statistical models to enable computer systems to improve their performance on a given task by learning from data without explicit programming. It involves employing a range of statistical and computational techniques to identify patterns and relationships within data, which can then be utilized to make predictions or informed decisions. In this study, several machine learning algorithms can be utilized, and they are described in detail below.

2.3.2 Neural Networks for Crop Yield Prediction

Artificial neural networks (ANNs) are machine learning algorithms inspired by the structure and function of the human brain. They consist of interconnected nodes, or "neurons," organized in layers, which process and transmit information through weighted connections. ANNs can be applied to tasks such as classification, regression, and prediction. The learning process in ANNs involves adjusting the connections between neurons based on examples or training data, enabling them to learn patterns and make predictions (Siganos & Stergiou, 1996). Several studies have demonstrated the effectiveness of ANNs in predicting crop yield (Kamilaris & Prenafeta-Boldú, 2018; Sahoo et al., 2021). These networks have been successfully applied to analyze and predict various factors influencing crop yield, such as weather conditions, soil properties, and management practices. Figure 2.1 provides an illustration of an artificial neural network architecture, highlighting its interconnected nodes and layered structure.

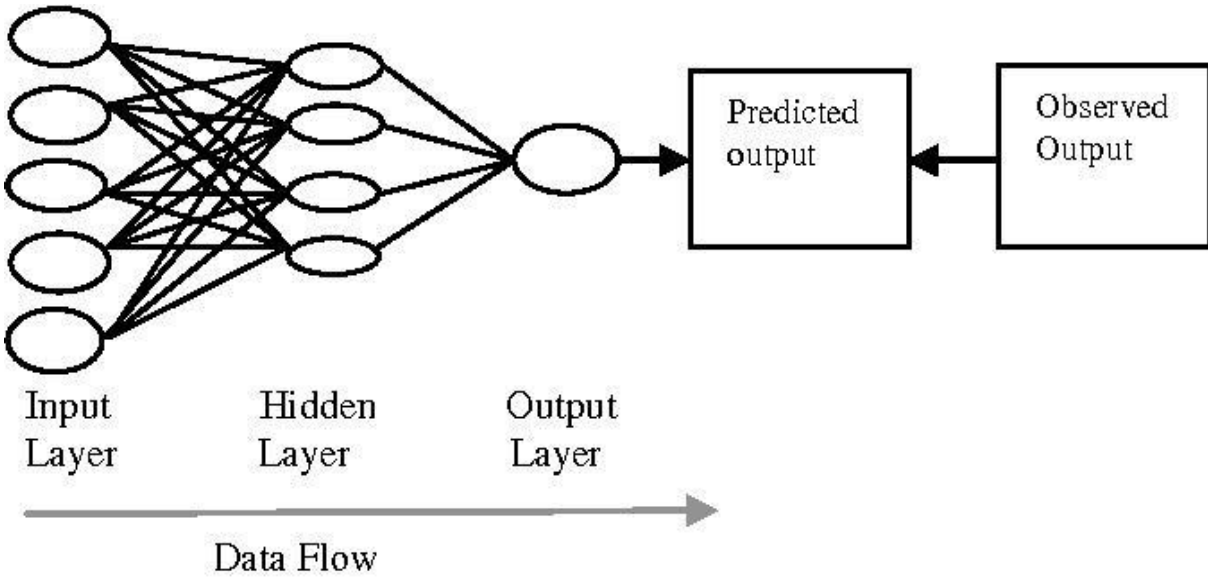


Figure 2.1: An artificial neural network (Siganos & Stergiou, 1996)

Zhang et al. (2019) conducted a study where they developed an artificial neural network model to predict maize yield using meteorological and remote sensing data. The model demonstrated high prediction accuracy, achieving a coefficient of determination of 0.93. Similarly, Gao et al. (2020) employed an artificial neural network model to predict rice yield by utilizing weather and soil data. The model accurately predicted rice yield, with a mean absolute error of 0.044 ton/ha. In a study by Manjula and Narsimha (2016), an Optimal Neural Network (ONN) classifier was utilized to predict crop yield in India through spatial data mining, which involves extracting valuable information from spatial databases. The prediction process involved three main stages: preprocessing, feature reduction, and prediction. Multilinear Principal Component Analysis (MPCA) was employed during the feature reduction phase. The accuracy of the prediction model was evaluated based on the prediction error and accuracy value. The authors concluded that the implementation of technology is crucial in enhancing food production and agriculture.

2.3.2 Convolutional Neural Networks for Crop Yield Prediction

Convolutional Neural Networks (CNNs) are a potent category of artificial neural networks specifically tailored for tasks involving image recognition and classification. They possess the capability to automatically extract distinctive features from raw data by employing filters that identify patterns within the data. Through a sliding window mechanism, these filters generate feature maps that highlight specific aspects of the input. CNNs consist of multiple layers,

including convolutional layers, pooling layers, and fully connected layers, enabling the network to learn increasingly intricate representations of the input data. CNNs have exhibited remarkable performance across various domains, including image recognition, natural language processing, and speech recognition (LeCun et al., 2015).

Researchers have explored the application of CNNs for crop yield prediction with promising results. For instance, Patel et al. (2021) developed a CNN model using satellite images and climate data for crop yield prediction. They employed the VGG16 transfer learning technique and trained the model on a dataset comprising satellite images and climate data collected from diverse locations in India. The CNN model exhibited superior performance compared to traditional machine learning models, achieving an accuracy of 89.68%. The study concluded that CNNs hold significant potential for crop yield prediction and have the capacity to enhance food security in agricultural regions. Similarly, Russello (2018) utilized 3-dimensional CNNs to predict crop yield by employing satellite images. Their CNN model outperformed alternative machine learning methods in crop yield prediction. Additionally, You et al. (2017) employed deep learning techniques, including CNNs and recurrent neural networks, to forecast soybean yield in the United States. They utilized a sequence of remotely sensed images captured before harvest. You et al. (2017) proposed an innovative approach that integrated a Gaussian process to address spatiotemporal errors and compared the performance of their proposed method with regression, decision tree, and Long Short-Term Memory (LSTM) baseline techniques. Their model outperformed traditional remote-sensing based methods by 15% in terms of Mean Absolute Percentage Error (MAPE). Moreover, the Gaussian-based CNN method showcased a 30% reduction in Root Mean Squared Error (RMSE) compared to other methods. Table 2.1 presents the RMSE values of the different methods employed in the study.

Table 2.1: RMSE Using Various Methods (You et al., 2017)

	Baselines			Deep Models			
Year	Ridge	Tree	DNN	LSTM	LSTM +GP	CNN	CNN +GP
2011	9.00	7.98	9.97	5.83	5.77	5.76	5.70
2012	6.95	7.40	7.58	6.22	6.23	5.91	5.68
2013	7.31	8.13	9.20	6.39	5.96	5.50	5.83
2014	8.46	7.50	7.66	6.42	5.70	5.27	4.89
2015	8.10	7.64	7.19	6.47	5.49	6.40	5.67
Avg	7.98	7.73	8.32	6.27	5.83	5.77	**5.55**

2.3.3 Deep Learning Approaches for Crop Yield Prediction

In their study, Kuwata and Shibasaki (2016) explored the application of deep learning techniques for corn yield estimation in the United States. They developed a deep neural network with six hidden layers, each consisting of 4000 neurons, to predict county-level corn yield across the entire country. The input data for the model included annual county-level corn yield data, satellite-based vegetative index data obtained from MODIS EVI (Enhanced Vegetation Index), and gridded weather patterns for the country. The results demonstrated that the deep neural network outperformed other methods in accurately estimating county-level corn yield nationwide. Additionally, a comparative analysis was conducted with support vector machine and autoencoder algorithms to determine the algorithm that provided the highest level of accuracy. To enhance the quality of the MODIS EVI data, a wavelength shrinkage method was employed to smooth the data, reducing data redundancy and removing noise. This data preprocessing technique involved using equations (2.1) and (2.2) for data compression and signal denoising, respectively.

$$W f(x) = \int_{-\infty}^{+\infty} f(x) \frac{1}{\sqrt{a}} \psi\left(\frac{a-b}{a}\right) dx \quad (2.1)$$

where ψ = a mother wavelet function
a = a scaling parameter
b = a shifting parameter

$$\lambda = \sigma\sqrt{2 \log n} \quad (2.2)$$

where λ = a threshold
σ = variance of noise
n = sample number of the signals

The smoothed data obtained for the entire country is visualized in Figure 2.2.

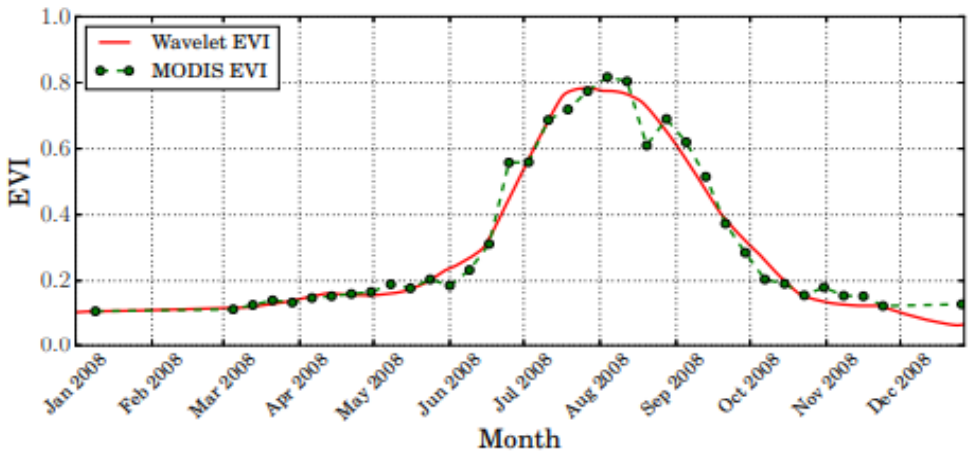


Figure 2.2: MODIS EVI and EVI smoothed using the wavelet transform method (Kuwataa & Shibasaki, 2016)

The evaluation of the estimation results for the three selected algorithms was based on the root mean square error (RMSE) and the coefficient of determination (R^2). The values for RMSE and R^2 are presented in Table 2.2 below.

Table 2.2: The estimation results of the three models (Kuwataa & Shibasaki, 2016)

	Daily input dataset		5-day accumulation dataset	
	RMSE	R^2	RMSE	R^2
SVM	20.4	0.727	17.7	0.792
DNN (six hidden layers)	18.5	0.773	18.2	0.780
Autoencoder	19.0	0.759	21.3	0.700

Based on the findings, it was observed that SVM exhibited higher accuracy when using the 5-day accumulation dataset compared to the daily input set. Conversely, the autoencoder algorithm demonstrated superior accuracy with the daily input dataset compared to the 5-day dataset. The DNN algorithm, which effectively extracted features from multidimensional input data, yielded more accurate results than SVM with the daily input dataset. Consequently, the DNN algorithm was recommended as the most suitable approach for crop yield prediction.

2.3.4 Random Forest Algorithm for Crop Yield Prediction

Jeong et al. (2016) conducted a comprehensive evaluation of the Random Forests (RF) machine-learning method for predicting crop yield responses to climate and biophysical variables. The study focused on wheat, maize, and potato crops at global and regional scales, using multiple linear regressions (MLR) as a benchmark for comparison. The researchers utilized crop yield data from various sources and regions, including global wheat grain yield, maize grain yield from US counties over thirty years, and potato tuber and maize silage yield from the northeastern seaboard region.

The performance of RF and MLR was assessed using several evaluation metrics, including root mean square error (RMSE), EF, and d. The results demonstrated that the RF algorithm

exhibited high predictive capabilities and consistently outperformed the MLR benchmarks across all performance statistics. Detailed comparisons of the performance metrics can be found in Table 2.3.

Table 2.3: Random Forests (RF) and multiple linear regression (MLR model performance evaluation statistics (Jeong et al., 2016)

Crop	**Scale**	**RF**			**MLR**		
		RMSE (tons/ha)	**EF**	***d***	**RMSE (tons/ha)**	**EF**	***d***
Wheat	Global	0.32	0.96	0.99	1.32	0.31	0.68
Maize (grain)	U.S.	1.13	0.76	0.92	1.93	0.30	0.67
Potato	NESR	2.77	0.75	0.95	5.62	-0.87	0.73
Maize (silage)	NESR	1.90	0.85	0.97	4.54	-0.41	0.75

Figure 2.3 illustrates the performance of the Random Forest (RF) model on test datasets. The plots depict the observed versus predicted values for the four case studies: (A) global wheat grain yield, (B) maize grain yield in the United States over a span of 30 years, (C) wet tuber yield of potatoes in the northeastern seaboard region (NESR), and (D) maize silage yield in NESR. In each plot, the dashed lines represent the 1:1 relation, while the solid line represents the linear regression between the observed values and the predictions made by the RF model on the test datasets.

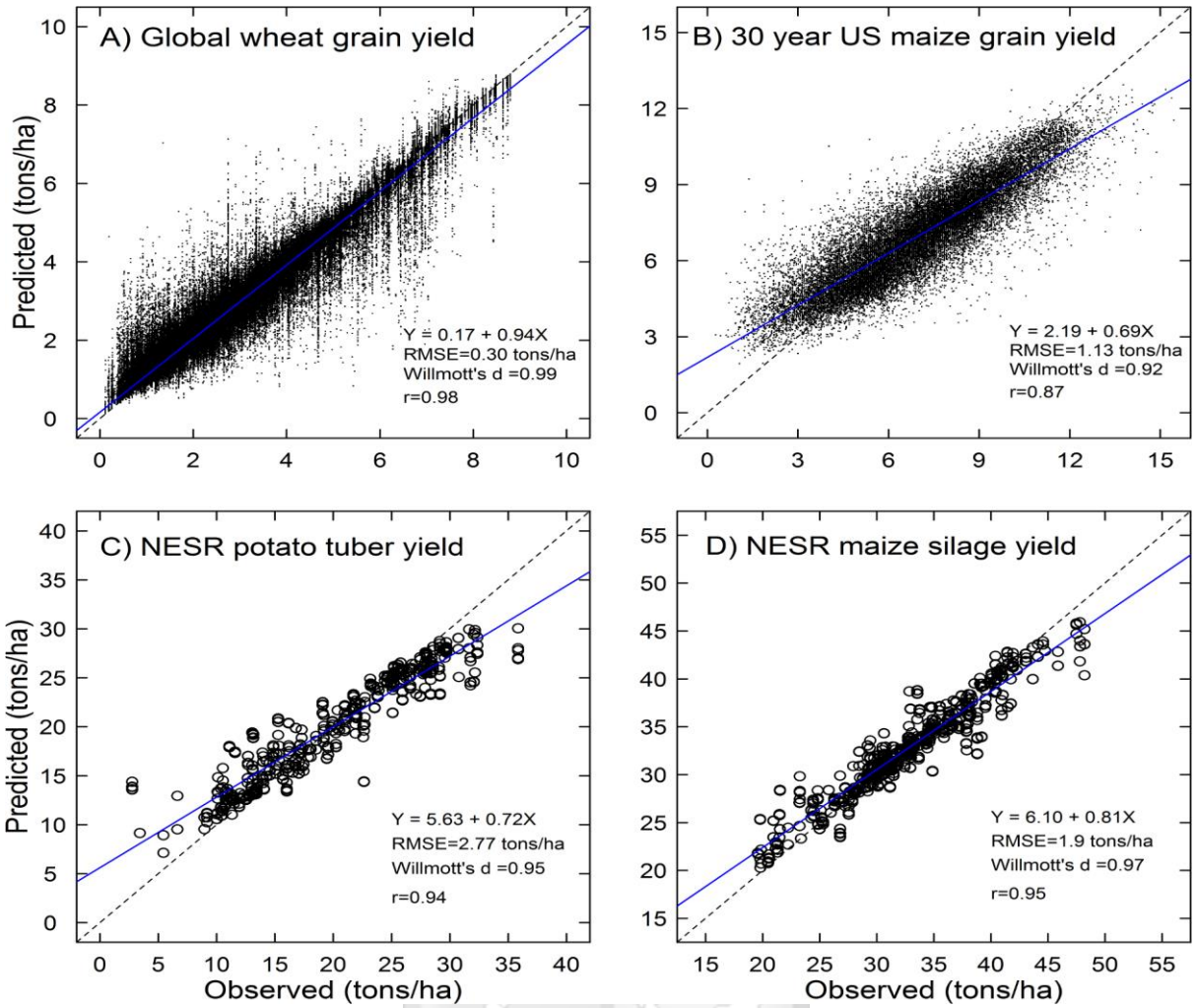


Figure 2.3: Random Forests model performance for test datasets (Jeong et al., 2016)

Jeong et al. (2016) highlights the potential of utilizing the Random Forest (RF) algorithm as a viable statistical modeling approach for crop yield prediction. However, the authors emphasize the importance of being cautious with RF to avoid overfitting the data, particularly when the training data is concentrated. They also note that the accuracy of RF may diminish when the training data is sparse. Moreover, it is advised to refrain from extrapolating beyond the boundaries of the training data when employing RF regression to ensure reliable predictions.

2.3.5 K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) is a versatile algorithm used for both classification and regression tasks. This algorithm involves determining the value of parameter k, which represents the number of nearest neighbors to consider. When classifying a new data point, KNN identifies its k-nearest neighbors in the training data by calculating the distance between the input variables and all data points in the dataset. Various distance measures, such as Euclidean

distance, Minkowski distance, and Mahalanobis distance, can be used for this purpose. It is worth noting that a larger value of k generally leads to better classification accuracy (Harrison, 2018; Brownlee, 2016).

In a study conducted by Karthikeya, Sudarshan, and Shetty (2020), an efficient crop harvesting system was developed by utilizing agricultural datasets from multiple portals in India. These datasets were organized in a structured manner, and the KNN algorithm was employed to achieve accurate predictions of crop yield. Another study by Pavani and Augusta (2022) focused on predicting sorghum crop yield in India, utilizing the K-nearest neighbor and support vector machine algorithms. The authors implemented these algorithms in MATLAB and evaluated their performance. The dataset included parameters such as soil moisture, humidity, temperature, and rainfall. The results indicated that the support vector machine model exhibited higher accuracy in predicting sorghum crop yield compared to the K-nearest neighbor algorithm.

2.3.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a versatile algorithm that can be applied to both regression and classification tasks. The main objective of the SVM algorithm is to identify a hyperplane in an N-dimensional space, where N represents the number of features, that effectively separates and classifies the data points. Figure 2.4 provides a visual representation of the potential hyperplanes that can be derived in a dimensional space (Gandhi, 2018).

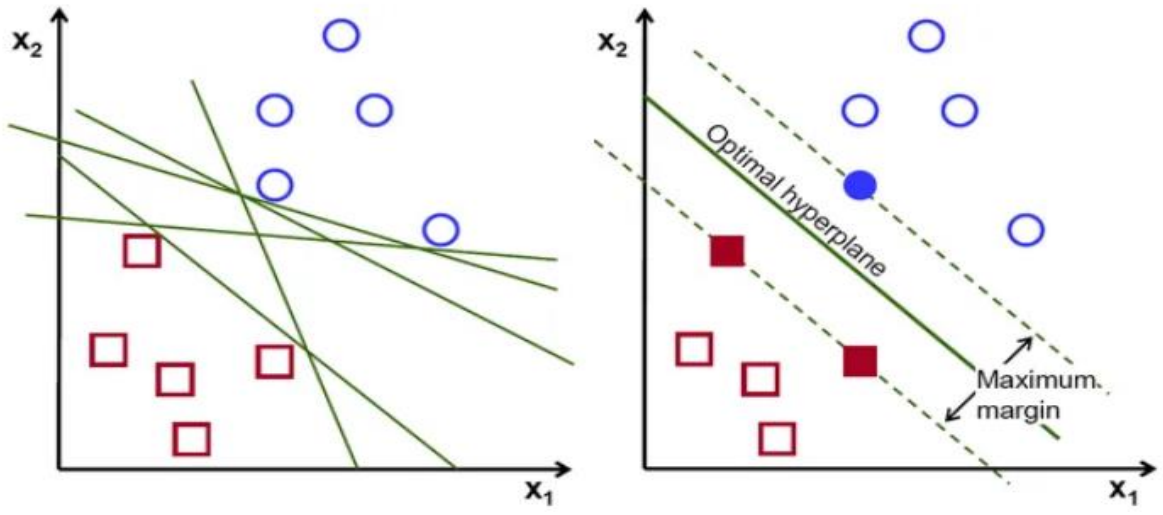


Figure 2.4: Possible SVM hyperplanes (Gandhi, 2018)

Dhiman et al. (2021) employed Support Vector Machine (SVM) to forecast rice crop yield in India, utilizing climatic variables such as temperature, rainfall, and humidity. The study findings indicated that SVM surpassed alternative machine learning algorithms like artificial neural networks and decision trees in terms of accuracy and robustness. Similarly, Zheng et al. (2019) applied SVM to predict corn yield in the United States, considering historical weather data and crop management practices. The results exhibited a high level of accuracy in corn yield prediction using SVM, highlighting its potential as a valuable tool for crop management and decision-making (Dhiman et al., 2021; Zheng et al., 2019).

2.3.7 Comparative Analysis of Different Learning Approaches

Kumar, Kumar, and Vats (2018) conducted a comparative study to identify an efficient crop yield estimation technique for sugarcane production using descriptive analytics. They utilized three datasets: soil, rainfall, and yield data, and evaluated the performance of three supervised learning techniques: K-Nearest Neighbor, Support Vector Machine (SVM), and Least Squared Support Vector Machine (LS-SVM). The authors assessed the accuracy of each training model and analyzed the error rates. They also discussed the advantages of LS-SVM, an extension of SVM that addresses classification and regression problems in complex and large datasets. LS-SVM is particularly suitable for large datasets as it solves problems in linear time by working with intervals or ranges. The classifiers were evaluated through cross-validation, Mean Squared Error (MSE), and Average Mean Squared Error (AMSE). The evaluation results are presented in Table 2.4, Figure 2.5, and Figure 2.6, which display the performance metrics and the Average Mean Squared Error of All Classifiers, respectively (Kumar et al., 2018).

Table 2.4: Validation Error of Classifiers at different Cross validation runs (Kumar, Kumar & Vats, 2018)

Cross Validation	**KNN**	**SVM**	**LS-SVM**
1	0.307428	0.147108	0.0319429
2	0.33877	0.109105	0.0273692
3	0.342234	0.124982	0.0258275
4	0.32295	0.110263	0.0234979
5	0.32094	0.116354	0.0374257
6	0.323151	0.141801	0.0406318
7	0.313512	0.111455	0.0498942
8	0.331188	0.148668	0.0371499
9	0.344364	0.113936	0.0484682
10	0.304701	0.135623	0.0271545

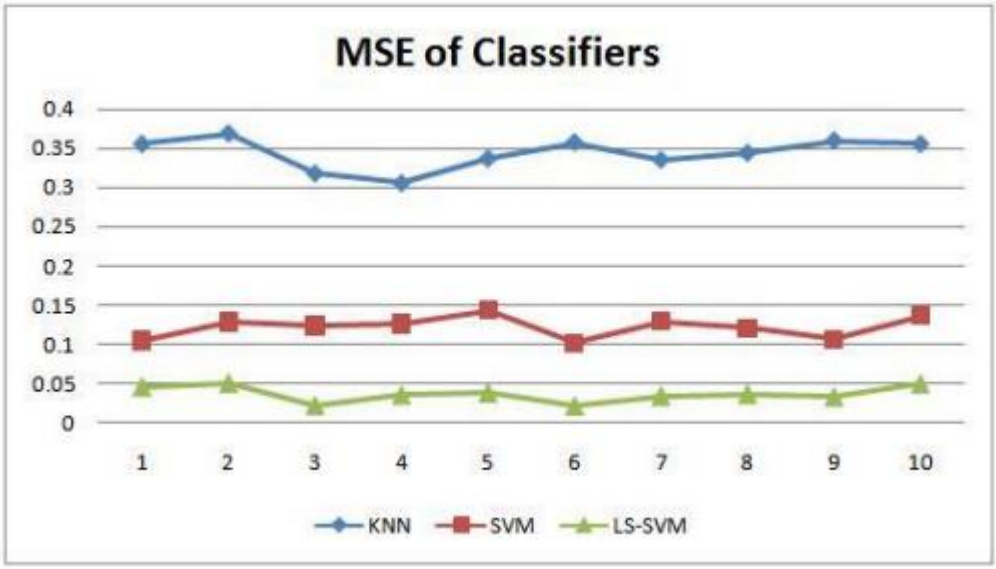


Figure 2.5: Mean Squared Error of all classifiers for Crop Yield Estimation (Kumar, Kumar & Vats, 2018)

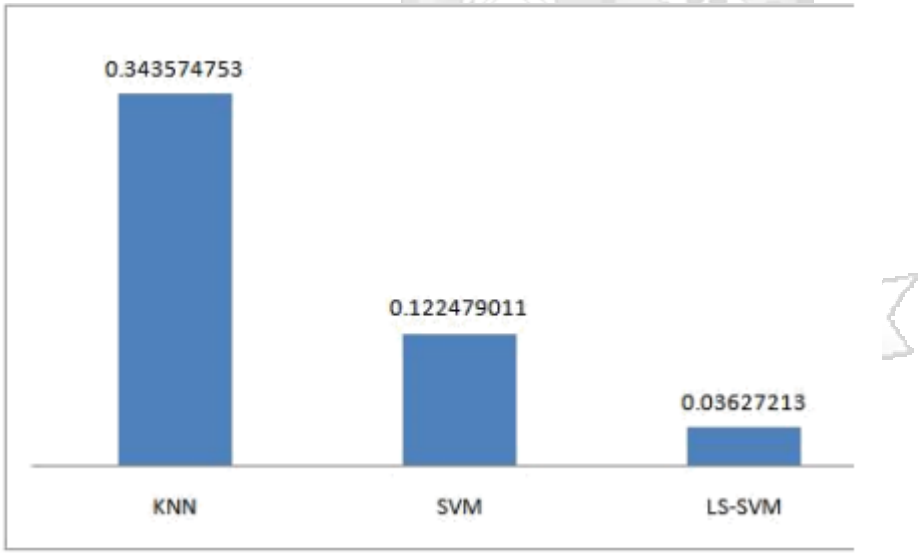


Figure 2.6: Average Mean Squared Error of all classifiers for Crop Yield Estimation (Kumar, Kumar & Vats, 2018)

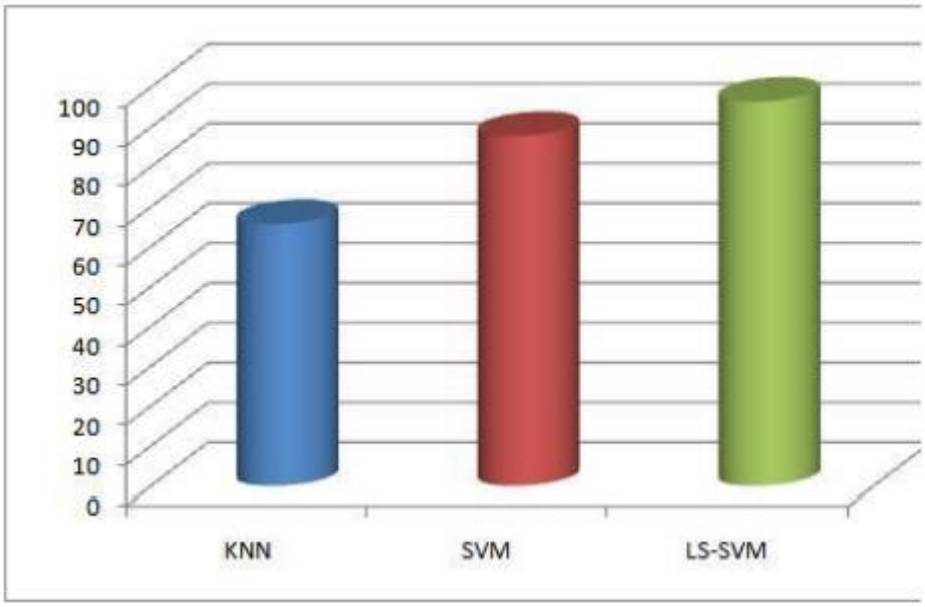


Figure 2.7: Average Accuracy of all classifiers for Crop Yield Estimation, LS-SVM Showing the Highest (Kumar, Kumar & Vats, 2018)

Compared to KNN and SVM, the LS-SVM classifier exhibited higher accuracy and lower error rate (Kumar, Kumar, & Vats, 2018).

2.4 Contribution of this study

The primary objective of this study is to leverage artificial intelligence and machine learning techniques to predict tea crop yield in Kenya. By incorporating environmental data and relevant variables, the study aims to develop a predictive model that can accurately forecast tea crop yields. The utilization of artificial intelligence in crop yield prediction has the potential to enhance decision-making processes and improve productivity in the tea industry.

2.5 Conceptual Framework

The conceptual framework for this study encompasses various key components that guide the research process. The study began by defining specific research objectives, including the prediction of tea crop yield, identification of influential factors, and the evaluation of predictive models. To establish a strong theoretical foundation, the study engaged in an extensive literature review, examining existing knowledge on crop yield prediction. Data collection played a critical role in the study, with a focus on collecting data on tea yield and associated variables from tea estates. The raw data was obtained from Eastern Produce Kenya Limited. The collected data underwent preprocessing to ensure data quality, including addressing

missing values, removing outliers, and preparing the dataset for analysis. Feature selection was another crucial step, involving the identification of the most significant variables that influenced tea yield. Development of the predictive model was a central aspect of the study. Calibration and validation of the developed model were carried out using historical data and independent datasets, respectively, to assess accuracy and reliability. To evaluate the performance of the developed model, appropriate evaluation metrics such as Mean Squared Error (MSE) and R-Squared (R2) were employed. Lastly, the conceptual framework acknowledged the importance of future research directions, through identification of key areas that warranted further investigation for the advancement and refinement of tea yield prediction model. The conceptual framework used in this study is depicted in Figure 2.8.

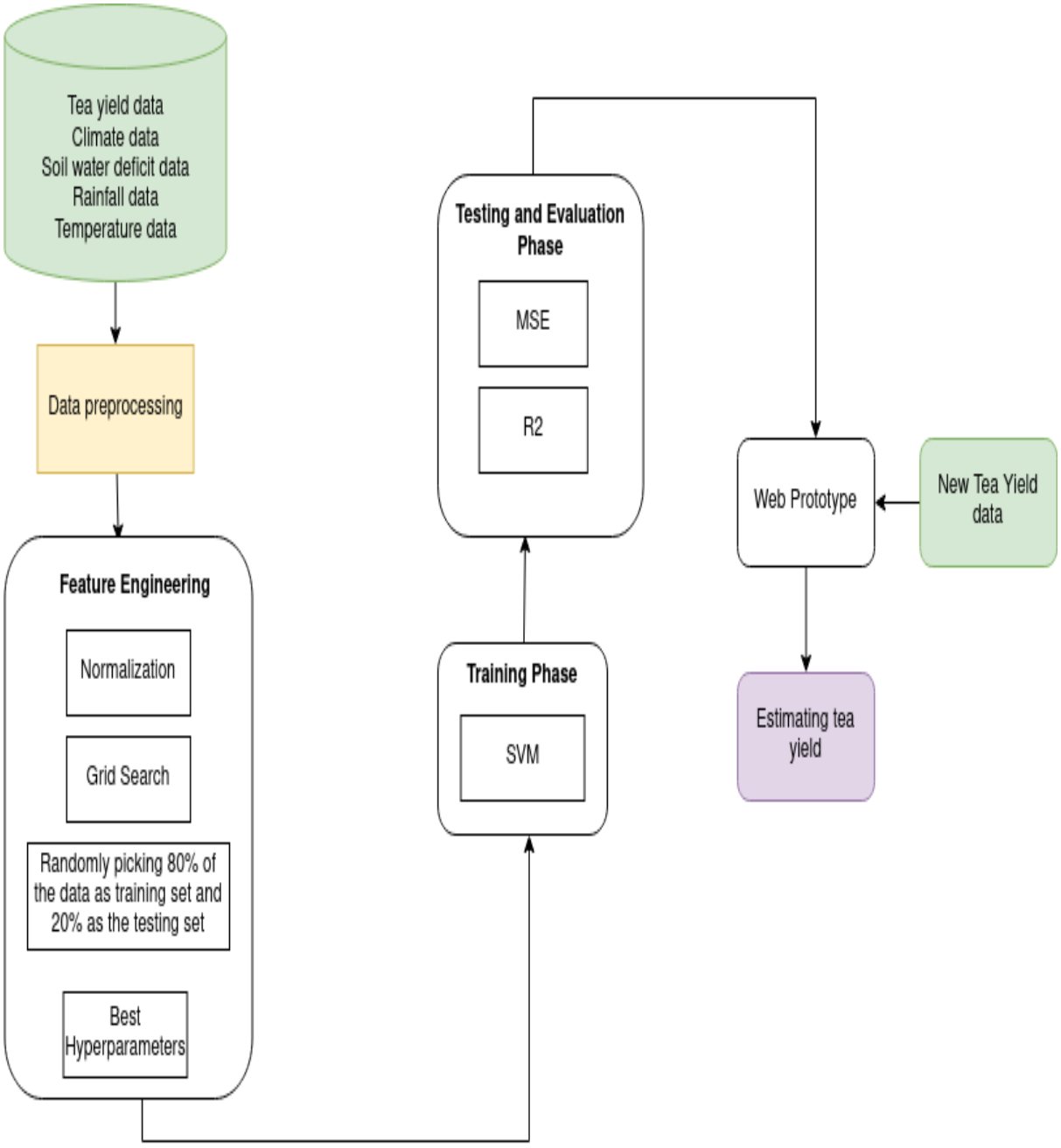


Figure 2.8: The Conceptual Framework

Chapter 3: Research Methodology

3.1 Introduction

This chapter provides details of the approach employed to fulfill the research objectives. The methods used for data collection, sampling, and data analysis are detailed, along with the selected system development methodology for constructing the crop yield prediction model.

3.2 The Research Design

Research design plays a crucial role in shaping the methodology, procedures, and strategies employed to address research questions. Creswell (2018) categorizes research design into four types: quantitative, qualitative, experimental, and mixed methods. Quantitative research design focuses on numerical data collection and analysis, while qualitative research design emphasizes non-numerical data. Experimental research design involves manipulating variables to observe their impact on the outcome of interest while controlling for other factors. Mixed methods research design integrates both numerical and non-numerical data analysis, preprocessing and exploration. The study objectives were accomplished by conducting a comprehensive literature review on crop prediction utilizing artificial intelligence and machine learning algorithms, while acknowledging their limitations. The developed model was subjected to testing and validation using the Mean Squared Error (MSE) and R-Squared (R2) metrics.

3.3 Research Data

The study utilized historical data sources to gather the necessary information. Greenleaf tea yield data in kilograms per month from 2012 to 2022 was obtained from Eastern Produce Kenya Limited. Additionally, data on rainfall, temperature, and soil water deficit for the same time period was collected. Hail damage figures, which could have a detrimental effect on the yield of greenleaf tea, were also provided. It is important to note that hail damage occurs occasionally and impacts the productivity of the greenleaf tea crop. Eastern Produce Kenya Limited is a prominent agricultural company headquartered in Kenya. Specializing in the cultivation and processing of tea, the company primarily focuses on greenleaf tea production. With a substantial presence in Kenya's tea industry, it plays a pivotal role in the country's tea sector.

3.3.1 Sampling Design

To facilitate model development and evaluation, the dataset was automatically and randomly divided into two sets: the training set and the test set. The training set was used to train the model, enabling it to learn patterns and relationships within the data. Meanwhile, the test set was used to assess the performance and accuracy of the trained model.

3.3.2 Data Preprocessing

Data preprocessing is a crucial step in the development of any predictive model. In this study, the collected raw data underwent thorough preprocessing to ensure its quality and suitability for analysis. This involved several key procedures aimed at addressing common issues such as missing values, duplicates, and errors.

3.3.3 Feature Engineering

In this study, feature engineering played an essential role in transforming the obtained secondary data into relevant features that improved the prediction power and accuracy of the model. This involved reviwing the features to ensure that they represent the research problem. The main features that were used to develop the tea yield prediction model included rainfall, temperature, hail damage, soil water deficit, and tea yield.

3.3.4 Data Analysis

Data analysis in the study involved examining and interpreting the collected data to derive meaningful insights and draw conclusions. The collected data underwent a thorough data cleaning and preprocessing phase. This involved handling missing values, addressing outliers, and ensuring data consistency. Data normalization or standardization techniques were also applied to ensure comparability and facilitate analysis. The analysis also involved applying appropriate statistical tests or machine learning algorithms to examine the relationships and dependencies between the predictor variables (such as rainfall, temperature, hail damage, and soil water deficit) and the target variable (tea yield). Regression analysis was employed to assess the strength and significance of these relationships.

3.4 System Development Methodology

The study utilized an iterative and incremental system development methodology to train the model continuously until it achieved a satisfactory level of performance. This approach provides more flexibility in making changes to the model compared to the gated-step approach.

3.4.1 Phases of Iterative and Incremental Development

Iterative and incremental development is a software development approach that emphasizes adaptability and flexibility throughout the development process. The iterative development model, as depicted in Figure 3.1, involves breaking down the development process into smaller iterations, each adding new features or functionality to the product. At the completion of each iteration, rigorous testing, review, and evaluation take place before progressing to the next iteration. This approach enables developers to respond to changes promptly and identify and resolve issues early on. The iterative and incremental approach promotes greater agility and allows for refinement and enhancement of the product until it meets the desired specifications and quality standards (Kochar, 2021).

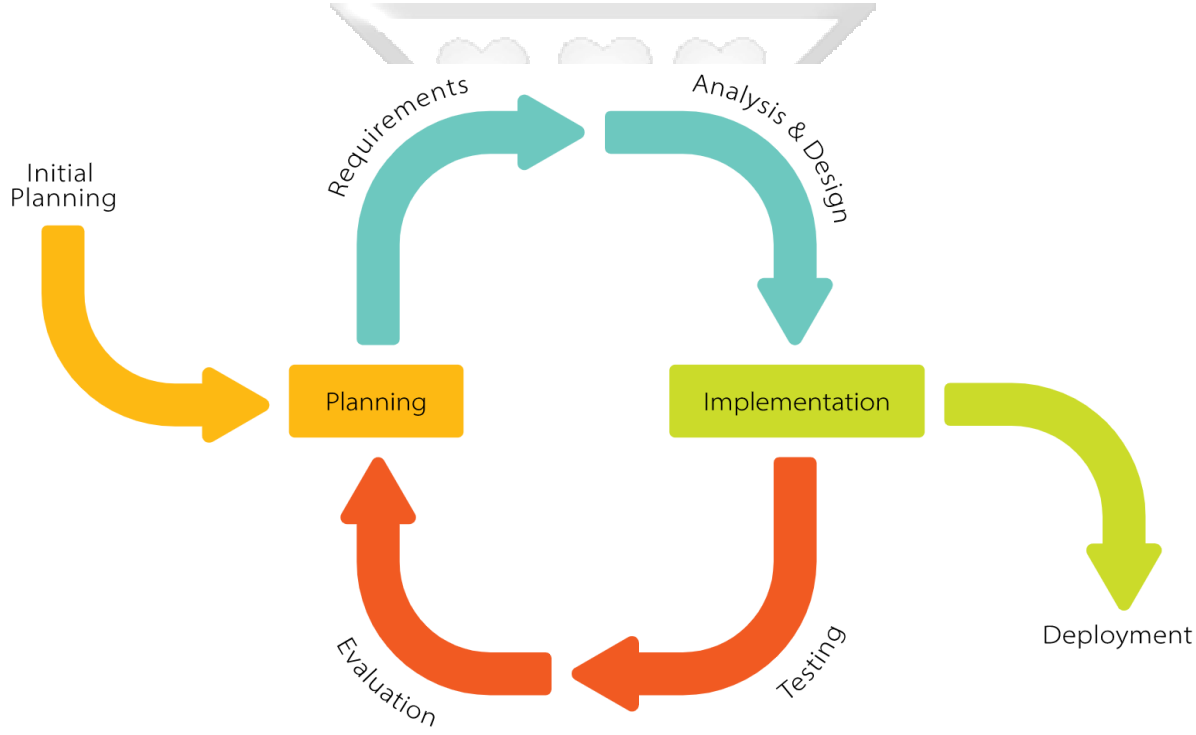


Figure 3.1: The Iterative and Incremental Development (Kochar, 2021)

3.4.2 The Planning Phase

The planning phase is the initial stage of a project that involves identifying objectives, defining requirements, allocating resources, and establishing timelines. Data sources were identified and extracted using extraction tools, and the data was cleaned to ensure consistency and reduce errors. Additionally, the tools to be used in building the model were identified during this phase.

3.4.3 Analysis and design phase

During this phase, the study conducted exploratory data analysis to gain insights into the data and identify the essential features necessary to achieve the research objectives. Python was the primary tool used for statistical analysis and visualization due to its popularity in data science and machine learning, its ease of use, and the extensive libraries it offers for scientific computing, data analysis, and visualization. Furthermore, to model the proposed solution, the study utilized use case diagrams, partial domain diagrams, and sequence diagrams.

3.4.4 Implementation

Python was utilized as the primary programming language in this study, along with libraries such as numpy, pandas, and matplotlib.pyplot to provide the necessary functions for data processing, visualization, and analysis. Support Vector Machine (SVM) machine learning algorithm was employed to train the data. The Scikit-Learn library was utilized, which offers built-in classes for various SVM algorithms. The training set was fitted to the SVM classifier.

3.4.5 Testing

Once the data preprocessing and cleaning procedures were completed, the performance of the developed model was assessed using the testing dataset obtained from the data split. The dataset used encompassed data from the years 2012 to 2022. The model split the data into two categories. Twenty percent of the dataset was used to test the model, while the remaining 80% was used for model training.

3.5 Research Quality

Research quality encompasses the accuracy, reliability, and validity of the data, methods, and findings in a study (Maxwell, 2013). It is crucial to ensure research quality as it helps minimize bias and enhances the credibility and generalizability of the study's outcomes. Boaz and Ashby (2003) identified several dimensions of research, including quality, transparency in reporting, technical execution, quality of signal, and fitness for the purpose of the research approach. These dimensions play a vital role in ensuring that the research can be effectively evaluated and utilized by others, providing valuable insights for policy and practice.

Once the training process was completed, the performance of the developed model was assessed using the test dataset. The Mean Squared Error (MSE) was employed as a metric to evaluate the model. MSE is a widely-used measure in regression problems, quantifying the

differences between the predicted values generated by a model and the observed values. This evaluation metric is well-suited as the square root function allows for the display of significant deviations in the numerical values. The formula for calculating MSE is as follows:

$$RMSE = \sqrt{\frac{\Sigma_{i=1}^{N} (x_i - \widehat{x_i})^2}{N}} \quad (3.1)$$

where i = variable i
N = number of non-missing data points
x_1 = actual observations time series
$\hat{x}_1$ = estimated time series

3.6 Ethical Considerations

Ethical considerations were taken into account in this study. The data utilized was sourced from Eastern Produce Kenya Limited (EPK). The data collected was solely used for the purpose of this study and was not shared with any third-party entities, to ensure data confidentiality and privacy.

3.7 Study Dissemination

The research findings will be published in the repositories of Strathmore University, where the study was conducted. By making the findings accessible through the university's repositories, other researchers, academics, and students will have the opportunity to access and refer to the research outcomes. Additionally, the study aimed to share the findings with the technical team at Eastern Produce Kenya Ltd., the tea estates from which the research data was obtained. This direct sharing of results with the industry professionals involved in tea production will allow them to gain insights into the study outcomes and potentially incorporate them into their practices and decision-making processes.

3.8 Research Risk Analysis

This study incorporates a comprehensive risk analysis to identify and address potential risks that may arise during the course of the research. Some risks were recognized and were managed effectively to ensure the validity and reliability of the study's findings. One of the key risks was the potential for data quality issues. It was acknowledged that the collected data may contain errors, inconsistencies, or missing values, which could impact the accuracy and reliability of

the study's results. To mitigate this risk, data cleaning and preprocessing techniques was implemented. These techniques included handling missing data, addressing outliers, and ensuring data consistency to enhance the overall quality of the dataset.

Another risk lied in the performance of the predictive model developed. There was a possibility that the model may not perform optimally, leading to inaccurate predictions. This risk could arise from factors such as inadequate feature selection, overfitting, or limitations of the chosen machine learning algorithm. To mitigate this risk, two evaluation metrics were employed to assess the model's performance. The risk of generalizability is also recognized in this study. The findings and conclusions drawn from the research may not be readily applicable to other regions or crops due to the specific focus on tea production in Nandi County in Kenya. The unique environmental and agricultural conditions of the study area may limit the generalizability of the developed model. To address this risk, the study acknowledged its limitations and provided recommendations for further research and considerations for broader applicability. This helped ensure that the study's outcomes were appropriately contextualized and that future research could build upon this foundation. Additionally, there were external factors beyond the control of the researcher that may have impacted tea crop yield. These factors included unpredictable weather events, changes in agricultural practices, or economic fluctuations. While it was impossible to completely mitigate these external influences, the study aimed to incorporate relevant historical data and explored potential correlations to account for such factors. This helped to provide a more comprehensive understanding of the dynamics between tea yield and external influences.

Chapter 4: System Analysis and Design

4.1 Introduction

This chapter explores the systematic analysis and design process for developing a robust tea yield prediction system. The goal is to identify the key requirements and specifications essential for building an effective tea yield predictive model. By understanding the underlying needs and constraints, the system was designed to meet functional and non-functional considerations while ensuring scalability, reliability, and usability.

4.2 System Requirements

The system requirements are essential features and characteristics necessary for the successful implementation of the tea yield prediction system. These requirements encompass both functional and non-functional aspects, ensuring that the system meets the required needs while maintaining performance and reliability.

4.2.1 Functional Requirements

Functional requirements define the specific functionalities and operations that the tea yield prediction system must perform to achieve its objectives. These requirements focus on the system's capabilities in terms of data processing, analysis, and prediction. The functional requirements included:

i. Data Input: The system allows users to enter various types of data, including temperature, rainfall, soil water deficit and hail damage figures.
ii. Prediction Generation: The system generates predictions for future tea yields based on input data provided by users or obtained from external sources.

4.2.2 Non-functional Requirements

Non-functional requirements define the quality attributes and constraints that govern the system's behavior and performance. These requirements address aspects such as usability, reliability, scalability, and security. Key non-functional requirements include:

i. Usability: The system features a user-friendly interface with intuitive data input forms, and clear explanations of predictions to facilitate user interaction and interpretation.

ii. Reliability: It provides accurate and reliable predictions consistently under varying environmental conditions and input scenarios, with minimal downtime or errors.

iii. Scalability: The system is scalable, capable of handling large volumes of data and accommodating increased user demand without sacrificing performance or responsiveness.

iv. Security: The system adheres to security best practices to safeguard sensitive data, prevent unauthorized access, and ensure data privacy and integrity throughout the prediction process.

4.3 System Architecture

A system architecture serves a visual representation that illustrates both the structure and behavior of the various components within a system. Figure 4.1 displays the system architecture of the tea yield prediction model, showcasing the interaction among different components to accomplish system functionality. Initially, raw data undergoes pre-processing to produce training and testing datasets. The model is then trained and validated, and converted into a format that can be embedded in a web application. The web interface receives queries from users then performs a prediction on the given inputs and sends the predicted yield result back to the web application. Finally, the user receives the yield prediction on the application.

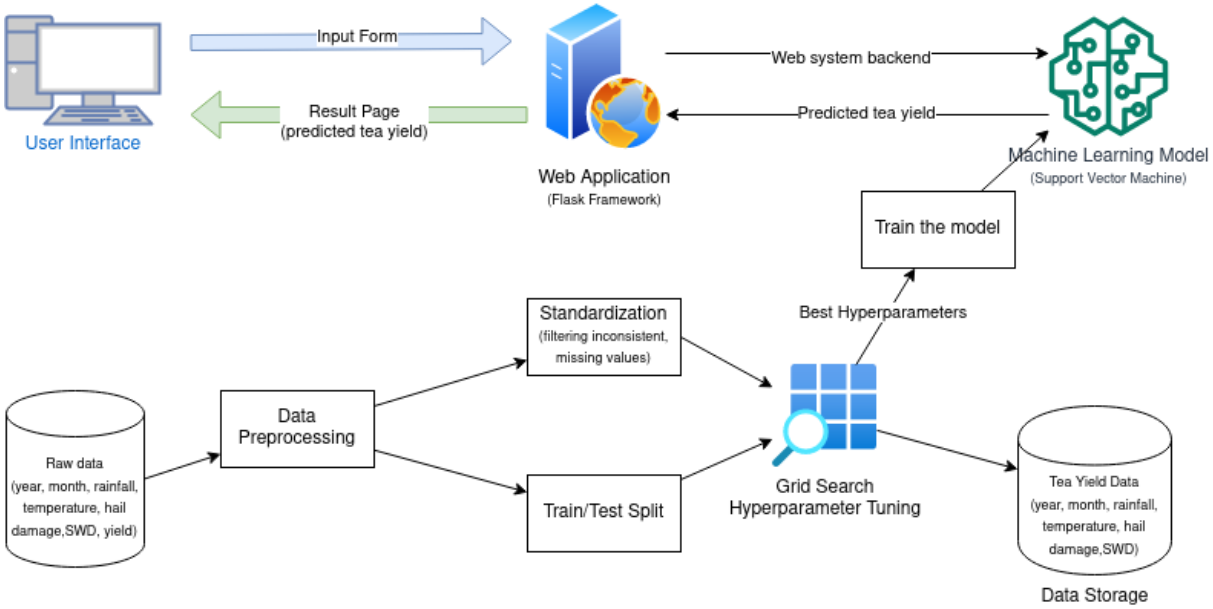


Figure 4.1: System Architecture

4.4 Use Case Diagram

A use case diagram is a visual representation that depicts the functional requirements of a system, illustrating the interactions between actors (users) and the system itself. It serves to identify the various use cases or functionalities of the system and how actors engage with them. Figure 4.2 presents the use case diagram for the system under consideration. The primary users of the system are farmers and agricultural researchers. Both the farmer and researcher have the capability to input specific data related to tea yield, such as the year, rainfall, and fertilizer usage. Additionally, the farmer can provide environmental data, including weather conditions, soil water deficit, and temperature. Both the farmer and researcher are able to view the predicted tea yield based on the inputted data. The agricultural researcher further conducts an analysis of historical tea yield data to discover patterns and trends. As a result of this analysis, the researcher may refine and adjust the prediction model. Lastly, the system administrator is responsible for managing user accounts, which entails tasks such as registration, login, and account settings.

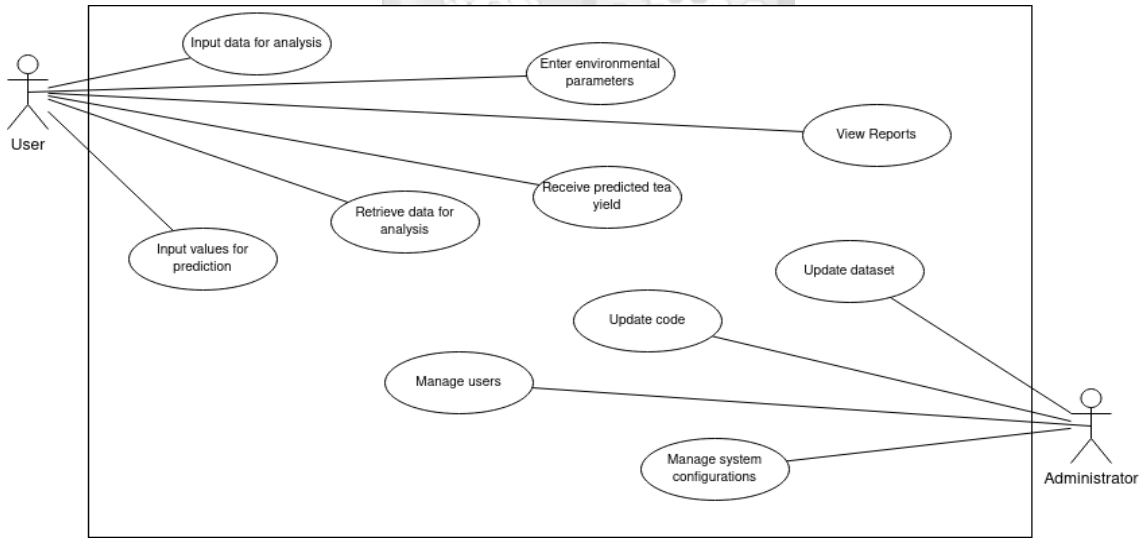


Figure 4.2: Use Case Diagram

4.5 Sequence Diagram

A sequence diagram is a visual representation that showcases the interactions between objects or components in a system over a specific period of time. Figure 4.3 below illustrates the flow of messages and the order of events between different classes in the tea yield prediction model.

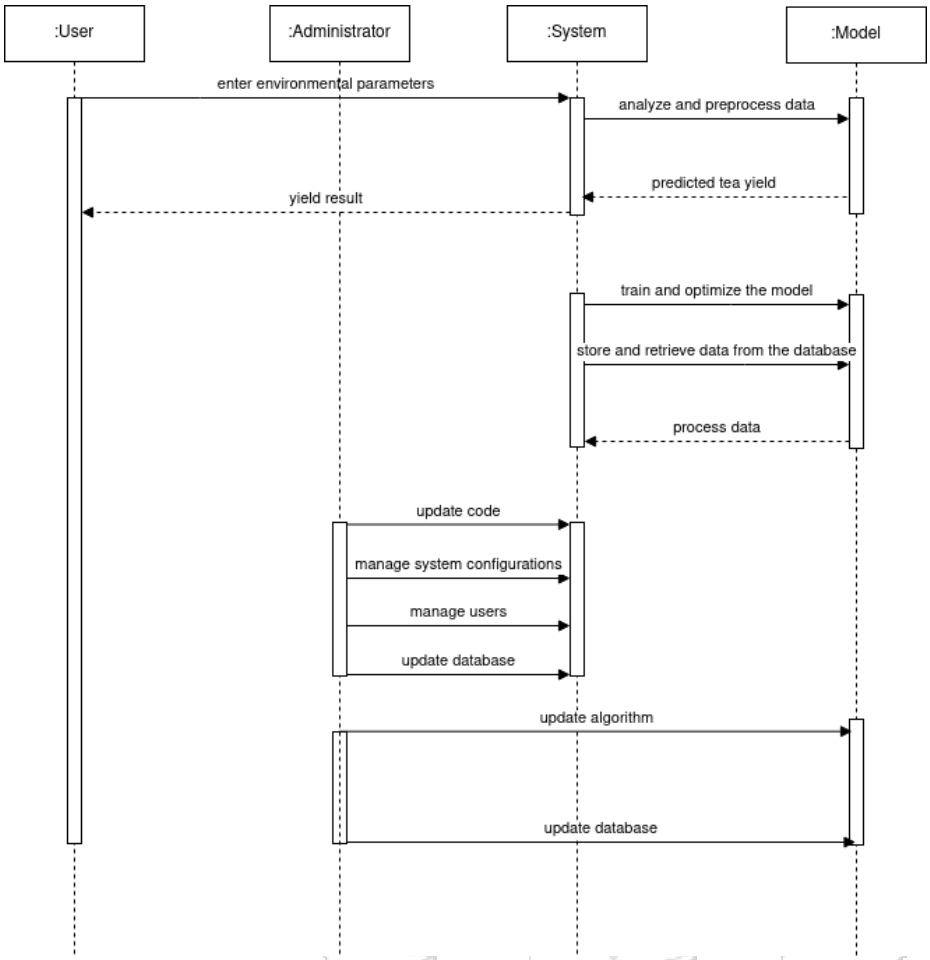


Figure 4.3: Sequence Diagram

4.6 Design Class Diagram

A design class diagram is a visual representation of the static structure of a system, showing the classes, their attributes, and the relationships between them. It provides an overview of the key classes and their associations, helping to understand the organization and structure of the system's components. The class diagram used to model the tea prediction model is as shown in Figure 4.4. This diagram helps in designing the structure of the system and establishing the foundation for implementing the functionality.

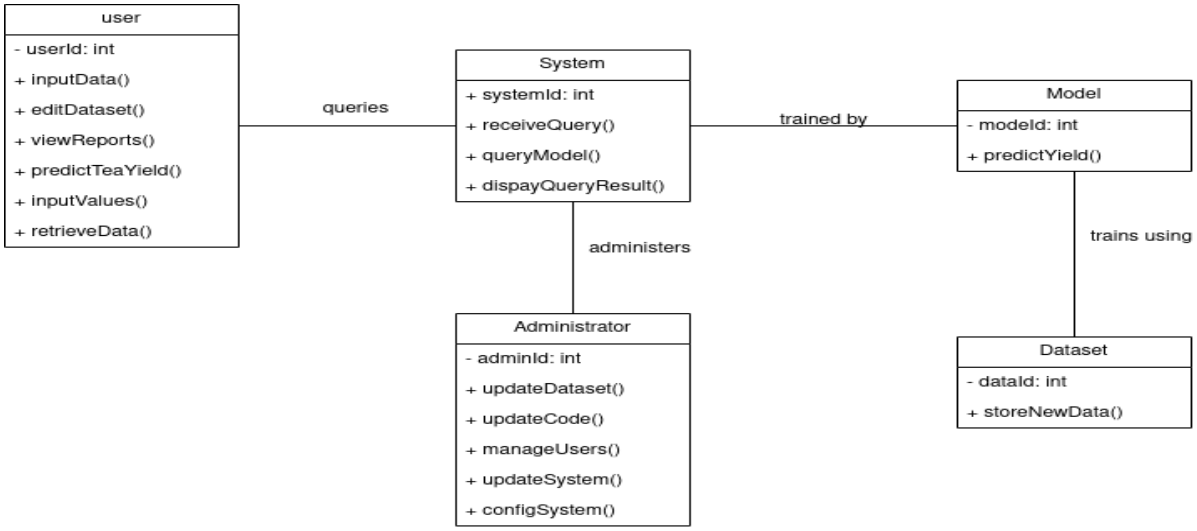


Figure 4.4: Design Class Diagram

4.7 Domain Model

A domain model is a simplified depiction of a specific domain or subdomain within a broader system, highlighting the fundamental entities, relationships, and attributes that pertain to that particular domain. In this study, a partial domain model has been utilized to represent the key concepts of the tea yield prediction system, illustrating the roles and regulations associated with the involved data. Figure 4.5 presents the domain model of the tea yield prediction model, offering a visual representation of the essential components and their interconnections.

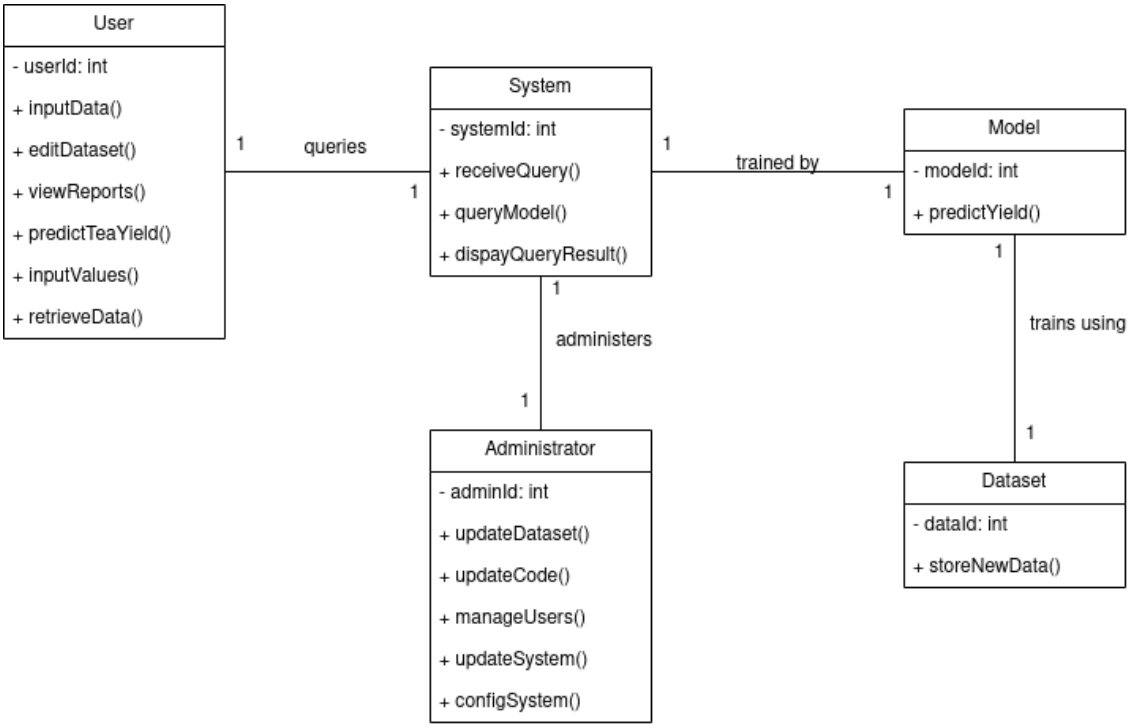


Figure 4.5: Partial Domain Model

Chapter 5: System Implementation and Testing

5.1 Introduction

In this chapter, focuses on the implementation and testing of the tea yield prediction model. The specific methods employed to develop the model, including the identification of inputs and their respective sources will be discussed. An overview of the algorithm used to build the model and highlight the results obtained from training the data will be provided.

5.2 Hardware and Software Environment

The implementation of the prediction model was carried out using the Python programming language. The development process was greatly facilitated by leveraging essential libraries such as NumPy, Pandas (Python data analysis), and Scikit-learn. These libraries provided robust tools for data manipulation, analysis, and the implementation of machine learning algorithms.

The environment used for this implementation evolved during the course of the project. Initially, the model development took place on a local machine with the specifications detailed in Table 5.1. However, to enhance scalability and accessibility, the environment was later transitioned to a virtual machine hosted in the cloud. This transition allowed for more efficient collaboration and utilization of computational resources.

Table 5.1: Operating System, Hardware, and Software Environment

Platform	**Application**	**Specification**
Software	NumPy	1.26.2
	Pandas	2.1.3
	Scikit-learn	1.3.2
	Python	3.12.0
Hardware	Hardware Model	HP ProBook 640 G5 - later moved to a virtual machine in the cloud
	Firmware Version	R72 Ver. 01.03.04
	Memory	16.0 GiB
	Processor	Intel® Core™ i7-8565U × 8
Operating System	OS Name	Ubuntu 22.04.3 LTS
	Kernel Version	5.15.0-92-generic

5.3 Data Sources and Preprocessing

In this study, the data utilized consisted of greenleaf tea harvests obtained from Eastern Produce Kenya Limited (EPK). Specifically, greenleaf data was chosen instead of made tea data since EPK's made tea comprises a substantial portion of tea from various growers whose climatic data were unknown to the study. In addition to greenleaf yields, several other data inputs were considered, including rainfall, temperature, hail damage figures, and soil water deficit measured in millimeters (mm). The impact of hail on tea field yield can be highly detrimental, and therefore, the data regarding hectares damaged and kilograms lost in Made Tea due to hail was included. These figures were subsequently converted to an equivalent weight representing the loss in the greenleaf harvest. It should also be considered that 1 kilogram of greenleaf is required to produce 0.235 kilograms of Made Tea.

The measurement of soil water deficit played a crucial role in the study. It represents the disparity between the current soil water content and the field capacity, expressed in millimeters. Soil water deficit adversely affects various aspects of tea plant growth, including leaf size, stem extension, root proliferation, and plant-water relations. Moreover, it reduces water use efficiency. Essentially, soil water deficit signifies the variance between the actual amount of water present in the soil and its maximum water-holding capacity. Table 5.2 is a sample of the data obtained.

According to Bailey (2021), the average yield of made tea per acre in Kenya is approximately 900 kilograms per month. Given that EPK produces 22 million kilograms of made tea annually, we can derive estimations for the yield per hectare. By converting the monthly yield from acres to hectares, the estimated made tea yield per hectare per month is approximately 2226.75 kilograms. Multiplying this monthly yield by 12 months, the estimated yield per hectare per year is approximately 26,721 kilograms. Consequently, with EPK's annual production of 22 million kilograms of made tea, we can infer that the approximate land area utilized for tea cultivation by EPK is 822.86 hectares.

Table 5.2: Sample data representing key features for tea yield prediction.

year	month	rainfall	temp_max	temp_min	hail_damage_ha	hail_damage_kgs	swd	yield
2021	2	152.35	25.96	12.93	0	0	0	725970
2021	3	110.95	27	13.4	0	0	0	832140

2021	4	161.65	26.7	12.6	302.85	37372	0	768230
2021	5	308.4	26	13	0	0	0	777060
2021	6	52.25	25	12	49.64	3565	0	989600
2021	7	69.45	23.5	13	0	0	0	1011845
2021	8	163.35	24.5	12.4	327.46	37153	0	621200
2021	9	157.7	24.6	13.2	272.46	31628	0	842670
2021	10	91.2	24.8	13.7	228.63	56224	0	727890
2021	11	51.4	25	15	0	0	0	950930
2021	12	15	26	14	0	0	0	759900
2021	1	134.6	27	11	0	0	-49.68	510480
2021	2	151.3	27	12	0	0	-19.2	460505
2021	3	88.6	27	12	145.58	5003	-63.24	467390
2021	4	178.2	26	13	126.04	50723	-4	538945
2021	5	347.8	25	13	164.53	28810	-3.6	512715
2021	6	97.25	25	11	0	0	-14.28	632320
2021	7	66.6	25	12	369.49	58027	-46.2	565575
2021	8	146.5	25	11	0	0	-26.8	469355
2021	9	276.85	25	13	191.72	12974	-17.38	646280
2021	10	89.7	27	13	113.3	19063	-33.3	746175
2021	11	64	26	12	0	0	-62.7	577370

5.4 The Tea Yield Prediction Model

The developed tea yield prediction model utilized Support Vector Machine (SVM), a machine learning algorithm to forecast future tea yield based on historical and environmental factors. This chapter provides an in-depth understanding of the model architecture, training, and evaluation processes.

5.5 Model Architecture

The model's foundation lay in the SVM algorithm, a supervised learning technique capable of regression and classification tasks. SVM was used to identify an optimal hyperplane to separate data points, therefore maximizing the margin between different classes. For the tea yield prediction, the algorithm considered features such as year, month, rainfall, temperature, hail damage, and soil water deficit.

5.6 Data Preprocessing

Various preprocessing techniques, including data cleaning, normalization, and feature scaling, were applied to refine the dataset further. These steps collectively aimed at optimizing the dataset's quality and relevance, laying a solid foundation for the subsequent stages of model training and evaluation. The model split the 20% and 80% of the data into training and testing sets respectively.

5.7 SVM Hyperparameters

SVM models rely on several hyperparameters that influence their performance and effectiveness in predicting outcomes. These hyperparameters need to be carefully tuned to achieve optimal model performance. The key SVM hyperparameters include the kernel, regularization parameter (C), gamma (γ), and epsilon (ε). The choice of kernel significantly affects the model's ability to capture complex relationships in the data. SVM can utilize different types of kernels, such as linear, polynomial, radial basis function (RBF), and sigmoid, to map the input data into a higher-dimensional space. Each kernel type has its advantages and disadvantages, and the selection depends on the specific characteristics of the dataset.

The regularization parameter (C) controls the trade-off between maximizing the margin and minimizing the classification error. A higher value of C allows the model to classify training points correctly, potentially leading to overfitting, while a lower value of C may result in a wider margin and better generalization to unseen data. Gamma (γ) defines the influence of a single training example, with low values indicating a broader influence and high values indicating a more localized influence. In the RBF kernel, gamma determines the width of the Gaussian function, impacting the flexibility of the decision boundary. Epsilon (ε) is specific to the epsilon-SVR variant of SVM and defines the margin of tolerance where no penalty is associated with errors. It regulates the trade-off between margin width and training error. Tuning these hyperparameters is crucial to optimize the SVM model's performance and generalization ability. Techniques such as grid search or randomized search can be employed to systematically explore different combinations of hyperparameters and identify the optimal configuration for the given dataset.

5.8 Model Evaluation and Visualization

The model's evaluation was conducted using the mean squared error (MSE) and R-squared (R2) as the chosen evaluation metrics. These metrics were selected due to their widespread

adoption and proven efficacy in assessing predictive accuracy across various studies on prediction algorithms. MSE is a metric that provides a measure of the average squared difference between the predicted values generated by the model and the actual observed values. It is particularly valuable in assessing the precision of the model by quantifying how closely the predicted values align with the true values. For this study, the negative mean squared error (neg MSE) was specifically utilized during hyperparameter tuning. The neg MSE, in the context of grid search with cross-validation, guides the search for hyperparameters by aiming to minimize this metric. The negative sign is used to align with the convention of grid search, where larger scores are considered better. Therefore, optimizing for the neg MSE effectively corresponds to minimizing the actual mean squared error.

R-squared is a statistical metric that gauges the proportion of the variance in the dependent variable (tea yield in this case) that can be predicted from the independent variables (features like year, month, rainfall, etc.). It serves as an indicator of the model's explanatory power, providing insights into how well the chosen features explain the variability in the target variable.

5.9 Hyperparameter Tuning

An essential aspect of optimizing the SVM model's performance lies in the tuning of hyperparameters. In this study, an initial regressor script, named 'svm_tea_prediction.py' was used. A hyperparameter optimization technique called GridSearch was used.

5.9.1 GridSearch: A Systematic Approach

GridSearch operates by systematically testing a predefined range of hyperparameter combinations, searching for the configuration that yields the best model performance. This process was crucial in enhancing the SVM model's ability to make accurate predictions.

5.9.2 Derived Hyperparameters

The derived hyperparameters, as illustrated in Figure 5.1, represent the result of this tuning process. The best configuration, determined by the grid search, is as follows: {'C': 1000000, 'epsilon': 0.01, 'gamma': 1, 'kernel': 'rbf'}. These derived hyperparameters values formed the basis for subsequent model training and evaluation. Figure 5.1 shows the hyperparameters that emerged as optimal after executing the initial regressor script, 'svm_tea_prediction.py'.

```
[CV] END .....C=10000000, epsilon=1, gamma=1000, kernel=rbf; total time=   0.1s
[CV] END .....C=10000000, epsilon=1, gamma=1000, kernel=rbf; total time=   0.1s
[CV] END .....C=10000000, epsilon=1, gamma=1000, kernel=rbf; total time=   0.2s
[CV] END ...C=10000000, epsilon=1, gamma=1000, kernel=linear; total time=   1.4s
[CV] END ..C=10000000, epsilon=1, gamma=10000, kernel=linear; total time=   1.7s
[CV] END ..C=10000000, epsilon=1, gamma=10000, kernel=linear; total time=   2.2s
[CV] END .....C=10000000, epsilon=1, gamma=10000, kernel=rbf; total time=   0.1s
[CV] END .....C=10000000, epsilon=1, gamma=10000, kernel=rbf; total time=   0.1s
[CV] END .....C=10000000, epsilon=1, gamma=10000, kernel=rbf; total time=   0.1s
[CV] END ..C=10000000, epsilon=1, gamma=10000, kernel=linear; total time=   1.6s
Best Hyperparameters: {'C': 1000000, 'epsilon': 0.01, 'gamma': 1, 'kernel': 'rbf'}
Mean Squared Error (MSE): 34534318599.94538
R-squared (R2): 0.5599338284209354
jmasai@snf-5717:~/research$
```

Figure 5.1: Optimal Hyperparameters from Initial Regressor Execution

Figures 5.2 – 5.3 illustrate the impact of different hyperparameters on the model's Mean Squared Error (MSE) and R-squared (R2) scores, guiding the selection of the best hyperparameters. Figure 5.2 shows the impact of hyperparameter Gamma, on the neg MSE. Gamma is a hyperparameter used in non-linear SVM kernels. The neg MSE shows a pattern of improvement as Gamma increases initially. This improvement continues up to a value of 1. However, beyond this point, the negative MSE values start to decline as the value of Gamma increases further. This suggests that there exists an optimal range of Gamma values, and going beyond this range might lead to decreased model performance in terms of MSE.

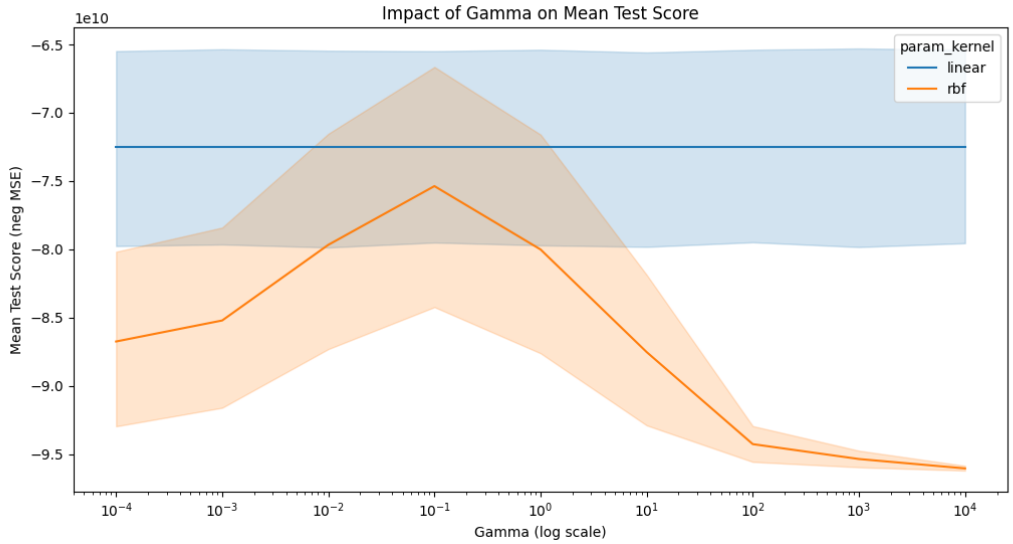


Figure 5.2: Impact of Hyperparameter Gamma on Negative Mean Squared Error (Neg MSE) in Support Vector Machine (SVM) Regression

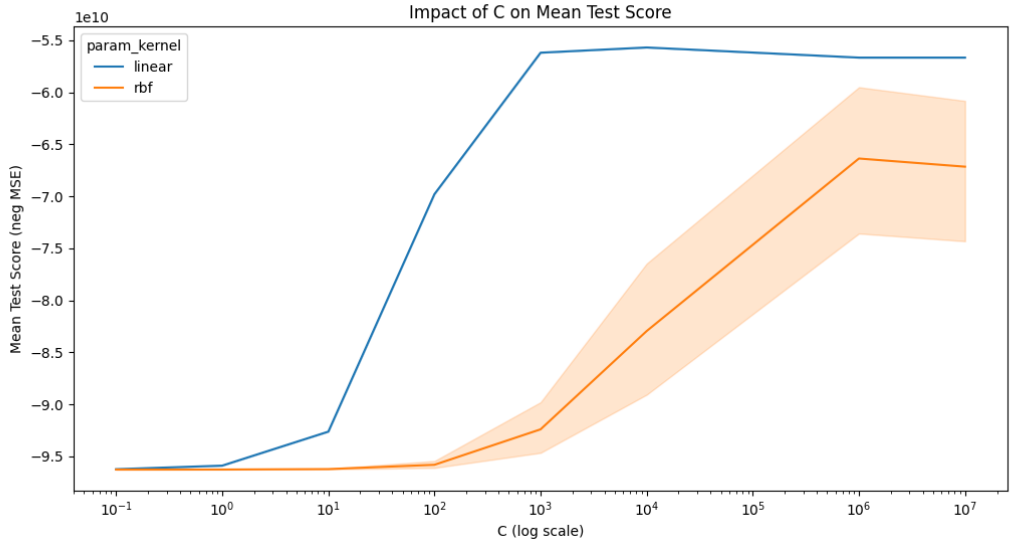


Figure 5.3: Impact of Hyperparameter C on Negative Mean Squared Error (Neg MSE) for Linear and RBF Kernels

Figure 5.3 shows the impact of hyperparameter C on Neg MSE for Linear and RBF Kernels. For the RBF kernel, the negative MSE initially increases as the value of C increases, reaching a peak around the value of 1000000. Beyond this point, the negative MSE starts to decline. This suggests that there exists an optimal range of C values for the RBF kernel, beyond which increasing C might lead to decreased model performance. On the other hand, for the linear kernel, the negative MSE improves steadily with an increase in C, up to a value of 1000. After this point, the negative MSE plateaus and remains relatively constant. This indicates that increasing C beyond 1000 does not significantly impact the performance of the linear kernel in terms of MSE.

5.10 Final Model Integration

The best hyperparameters obtained from the script were then integrated into the final prediction model, defined in the 'app.py' script. This integration ensured that the SVM model was configured with the optimal settings for accurate tea yield predictions. Figure 5.4 below show the integration of the best hyperparameters obtained into the final script.

```
# Train the SVM model
final_svm_regressor = SVR(C=1000000, epsilon=0.01, gamma=1, kernel='rbf')
final_svm_regressor.fit(X_train_scaled, y_train)
```

Figure 5.4: Integration of Optimal Hyperparameters into the Final Model Script

Following the training of the final model, evaluation was conducted using the derived optimal metrics (C=1000000, epsilon=0.01, gamma=1, kernel=rbf) on the test set. Figure 5.5 presents a visual representation of the model's predictive accuracy by plotting the actual tea yields against the corresponding predicted values.

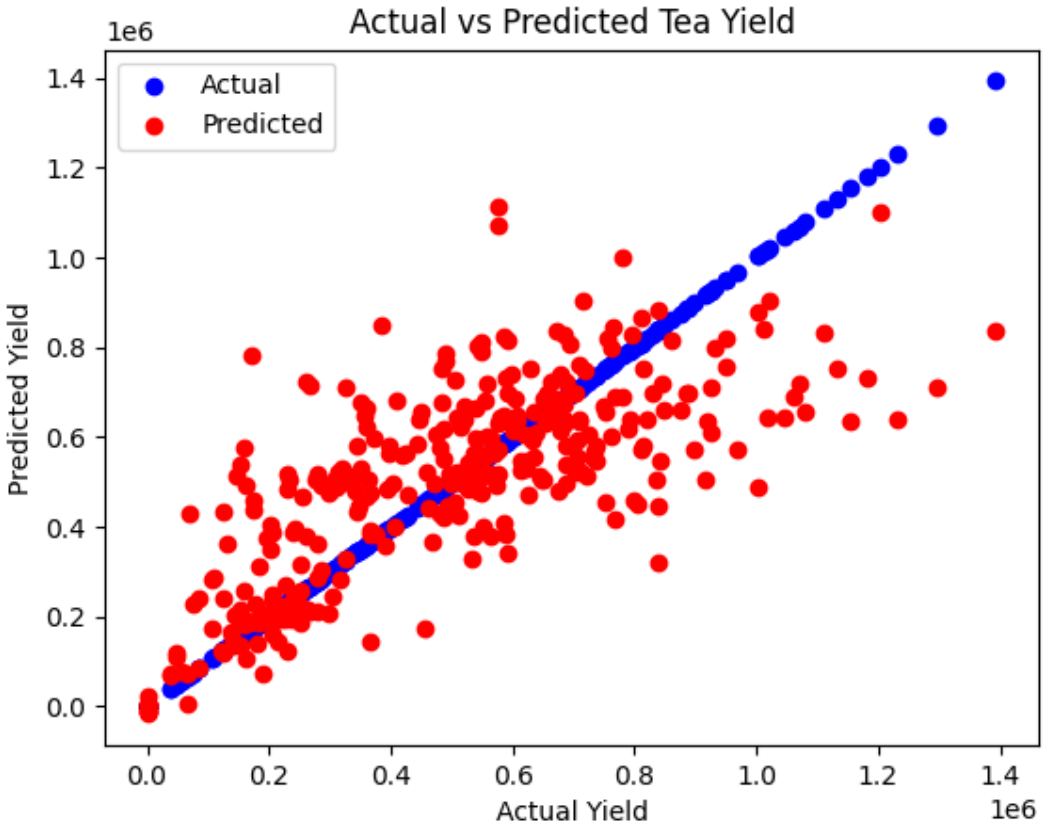


Figure 5.5: Comparison between Actual and Predicted Tea Yields

Figure 5.5 is a scatter plot illustrating the predictive accuracy of the model. It showcases the relationship between the actual tea yield, plotted on the x-axis, and the corresponding predicted values generated by the final SVM regression model, plotted on the y-axis.

5.11 Model Deployment: Interactive Prototype Web Portal

In the final phase of the model development, a user-friendly web portal was created to enable seamless interaction with the SVM model. Leveraging the Flask web framework, this portal allowed users to input pertinent environmental factors and obtain accurate predictions of tea yields. Figure 5.6 demonstrates the execution of the final regressor script, initiating the operation of the web frontend.

Figure 5.6: Execution of the Final Regressor Script Initiating the Web Frontend

5.11.1 Flask Web Framework

Flask, a micro web framework for Python, was chosen for its simplicity and flexibility. It provided a straightforward and efficient way to build a web application. Its lightweight nature made it ideal for this specific use case – deploying a predictive model through a minimalistic web interface.

5.11.2 Portal Development

The ‘app.py’ script served as the backbone of the web portal. This script defined two routes – the home route (‘/’) and the prediction route (‘/predict’). The home router rendered the input form using the ‘index.html’ template, while the prediction route processed user inputs, invoked the SVM model, and displayed the prediction on the ‘result.html’ template.

5.11.3 Web Interface Screenshots

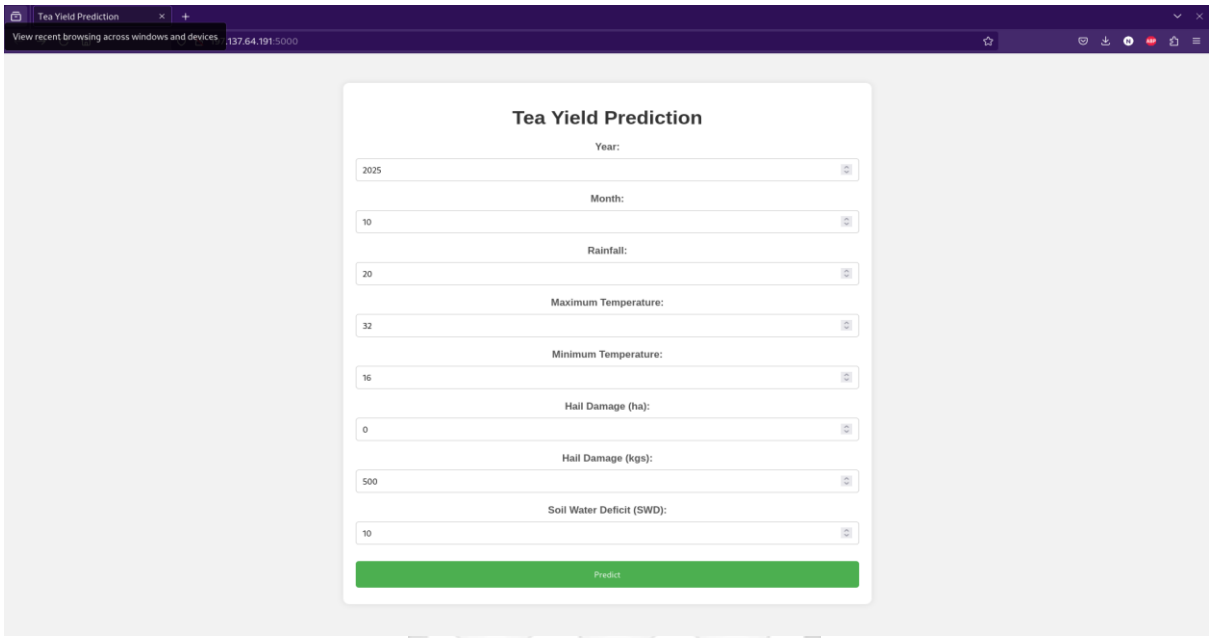


Figure 5.7: Home Page – Input page

Figure 5.7 captures the user interface of the home page, showing the input form where users can enter relevant environmental factors for tea yield prediction. The form collects data such as year, month, rainfall, temperature, hail damage, and soil water deficit (SWD).

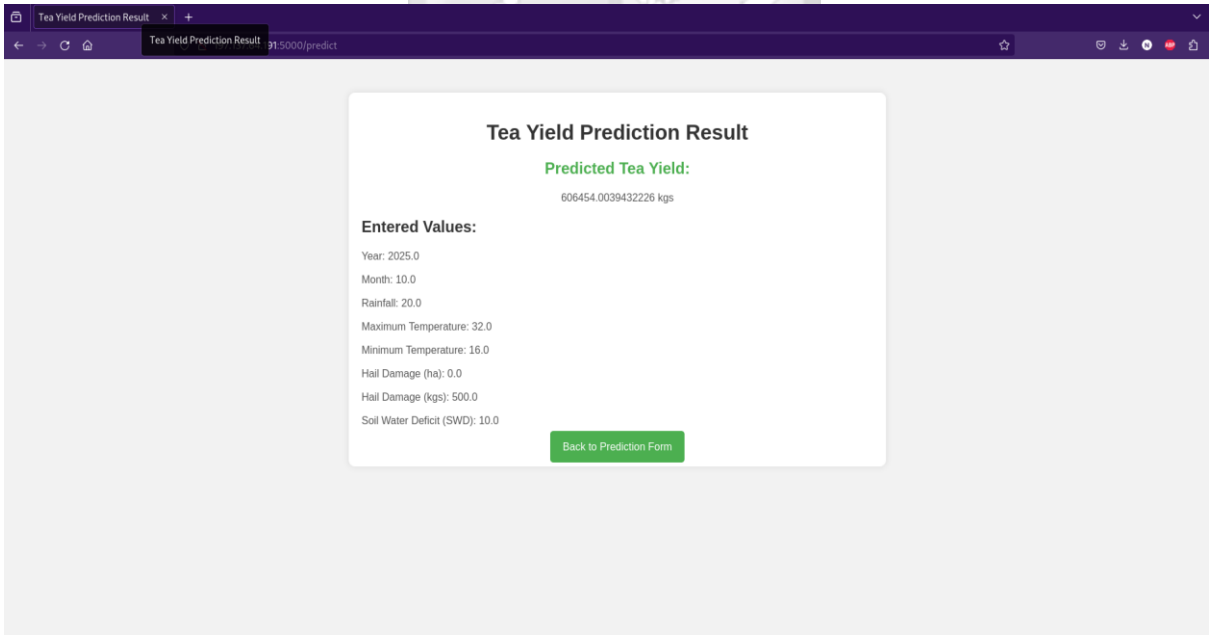


Figure 5.8: Prediction Result Page

Figure 5.8 illustrates the prediction result page displaying the model's forecasted tea yield. The user receives the predicted yield along with a breakdown of the input factors.

5.10.4 Why Flask?

Flask was chosen for its simplicity and ease of integration with machine learning models. Its minimalistic structure allowed for rapid development and deployment of the predictive model, making it an ideal choice for the prototype web portal. Additionally, Flask’s active community and available documentation provided resources for implementation.

Chapter 6: Discussion of Results

6.1 Introduction

This chapter details the findings that address the questions posed in Chapter 1. The primary objective was to develop a predictive model for tea yield, contributing to the enhancement of food security in Kenya. The selected machine learning algorithm for this task was the Support Vector Machine (SVM). A dataset sourced from Eastern Produce Kenya Limited, was used in shaping the predictive capacity of the model.

6.2 Methodological Recap

The SVM model was trained and tuned using a grid search approach, seeking the optimal hyperparameters for enhanced predictive accuracy. The best configuration, as determined by the grid search consisted of radial basis function (RBF) kernel, a regularization parameter (C) of 1, epsilon of 0.01, and a gamma value of 1. This model configuration was then evaluated on a test set.

6.3 Research Findings

6.3.1 Model Precision

The precision of the SVM model was assessed using the Mean Squared Error (MSE). The MSE, representing the average squared difference between predicted and actual values, revealed a value of 34534318599.95. While this provides an indication of the model's precision, it is important to interpret this value relative to the scale of the tea yield data.

The precision of the Support Vector Machine (SVM) model was evaluated using the Mean Squared Error (MSE), which serves as a measure of the average squared difference between the predicted and actual values. The MSE value obtained from the initial regressor was calculated to be 34534318599.95. This metric plays a crucial role in understanding the model's accuracy and effectiveness in predicting tea yield. However, to interpret the significance of this MSE value, it is essential to consider the scale and context of the tea yield data. MSE values are not inherently interpretable on their own, as they are influenced by the scale of the target variable. Therefore, it becomes imperative to contextualize the MSE relative to the range and variability of the tea yield data.

In the context of tea yield prediction, where the target variable represents the amount of tea produced, a MSE of 34534318599.95 indicates the average squared difference between the predicted and actual tea yield values across the dataset. This implies that, on average, the model's predictions deviate from the actual tea yield values by approximately 34534318599.95 units squared. However, without a clear understanding of the scale of the tea yield data, it is challenging to gauge the practical significance of this MSE value. For instance, if the tea yield data has a wide range and high variability, a MSE of 34534318599.95 might be considered acceptable. Conversely, if the tea yield data has a narrow range and low variability, this MSE value might indicate poor model performance. Therefore, further analysis is required to provide a comprehensive assessment of the model's precision. This analysis should involve comparing the MSE to other relevant metrics, such as the range of tea yield values, the distribution of prediction errors, and the model's performance relative to alternative methods or benchmarks.

Additionally, exploring the impact of different hyperparameters, such as the regularization parameter C, the kernel function, and the choice of kernel parameters (e.g., Gamma γ for RBF kernel), could offer insights into improving the model's precision. Fine-tuning these hyperparameters through techniques like cross-validation can help optimize the model's performance and reduce prediction errors. While the MSE provides a quantitative measure of the model's precision, its interpretation requires careful consideration of the tea yield data's scale and context. Further analysis and experimentation are needed to enhance the model's accuracy and effectiveness in predicting tea yield.

6.3.2 Predictive Power

R-squared (R2), a measure of the proportion of variance in the dependent variable explained by the independent variables, was also employed. The R2 value of 0.56 indicates that the model captured approximately 56% of the variability in tea yields, suggesting a moderate level of predictive power.

6.4 Research Objectives

The research objectives outlined in this study served as guiding principles for investigating and addressing key aspects related to the prediction of tea yield. These objectives were designed to systematically explore existing approaches, identify challenges, develop a predictive model, and validate its effectiveness. The following subtopics elaborate each research objective:

i. Analyze Existing Approaches for Crop Yield Prediction:

The first objective aimed to conduct a comprehensive analysis of existing approaches used for crop yield prediction. This objective was achieved through an extensive literature review, which encompassed various methodologies, techniques, and models employed in the field of agricultural yield forecasting. The review provided insights into the strengths, weaknesses, and limitations of different approaches, laying the groundwork for developing an effective prediction model for tea yield.

ii. Review the Current Approaches in Forecasting Crop Yield:

The second objective sought to examine the challenges associated with current approaches in forecasting crop yield. Through the literature review and empirical analysis, several challenges were identified, including data scarcity, model complexity, and the influence of environmental factors on yield variability. Understanding these challenges helped in devising strategies to address them in the development and validation of the tea yield prediction model.

iii. Development of a Model for Predicting Tea Yield:

The third objective aimed to develop a robust model for predicting tea yield. Leveraging insights from the literature review and employing advanced machine learning techniques, such as Support Vector Regression (SVR), a predictive model was constructed. The model incorporated relevant features such as climatic variables, soil conditions, and historical yield data to generate accurate predictions of tea yield.

iv. Validation of the Developed Model:

The final objective focused on validating the developed model to assess its performance and reliability. This validation was conducted using rigorous evaluation metrics, including Mean Squared Error (MSE) and R-squared (R2), to gauge the model's predictive accuracy. The results of model validation demonstrated its effectiveness in capturing the underlying patterns in tea yield data and making reliable predictions.

Chapter 7: Conclusion, Recommendations, and Future Work

7.1 Conclusion

In conclusion, the Support Vector Machine (SVM) model, employed for predicting tea yield, has showcased a promising level of predictive ability, as evidenced by its coefficient of determination (R2). This metric indicates the proportion of the variance in the tea yield data that is predictable from the independent variables included in the model. The R2 value attained signifies that a substantial portion of the variance in tea yield can be explained by the climatic and environmental factors considered in the model. However, a deeper analysis reveals that while the R2 value provides an indication of the model's overall performance, it is essential to consider additional metrics such as the Mean Squared Error (MSE) to gain a comprehensive understanding of its precision. The MSE, representing the average squared difference between the predicted and actual tea yield values, suggests that there is room for improvement in the model's precision. The magnitude of the MSE value indicates the extent to which the predictions deviate from the actual values, highlighting areas where the model may benefit from refinement.

7.2 Recommendations

Despite the encouraging performance demonstrated by the SVM model, it is imperative to acknowledge its limitations and areas for enhancement. The research findings underscore the complexity inherent in predicting tea yield, influenced by a multitude of factors ranging from climatic variations to agronomic practices. Addressing these complexities requires a multifaceted approach that incorporates advanced modeling techniques, comprehensive data collection, and domain-specific expertise. Moving forward, it is recommended to explore avenues for enhancing the predictive accuracy of the SVM model. This may involve refining the model architecture, optimizing hyperparameters, and incorporating additional features that capture the nuances of tea cultivation more comprehensively. Furthermore, efforts should be directed towards expanding the scope of the dataset, encompassing a broader geographical range and a longer temporal span, to improve the model's generalizability and robustness.

Based on the research findings, several recommendations can be proposed to further enhance the efficacy of tea yield prediction models:

i. Explore the inclusion of additional features or engineering new variables that capture the underlying mechanisms influencing tea yield variability. This may involve

integrating remote sensing data, soil characteristics, or socioeconomic indicators to provide a more comprehensive input space for the model.

ii. Investigate the potential benefits of ensemble learning techniques, such as combining multiple SVM models or integrating different machine learning algorithms, to leverage the strengths of individual models and improve predictive accuracy.

iii. Validate the developed models using independent datasets and conduct rigorous verification procedures to assess their robustness across diverse environmental conditions and geographical regions.

iv. Collaborate with stakeholders in the tea industry, including growers, agronomists, and policymakers, to ensure that the developed models address practical challenges and provide actionable insights for decision-making.

7.3 Future Work

Building upon the current research findings, several avenues for future work emerge, offering opportunities to advance the field of tea yield prediction:

i. Future studies could explore the temporal dynamics of tea yield variability and investigate the seasonality patterns and long-term trends using time series analysis techniques.

ii. Future studies could explore ensemble techniques, such as combining SVM with other algorithms, which could lead to a more robust and accurate predictive model.

iii. Future work could involve the implementation of a mechanism for continuous learning, where the model is periodically updated with new data, ensuring its relevance and effectiveness over time.

iv. Future studies could consider sending recommendations to farmers through SMS notifications, providing advice on agricultural practices and products to improve yield.

v. Incorporate IoT to gather real-time environmental data, contributing to the overall predictive accuracy of the model.

References

Apan, A. A., Phinn, S. R., & Scarth, P. (2017). Seeing the forest for the trees: A review of the application of optical remote sensing to forest biomass estimation. *Applied Geography*, 78,11-26.

Asseng, S., Ewert, F., Rosenzweig, C., Jones, J. W., Hatfield, J. L, Ruane, A. C., & Thorburn, P. J. (2013). Uncertainty in simulating wheat yields under climate change. *Nature Climate Change*, *3(9)*, 827–832.

Bailey, M. (2021). Kenyan Tea Market. Steeped Content. Retrieved March 29, 2024 from https://www.steepedcontent.com/blogs/blog/kenyan-tea-market.

Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.

Brownlee, J. (2016). *K Nearest Neighbors for Machine Learning.* Retrieved April 13, 2023 from https://machinelearningmastery.com

Chacko, S. M., Thambi, P. T., Kuttan, R., & Nishigaki, I. (2010). Beneficial effects of green tea: A literature review. Chinese Medicine, 5(1), 13.

Chen, J., Liu, W., Chen, H., & Chen, J. (2019). Crop yield prediction using deep convolutional neural networks. *Computers and Electronics in Agriculture*, 157,218-227. https://doi.org/10.1016/j.compag.2018.12.005

Chepkwony, J., Mburu, J., & Gitau, M. (2021). Use of Remote Sensing and Machine Learning for Crop Yield Prediction: A Review. *IEEE Conference on Sensors Applications Symposium (SAS)*, 1–6. IEEE.

Chepkwony, J. K., Kiruiro, E. M., Otiende, B. A., & Langat, P. K. (2021). An artificial intelligence-based crop yield prediction system for enhancing food security in Kenya. *International Journal of Advanced Research in Computer Science*, *12(2)*, 73–82. https://doi.org/10.26483/ijarcs.v12i2.7991

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, *13(1)*, 21–27.

Creswell, J. W. (2014). Research design: Qualitative, quantitative, and mixed methods approaches. *SAGE Publications.*, *4th ed*.

Dhiman, A., Singh, S. K., & Prakash, V. (2021). Support vector machine for prediction of rice yield in India using climate variables. *Environmental Processes*, 8(1), 257-271. doi: 10.1007/s40710-021-00591-1

FAO. (2003). Trade reforms and food security: Conceptualizing the linkages. *Food and Agriculture Organization of the United Nations*.

FAO. (2016). Crop yield. In FAO glossary of terms. *Food and Agriculture Organization of the United Nations.*

FAO. (2019). *Kenya: Climate change impacts and adaptation in agriculture.* http://www.fao.org/3/ca4373en/CA4373EN.pdf

FAO. (2021). *Kenya: Country Programming Framework 2018-2022.* http://www.fao.org/kenya/en/

Gandhi, R. (2018). *Support Vector Machine—Introduction to Machine Learning Algorithms*. Retrieved April 13, 2023 from https://towardsdatascience.com

Gao, F., Yao, X., Zhu, X., Huang, Y., & Chen, X. (2018). Predicting maize yield using remote sensing data combined with climate data and crop phenology information. *Remote Sensing*, *10(10)*, 1577.

Gao, Y., Chen, T., Huang, Q., & Zhang, C. (2020). Predicting rice yield using random forest model based on weather and soil data. *Information Processing in Agriculture*, 7(4), 584-591. doi: 10.1016/j.inpa.2020.05.001

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

Gómez-Sanchis, J., Palacios-Rodríguez, G., García-Sánchez, F., & García-Sánchez, E. (2020). *Agrometeorological models for the prediction of grapevine yield: A review. Computers and Electronics in Agriculture,*. 177. https://doi.org/10.1016/j.compag.2020.105691

Harrison, O. (2018). *Machine Learning Basics with the KNearest Neighbors Algorithm.* Retrived April 14, 2023 from https://towardsdatascience.com

Holzworth, D. P., Huth, N. I., deVoil, P. G., Zurcher, E. J., Herrmann, N. I., McLean, G., ... & Hochman, Z. (2014). APSIM—Evolution towards a new generation of agricultural systems simulation. Environmental Modelling & Software, 62, 327-350.

Huang, Z., Lan, Y., Zhang, H., Huang, C., Yang, J., & Liu, Z. (2018). Crop yield prediction using deep residual network with auxiliary classifier on UAV images. *Remote Sensing*, 10(11), 1798. doi: 10.3390/rs10111798

IPCC. (2014). Climate change 2014: Synthesis report. *Contribution of Working Groups I, II, and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change.* IPCC.

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.

Jones, J. W., Porter, C. H., Dinapoli, S., Gijsman, A. J., Batchelor, W. D., & Hunt, L. A. (2017). The DSSAT cropping system model. European Journal of Agronomy, 82, 10-25.

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147, 70-90. doi: 10.1016/j.compag.2018.02.016

Karthikeya, R., Sudarshan, M. R., & Shetty, S. M. (2020). Crop yield prediction using K-Nearest Neighbor algorithm. *International Journal of Computer Science and Information Security*, 18(5), 159-165.

Kangogo, M., Korir, N., Njiruh, P. N., & Kimutai, J. (2018). Assessing the status of tea fields and the effect on tea production in Bomet County, Kenya. *Journal of Agriculture and Environmental Sciences*, 7(2), 1-11. doi: 10.15640/jaes.v7n2a1

Kariuki, J., Musyimi, D. M., Nyaga, J. N., & Gichimu, B. M. (2020). Predictive modeling of tea yield in Kenya using time series analysis. *Advances in Agriculture*, 2020, 1-9. doi: 10.1155/2020/8885264

Karthikeya, H. K., Sudarshan, K., & Shetty, D. S. (2020). Prediction of agricultural crops using KNN algorithm. *Int. J. Innov. Sci*. Res. Technol, 5(5), 1422-1424.

Keating, B. A., Carberry, P. S., Hammer, G. L., Probert, M. E., Robertson, M. J., Holzworth, D., & Meinke, H. (2003). An overview of APSIM, a model designed for farming systems simulation. *European Journal of Agronomy*, 18(3-4), 267-288.

Khaled, A. A., Fathy, A. M., Nasr, E. S. A., & El-Bendary, N. (2020). Crop yield prediction using hybrid machine learning techniques. Computers and Electronics in Agriculture, 169, 105182.

Kihoro, J. M. (2019). Impact of climate change on tea production in Kenya. *International Journal of Science and Research (IJSR)*, 8(4), 1546-1552.

Kinyua, J., Karuma, A. N., & Ochora, J. (2021). The role of smallholder tea farmers in Kenya's economy. *Journal of Development and Agricultural Economics*, 13(2), 60-68.

Kochar, A. (2021). *Iterative Development: A Starter's Guide.* Retrieved April 05, 2023 from https://distantjob.com/blog/iterative-development/

Kuwataa, K., & Shibasaki, R. (2016, July 12-19). Estimating Corn Yield in The United States With Modis Evi and Machine Learning Methods. I*SPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, III(8).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

Liu, Y., Yang, G., Huang, Y., Liu, L., & Sun, Y. (2020). Estimating rice yield based on a novel remote sensing index and machine learning algorithms. I*nternational Journal of Remote Sensing*, 41(2), 504-520.

Manjula, A., & Narsimha, D. G. (2016). Crop Yield Prediction with Aid of Optimal Neural Network in Spatial Data Mining: New Approaches. *International Journal of Information & Computation Technology*, 6(1), 25-33.

Maxwell, J. A. (2021). *Qualitative research design: An interactive approach.* SAGE Publications.

Mitchell, T. (1997). *Machine learning*. McGraw Hill.

Montesinos-López, O. A., Martín-Vallejo, J., Crossa, J., Gianola, D., Hernández-Suárez, C. M., Montesinos-López, A., & Cuevas, J. (2021). A benchmarking between eight classes of machine learning algorithms for predicting complex quantitative traits using simulated and real plant breeding data. The Plant Genome, 14(2), e20089.

Moshou, D., Bravo, C., Oberti, R., West, J., McCartney, A., Ramon, H., & van der Sluijs, M. (2019). Advances in technologies for precision agriculture in grapevine. *Precision Agriculture*, 20(6), 1026-1042. doi: 10.1007/s11119-019-09667

Mureithi, J. G., Mwenda, E. M., Ndubi, J., & Muriungi, J. K. (2020). Predicting tea yields in small-scale farming systems using meteorological data: A case study of Tigania West sub-county, Meru County, Kenya. *African Crop Science Journal*, 28(4), 399-406. doi: 10.4314/acsj.v28i4.2

Mureithi, S. M., Ouma, G., & Kamau, J. (2020). Performance evaluation of crop yield prediction models in Kenya: A review. *Sustainable Agriculture Research*, 9(2), 1-12.

Pavani, S., & Augusta Sophy Beulet, P. (2022). Prediction of Jowar Crop Yield Using K-Nearest Neighbor and Support Vector Machine Algorithms. *In Futuristic Communication and Network Technologies: Select Proceedings of VICFCNT 2020* (pp. 497-503). Springer Singapore.

Rasheed, A., Zhang, Y., Xiao, Y., Zhang, Y., Xiao, Y., Cao, Y., & He, Z. (2019). Crop yield prediction using deep learning: A survey. Computers and Electronics in Agriculture, 162, 219-236.

Russell, S. J., & Norvig, P. (2016). Artificial intelligence: A modern approach (3rd ed.). Pearson.

Sahoo, S. K., Bhattacharya, A., & Sudarshan, P. B. (2021). Prediction of rice yield using machine learning algorithms. *Computers and Electronics in Agriculture*, 186, 106051.

Siganos, D., & Stergiou, C. (1996). Neural Networks. *SURPRISE 96 Journal*, vol 4.

Torres-Rua, A. F., Mares, V., Garcia, M. E., & Jones, J. W. (2020). Analysis of the accuracy and cost of satellite and UAV imagery for crop yield prediction. *Remote Sensing*, 12(15), 2478.

Wang, T., Liu, S., Zhang, L., & Wu, Y. (2019). A yield prediction model of winter wheat using an improved convolutional neural network. *Journal of Integrative Agriculture*, 18(7), 1637-1646. doi: 10.1016/S2095-3119(18)62164-2

You, J., Li, X., Low, M., Lobell, D., & Ermon, S. (2017). Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), (pp. 4559-4565).

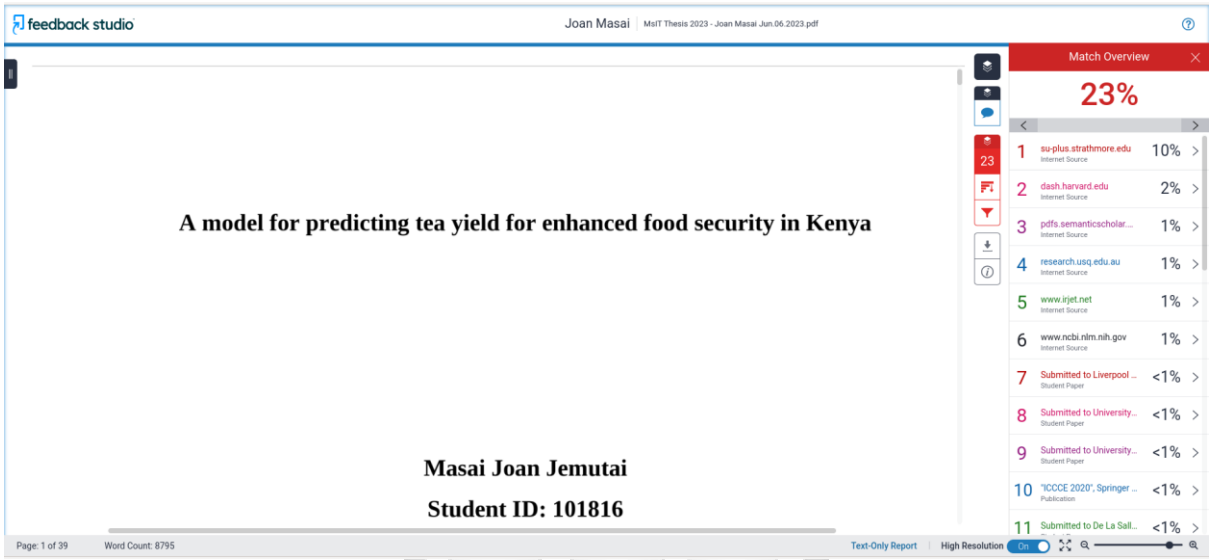
Zhang, H., Wei, S., & Zhang, H. (2019). A deep learning model for predicting maize yield using meteorological and remote sensing data. *Computers and Electronics in Agriculture*, 162, 653-662. doi: 10.1016/j.compag.2019.04.006

Zhang, H., Xie, W., Xiong, D., & Wang, X. (2020). Crop yield prediction model based on multi-source data fusion and machine learning algorithm. *Journal of Physics: Conference Series*, 1633, 012005. doi: 10.1088/1742-6596/1633/1/012005

Zhou, Y., Liu, X., Wu, X., Huang, Y., & Zhang, X. (2020). A hybrid model for crop yield prediction based on multiple data sources. *Journal of Integrative Agriculture*, 19(1), 244-255. doi: 10.1016/S2095-3119(19)62629-3

Appendices

Appendix A: Originality Report



Appendix B: Ethical Clearance Certificate

Strathmore UNIVERSITY

2nd August 2023

Ms Masai Joan Jemutai,
noajmasai@gmail.com

Dear Ms Masai,

RE: A Model for Predicting Tea Yield for Enhanced Food Security in Kenya

This is to inform you that SU-ISERC has reviewed and **approved** your above **SU-masters** research proposal. Your application reference number is **SU-ISERC1768/23.** The approval period is from **2nd August 2023 to 1st August 2024.**

This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used.
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-ISERC.
iii. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-ISERC within 72 hours of notification.
iv. Any changes anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-ISERC within 72 hours.
v. Clearance for the export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to the expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days of completion of the study to SU-ISERC.

Before commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology, and Innovation (NACOSTI) https://research-portal.nacosti.go.ke/ and obtain other clearances needed.

Yours sincerely,

Mr Ambrose Rachier,
Chairperson; SU-ISERC

STRATHMORE UNIVERSITY INSTITUTIONAL SCIENTIFIC AND ETHICAL REVIEW COMMITTEE
(SU-ISERC)
02-Aug-2023
Email:ethicsreview@strathmore.edu
P.O BOX 59857-00200
NAIROBI-KENYA

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email admissions@strathmore.edu www.strathmore.edu

Appendix C: Sample Data

year	month	rainfall	temp_max	temp_min	hail_damage_ha	hail_damage_kgs	swd	yield
2021	8	150	0	0	0	0	0	172685
2021	9	210	0	0	0	0	0	240320
2021	10	110	0	0	0	0	0	283360
2021	11	49	0	0	0	0	0	211175
2021	12	16	0	0	0	0	0	199770
2022	1	81.4	27.68	9.32	0	0	-202.46	645290
2022	2	65.7	27	9	0	0	-209.72	447605
2022	3	77.4	31	10	0	0	-247.04	489725
2022	4	192.5	26.8	10.2	0	0	-141.34	483345
2022	5	146.1	26	10	0	0	-86.48	1252055
2022	6	158.4	25	8.8	0	0	-2.4	1061285
2022	7	81.9	24.07	9.9	0	0	0	812975
2022	8	137.7	24.06	9.68	0	0	-2.4	1166235
2022	9	163.3	25.03	8.7	0	0	-10.32	816645
2022	10	75.1	25.94	9.48	0	0	-28.94	932800
2022	11	83.1	25.47	10.47	0	0	-31.24	1007040
2022	12	142	25.58	9.81	0	0	-14.8	863740
2022	1	69.5	24	11	84.04	7761	0	304750
2022	2	64	0	12	115.78	4729	0	224400
2022	3	70	25	12	0	0	0	300855
2022	4	223.5	24	12	196.58	4597	0	302660
2022	5	142	23	12	0	0	0	484760
2022	6	138	22	11	0	0	0	593645
2022	7	123	22	11	63.12	3916	0	410975
2022	8	173	21	11	56.85	17948	0	476550
2022	9	129.5	22	11	0	0	0	414875
2022	10	159	23	11	233.15	23525	0	460045
2022	11	106	23	12	43.38	7630	0	507895
2022	12	123	24	11	0	0	0	473095
2022	1	62	28.26	11.65	0	0	0	378345
2022	2	65	27	12	0	0	0	236975
2022	3	95.75	29	12	0	0	0	287105

2022	4	206.25	27	13	147.35	1254	0	291350
2022	5	162.5	27	13	0	0	0	886855
2022	6	166.75	22	11	174.72	57968	0	822360
2022	7	119.5	24	11	0	0	0	424170
2022	8	229.75	24	11	0	0	0	848465
2022	9	179	26	11	0	0	0	541965
2022	10	92.5	27	11	79.34	1015	0	685220
2022	11	51.25	26	11	0	0	0	683120
2022	12	94	26	10	0	0	0	473980
2022	1	71.2	24.29	12.96	0	0	0	480085
2022	2	80.7	26.9	11.2	233.88	38535	0	334475
2022	3	85.3	28.2	11.6	0	0	0	467925
2022	4	154.7	24	13	0	0	0	496450
2022	5	119.8	24	13	0	0	0	761115
2022	6	141.9	26.7	10.6	0	0	0	718230
2022	7	142.2	24.2	11.2	142.54	35573	0	603325
2022	8	198.2	21.98	12.4	347.6	94578	0	648780
2022	9	139	25.7	11.3	0	0	0	539990
2022	10	101.3	26.4	11.4	165.57	25383	0	765720
2022	11	122.3	24.7	11.5	0	0	0	801310
2022	12	131.2	23	12	0	0	0	634030
2022	1	88.5	21.7	10.7	0	0	0	447185
2022	2	162.2	28.1	10	0	0	0	320975
2022	3	112	28.1	9.6	0	0	0	546260
2022	4	198	29	8	0	0	0	576255
2022	5	141.7	28.03	8.84	0	0	0	707710
2022	6	127.8	28.2	8.7	0	0	0	649700
2022	7	136	28.5	8.6	0	0	0	565355
2022	8	229.5	28.3	9.2	0	0	0	634390
2022	9	208	26.2	9.87	86.75	154	0	502080
2022	10	101	26	11.1	0	0	0	712225
2022	11	95	29.8	8.8	0	0	0	755240
2022	12	166.5	30.9	6.3	0	0	0	521720
2022	1	69	27.13	7.81	0	0	0	358850
2022	2	76	26	8	0	0	0	238505
2022	3	78	28	8	0	0	0	300730

2022	4	255	27	10	0	0	0	332985
2022	5	161	27	9	22.5	167.27	0	751840
2022	6	179.5	25	9	46.38	142.36	0	703295
2022	7	160	24	10	103.96	743	0	516990
2022	8	244	23.61	10.16	436.45	41592	0	713035

Appendix D: Code for Initial Regressor

```
# Import necessary libraries
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.svm import SVR
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
import seaborn as sns
# Load the dataset from CSV
data = pd.read_csv('teadataset.csv')
# Split the dataset into features (X) and target (y)
X = data[['year', 'month', 'rainfall', 'temp_max', 'temp_min', 'hail_damage_ha', 'hail_damage_kgs',
'swd']]
y = data['yield'].ravel()
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.22, random_state=42)
# Standardize the features (important for SVM)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
# Define a dictionary of hyperparameters to tune
param_grid = {
    'kernel': ['linear', 'rbf'],
    'C': [0.1, 1, 10, 100, 1000, 10000, 1000000, 10000000],
    'gamma': [0.001, 0.0001, 0.01, 0.1, 1, 10, 100, 1000, 10000],
    'epsilon': [0.01, 0.1, 1]
}
# Initialize Grid Search with cross-validation
grid_search = GridSearchCV(estimator=SVR(), param_grid=param_grid,
scoring='neg_mean_squared_error', cv=3, n_jobs=-1, verbose=2)
# Fit the Grid Search to the training data
grid_search.fit(X_train, y_train)
# Get the best hyperparameters from Grid Search
best_params = grid_search.best_params_
print(f'Best Hyperparameters: {best_params}')
# Tabulate the results
results = pd.DataFrame(grid_search.cv_results_)
results.to_csv('grid_search_results.csv', index=False)
# Visualize the impact of different hyperparameters
# Line plot for C
plt.figure(figsize=(12, 6))
sns.lineplot(x='param_C', y='mean_test_score', hue='param_kernel', data=results)
plt.xscale('log')
plt.title('Impact of C on Mean Test Score')
plt.xlabel('C (log scale)')
plt.ylabel('Mean Test Score (neg MSE)')
plt.savefig('clineplot.png')
plt.show()
# Line plot for gamma
plt.figure(figsize=(12, 6))
sns.lineplot(x='param_gamma', y='mean_test_score', hue='param_kernel', data=results)
```

```
plt.xscale('log')
plt.title('Impact of Gamma on Mean Test Score')
plt.xlabel('Gamma (log scale)')
plt.ylabel('Mean Test Score (neg MSE)')
plt.savefig('gammalineplot.png')
plt.show()
# Line plot for R2 scores
r2_results = results.groupby(['param_kernel', 'param_epsilon', 'param_gamma',
'param_C'])['mean_test_score'].max().reset_index()
for kernel in ['linear', 'rbf']:
    for epsilon in param_grid['epsilon']:
        plt.figure(figsize=(12, 6))
        for gamma in param_grid['gamma']:
            subset = r2_results[
                (r2_results['param_kernel'] == kernel) &
                (r2_results['param_epsilon'] == epsilon) &
                (r2_results['param_gamma'] == gamma)
            ]
            plt.plot(subset['param_C'], subset['mean_test_score'], label=f'Gamma = {gamma}')
        plt.title(f'R2 Score for {kernel} kernel and Epsilon = {epsilon}')
        plt.xlabel('C (Regularization parameter)')
        plt.ylabel('R2 Score')
        plt.legend(title='Gamma')
        plt.xscale('log')
        # Save the plot as a PNG file
        plt.savefig(f'R2_{kernel}_epsilon_{epsilon}.png')
# Scatter plot for Actual vs Predicted Yield
final_svm_regressor = SVR(**best_params)
final_svm_regressor.fit(X_train, y_train)
y_pred = final_svm_regressor.predict(X_test)
plt.scatter(y_test, y_pred, color='blue')
plt.title('Actual vs Predicted Tea Yield')
plt.xlabel('Actual Yield')
plt.ylabel('Predicted Yield')
plt.savefig('scatter.png')
plt.show()
```

Appendix E: Code for Final Regressor

```
from flask import Flask, render_template, request
import pandas as pd
from sklearn.svm import SVR
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
app = Flask(__name__)
# Load your dataset
data = pd.read_csv('teadataset.csv')
# Features (X) and target variable (y)
X = data[['year', 'month', 'rainfall', 'temp_max', 'temp_min', 'hail_damage_ha', 'hail_damage_kgs',
'swd']]
y = data['yield']
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Standardize the data
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
# Train the SVM model
final_svm_regressor = SVR(C=1000000, epsilon=0.01, gamma=1, kernel='rbf')
final_svm_regressor.fit(X_train_scaled, y_train)
# Home route
@app.route('/')
def home():
    return render_template('index.html')
# Prediction route
@app.route('/predict', methods=['POST'])
def predict():
    # Get input values from the form
    year = float(request.form['year'])
    month = float(request.form['month'])
    rainfall = float(request.form['rainfall'])
    temp_max = float(request.form['temp_max'])
    temp_min = float(request.form['temp_min'])
    hail_damage_ha = float(request.form['hail_damage_ha'])
    hail_damage_kgs = float(request.form['hail_damage_kgs'])
    swd = float(request.form['swd'])
    # Standardize the input data
    input_data = scaler.transform([[year, month, rainfall, temp_max, temp_min, hail_damage_ha,
hail_damage_kgs, swd]])
    # Make a prediction using the trained model
    tea_yield_prediction = final_svm_regressor.predict(input_data)[0]
    return render_template('result.html', prediction=tea_yield_prediction,
                           year=year, month=month, rainfall=rainfall, temp_max=temp_max,
                           temp_min=temp_min, hail_damage_ha=hail_damage_ha,
                           hail_damage_kgs=hail_damage_kgs, swd=swd)
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
from flask import Flask, render_template, request
import pandas as pd
from sklearn.svm import SVR
from sklearn.model_selection import train_test_split
```

```
from sklearn.preprocessing import StandardScaler
app = Flask(__name__)
# Load your dataset
data = pd.read_csv('teadataset.csv')
# Features (X) and target variable (y)
X = data[['year', 'month', 'rainfall', 'temp_max', 'temp_min', 'hail_damage_ha',
'hail_damage_kgs', 'swd']]
y = data['yield']
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Standardize the data
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
# Train the SVM model
final_svm_regressor = SVR(C=1000000, epsilon=0.01, gamma=1, kernel='rbf')
final_svm_regressor.fit(X_train_scaled, y_train)
# Home route
@app.route('/')
def home():
    return render_template('index.html')
# Prediction route
@app.route('/predict', methods=['POST'])
def predict():
    # Get input values from the form
    year = float(request.form['year'])
    month = float(request.form['month'])
    rainfall = float(request.form['rainfall'])
    temp_max = float(request.form['temp_max'])
    temp_min = float(request.form['temp_min'])
    hail_damage_ha = float(request.form['hail_damage_ha'])
    hail_damage_kgs = float(request.form['hail_damage_kgs'])
    swd = float(request.form['swd'])
    # Standardize the input data
    input_data = scaler.transform([[year, month, rainfall, temp_max, temp_min,
hail_damage_ha, hail_damage_kgs, swd]])
    # Make a prediction using the trained model
    tea_yield_prediction = final_svm_regressor.predict(input_data)[0]
    return render_template('result.html', prediction=tea_yield_prediction,
                           year=year, month=month, rainfall=rainfall, temp_max=temp_max,
                           temp_min=temp_min, hail_damage_ha=hail_damage_ha,
                           hail_damage_kgs=hail_damage_kgs, swd=swd)
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

Appendix F: Code for Web Prototype – Home Page Input Form

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Tea Yield Prediction</title>
</head>
<body>
   <h1>Tea Yield Prediction</h1>
   <form action="/predict" method="post">
      <!-- Add input fields for each feature -->
      <label for="year">Year:</label>
      <input type="text" name="year" required>
      <label for="month">Month:</label>
      <input type="text" name="month" required>
      <label for="rainfall">Rainfall:</label>
      <input type="text" name="rainfall" required>
      <label for="temp_max">Maximum Temperature:</label>
      <input type="text" name="temp_max" required>
      <label for="temp_min">Minimum Temperature:</label>
      <input type="text" name="temp_min" required>
      <label for="hail_damage_ha">Hail Damage in hactares:</label>
      <input type="text" name="hail_damage_ha" required>
      <label for="hail_damage_kgs">Hail Damage in Kilograms:</label>
      <input type="text" name="hail_damage_kgs" required>
      <label for="swd">Soil Water Deficit:</label>
      <input type="text" name="swd" required>
      <!-- Add other input fields as needed -->
      <button type="submit">Predict</button>
   </form>
</body>
</html>
```

Appendix G: Code for Web Prototype – Prediction Result Page

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Tea Yield Prediction Result</title>
</head>
<body>
   <h1>Tea Yield Prediction Result</h1>
      {% if prediction is defined %}
      <p>Entered Values:</p>
      <ul>
         <li>Year: {{ year }}</li>
         <li>Month: {{ month }}</li>
         <li>Rainfall: {{ rainfall }}</li>
         <li>Temp Max: {{ temp_max }}</li>
         <li>Temp Min: {{ temp_min }}</li>
         <li>Hail Damage Ha: {{ hail_damage_ha }}</li>
         <li>Hail Damage Kgs: {{ hail_damage_kgs }}</li>
         <li>SWD: {{ swd }}</li>
      </ul>
      <p>Predicted Tea Yield: {{ prediction }}</p>
   {% else %}
      <p>There was an error in prediction.</p>
   {% endif %}
   <p><a href="/">Go back to input page</a></p>
</body>
</html>
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
```