University of Cape Town

Applications of Machine Learning in Apple Crop Yield Prediction

Student:
Deirdre van den Heever
VHVDEI001

Supervisor:
Mr Stefan S. Britz

Minor Dissertation for the degree Master's in Data Science

at the

Department of Statistical Sciences

October 24, 2021

Declaration of Authorship

I, Deirdre van den Heever, declare that this thesis titled, "Applications of Machine Learning in Apple Crop Yield Prediction" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Signed by candidate

Date: 27 October 2021

UNIVERSITY OF CAPE TOWN

Abstract

Faculty of Science
Department of Statistical Sciences

Master's in Data Science

Applications of Machine Learning in Apple Crop Yield Prediction

by Deirdre van den Heever

This study proposes the application of machine learning techniques to predict yield in the apple industry. Crop yield prediction is important because it impacts resource and capacity planning. It is, however, challenging because yield is affected by multiple interrelated factors such as climate conditions and orchard management practices.

Machine learning methods have the ability to model complex relationships between input and output features. This study considers the following machine learning methods for apple yield prediction: multiple linear regression, artificial neural networks, random forests and gradient boosting. The models are trained, optimised, and evaluated using both a random and chronological data split, and the out-of-sample results are compared to find the best-suited model.

The methodology is based on a literature analysis that aims to provide a holistic view of the field of study by including research in the following domains: smart farming, machine learning, apple crop management and crop yield prediction. The models are built using apple production data and environmental factors, with the modelled yield measured in metric tonnes per hectare.

The results show that the random forest model is the best performing model overall with a Root Mean Square Error (RMSE) of 21.52 and 14.14 using the chronological and random data splits respectively. The final machine learning model outperforms simple estimator models showing that a data-driven approach using machine learning methods has the potential to benefit apple growers.

Acknowledgements

I would like to express my sincere gratitude to the following people:

- Stefan Britz, my study leader, for his support and dedicated guidance
- Hugo van den Heever, for his continued love, motivation and encouragement
- Our Heavenly Father, for giving me the strength and determination to complete this project
- My immediate and extended family, for their support

The Author, June 2021

Contents

List of Figures

List of Tables

List of Abbreviations

MLR	**M**ultiple **L**inear **R**egression
ANN	**A**rtificial **N**eural **N**etwork
IoT	**I**nternet **o**f **T**hings
AI	**A**rtificial **I**ntelligence
SVM	**S**upport **V**ector **M**achines
WSU	**W**ashington **S**tate **U**niversity
FAO	**F**ood and **A**griculture **O**rganization of the United Nations
GDD	**G**rowing **D**egree **D**ays
PCA	**P**rinciple **C**omponent **A**nalysis
CV	**C**ross **V**alidation
RMSE	**R**oot **M**ean **S**quare **E**rror
OOB	**O**ut-**o**f-**B**ag
RSS	**R**esidual **S**um of **S**quares
RSE	**R**esidual **S**tandard **E**rror
MLP	**M**ulti-**L**ayer **P**erceptron
ReLU	**Re**ctified **L**inear **U**nit
Adam	**Ada**ptive **M**omentum (estimation)

Chapter 1

Introduction

This study proposes the application of machine learning techniques to predict yield in the apple industry. The purpose of this chapter is to introduce the study by contextualising the research and providing an overview of the research process. It includes background to the research problem, the research objectives and the dissertation outline.

1.1 Background

Agriculture is an important sector because it is the primary source of food in the world (United Nations, 2003). The sector faces many challenges as it is under pressure to produce more and better food. Agriculture is undergoing a digital revolution that has the potential to address some of these challenges (Bronson and Knezevic, 2016). In particular, there is a shift from traditional skills-based activities to digital practices, with the transformation of data into knowledge playing a central role. Machine learning is one of the most popular methods used to transform and make sense of data to enable better decision-making, and therefore an important enabler for the agricultural digital revolution (Kamilaris, Kartakoullis, and Prenafeta-Boldú, 2017).

Apples are widely cultivated and a prominent crop across the globe. In 2019, 87 million tonnes of apples were produced in the world (FAO, 2021). In the South African context, apples are one of the most important deciduous fruits when considering foreign exchange earnings, employment creation, and secondary agricultural activities. In South Africa, 900 000 tonnes of apples were produced in 2019, the industry employed 30 213 people and had a turnover of approximately R6 billion (Hortgro, 2019). Apple yield, measured as the volume and quality of fruit produced, is a critical success factor for the economic viability of an orchard (Bravin, Kilchenmann, and Leumann, 2009).

According to the Oxford dictionary, to predict means to "say something will happen in the future". The prediction of yield is an estimate for the yield that will be obtained over a specific time period. With machine learning, the estimate is calculated using a model that is based on a set of predictor variables. The prediction of yield in the fruit industry is important as it is the foundation for planning and decision-making in the fruit supply chain (Khaki and Wang, 2019; Liakos et al., 2018). It is also essential for sustainability to ensure an efficient use of natural resources (Paudel et al., 2021). Crop yield predictions impact resource decisions such as the number of people to employ, amount of packing material to procure, and sales programmes.

The use of relevant data and a model that suits the characteristics of the problem are two key elements of crop yield prediction. A good understanding of the factors impacting yield is a prerequisite for the choice of data. Machine learning methods are viewed as an appropriate consideration to model apple yield prediction. This is because machine

learning methods have shown benefits in many industries and have also been successfully applied to predict crop yield in other studies (Mishra, Mishra, and Santra, 2016). These methods can be used to model the relationship between input and output features and are especially valuable for non-linear, complex relationships such as in the case of yield prediction.

1.2 Research Problem

Yield prediction is challenging because the yield is impacted by a number of interrelated qualitative and quantitative factors such as the type of fruit, weather patterns, area of growth, and soil quality. Furthermore, the yield is also impacted by agricultural decisions such as irrigation and fertiliser applications. In practice, most predictions are based on expert knowledge or simple estimators calculated using historic performance (Cheng et al., 2017).

Due to the complex interaction between input and output factors, there is a need for an objective data-driven analytical model for yield prediction. Analytical models have the ability to take into account a large number of interrelated factors and has benefited numerous industries. In particular, the application of machine learning techniques is a growing trend and commonly used to solve business problems (Kamilaris, Kartakoullis, and Prenafeta-Boldú, 2017).

Numerous studies have been done in the field of crop yield prediction with the majority of recent studies involving at least some form of machine learning (Liakos et al., 2018). In apple yield prediction, most studies have however considered only one or two methods. An objective evaluation of various methods have shown to benefit yield prediction studies for other crop types. These studies have also shown that there is no single machine learning method that outperforms all the others and that the best-suited method and model is dependent on the problem (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014).

In addition, existing studies mostly consider a short prediction time-frame, ranging from two to five months prior to harvest. This approach benefits from the fact that better and more data is available closer to time of harvest. However, early yield prediction, defined as four to six months prior to harvest, is important for resource planning and budgeting purposes. Therefore, there is opportunity to develop a model to help with this type of prediction in the apple industry.

1.3 Research Objectives

This study aims to address the aforementioned needs by evaluating various machine learning methods for early apple yield prediction. In particular, the following methods are considered: Multiple Linear Regression (MLR), Artifical Neural Networks (ANN), random forests, and gradient boosting (also referred to as boosting). The primary objective of this study is as follows:

Determine the best-suited machine learning model for early apple yield prediction.

This study builds upon previous studies performed in the field of crop yield prediction, with a specific focus on an objective evaluation of various methods for early apple yield prediction. The primary objective of the study can be split into the following sub-objectives:

1. Establish the state of the digital revolution in agriculture
   (a) Define the concept of smart farming
   (b) Review machine learning applications in agriculture
2. Understand factors impacting apple yield
   (a) Introduce apple crop management
   (b) Define factors impacting apple yield
3. Define methods for crop yield prediction
   (a) Identify machine learning techniques used for crop yield prediction
   (b) Review previous crop yield prediction studies
4. Understand the data
   (a) Define predictor variables and collect data
   (b) Understand the data structure, trends and patterns
5. Develop machine learning models for apple yield prediction
   (a) Define the methodology
   (b) Explain the models that are applied
   (c) Build and optimise the models
   (d) Evaluate and compare results

The models are applied and evaluated using apple production data from the Western Cape region of South Africa. In 2019, this region produced approximately 80% of South African apples on 19 449 hectare (Hortgro, 2019). The dataset includes orchards from different farms in the area with the data spanning from 2016 to 2019. Primary production data, historic yield, climate-related data and secondary production data are considered as possible predictors for the yield.

1.4 Dissertation Outline

The thesis is structured in a logical manner to ensure that there is a continuous flow of key concepts. Each chapter aims to address certain research objectives that are shown in Table 1.1. The structure of the dissertation is outlined in separate sections below.

Chapter 1: Introduction
This chapter serves as the introduction for the study. It provides background to the problem, the problem statement, research objectives and the dissertation outline.

Chapter 2: Literature review
This chapter constitutes the literature analysis. It starts by providing context to the

Chapter	Section	Objectives
2 Literature Review	2.1 Agricultural digital revolution 2.2 Apple crop management 2.3 Crop yield prediction	1a; 1b 2a; 2b 3a; 3b
3 Data		4a; 4b
4 Methodology		5a; 5b
5 Application		5c; 5d

TABLE 1.1: Dissertation outline compared to research objectives

digital revolution in agriculture. Thereafter, apple crop management is explained, and finally existing studies on crop yield predictions are summarised.

Chapter 3: Data
The data used in the study are discussed in this chapter. It includes a description of the variables, a summary of the data, and trend analysis.

Chapter 4: Methodology
The methodology is explained in this chapter. First, the approach is outlined and thereafter a technical overview of the following machine learning methods are provided: MLR, ANNs and regression trees. The overview on regression trees includes a description of the random forest and boosting algorithms.

Chapter 5: Application
This chapter discusses the application of the following machine learning methods: MLR, ANNs, random forests and boosting. The aim is to use each method to build and optimise a model that predicts apple yield. The performance of the models are then compared to determine the best-suited method for the problem.

Chapter 6: Discussion and conclusion
The final chapter provides a discussion on the results and is used to conclude the study. It includes an evaluation and analysis on the final model's performance. Furthermore, research implications and opportunities for future work are also listed.

This concludes the introduction in which background to the study was provided, research objectives were listed, and the dissertation was outlined. The next chapter constitutes the literature review.

Chapter 2

Literature Review

The aim of this chapter is to gain an understanding of existing research and debates related to apple yield prediction, with a specific focus on machine learning applications in this field. The literature review includes various study domains in an attempt to give a comprehensive view of the research topic.

The review is organised as follows: First, the concept of the digital revolution in agriculture is introduced. It includes relevant enablers for crop prediction such as smart farming and machine learning. Thereafter, the basic principles of apple crop management are explained. An understanding of the factors that influence apple yield is important when performing predictions and, therefore, apple orchard design principles, orchard management practices, and environmental factors are discussed in this section. Lastly, a summary of existing crop yield prediction studies is provided. The discussion introduces some of the machine learning methods used for crop yield prediction, focusing on previous studies done in this field.

2.1 Agriculture Digital Revolution

Agriculture is an important sector in many countries because it is the primary source of food in the world (United Nations, 2003). The sector is under pressure, because it is challenged to produce more and better food to cater for the expansion of the human population, while increasing environmental, social en economic sustainability. According to Shankarnarayan and Ramakrishna (2020), these challenges are especially prevalent in developing countries, such as South Africa.

The advancement and integration of digital technologies into agricultural practices have the potential to address some of the challenges faced by the sector. According to Bronson and Knezevic (2016), agriculture is undergoing a digital revolution that can improve productivity and revolutionise not only the agricultural sector, but the entire food-to-table supply chain. Himesh et al. (2018) add that the revolution is driven by the Internet of Things (IoT), big data, cloud computing and sensor technologies, and that there is a rapid transformation from traditional skill-based agriculture to digital and knowledge-based practices, more commonly referred to as smart farming. These practices impact the way yield is predicted and therefore this section defines the concept of smart farming and the application of machine learning in agriculture.

2.1.1 Smart Farming

Smart farming is the application of information-driven and technology-enabled models to advance agricultural activities. It is a data-intensive approach that incorporates IoT,

sensors, smart devices, and big data analytics tools to enable the use of data in agricultural management. The concept of smart farming is often used synonymously with precision agriculture. However, according to Himesh et al. (2018), precision agriculture takes into account only the on-field variability, whereas smart farming includes the whole supply chain and is enhanced by context and situation awareness as well as real-time event triggers.

The concept of smart farming has gained popularity in recent years (Balducci, Impedovo, and Pirlo, 2018). Bacco et al. (2019) conducted a survey on research activities in this field, concluding that the main areas of interest for funded projects are cloud-based systems, unmanned vehicles and satellite-based activities; and that the trending topics in scientific literature include sensing techniques and management systems, unmanned vehicles, IoT platforms, and decision support systems. Decision support systems were highlighted as one of the most popular smart farming tools.

Balducci, Impedovo, and Pirlo (2018) looked at the management of data in the agricultural sector. The focus of the study was to design and deploy practical tasks to direct data-related efforts and investment. Among the tasks attempted were data forecasts, with the results showing that there is a large margin for innovation in the field. Issad, Aoudjit, and Rodrigues (2019) reviewed studies on smart agriculture and conclude that appropriate analytical techniques, such as data mining, are essential to analyse the increasing amounts of data.

The digital revolution in agriculture and smart farming tools are fuelled by data. Agricultural data are rapidly generated with the advancement in sensor and GPS technologies, IoT applications and communication technologies. Shankarnarayan and Ramakrishna (2020) mention that 90% of agricultural data have been collected in the past 5 years. This type of data is often referred to as big data, and characterised by their volume, variety and velocity. The generation of data and type of data generated impact traditional data collection, storage and reporting practices (Himesh et al., 2018). It also leads to business analytics at a scale and speed that was not possible previously, and has the potential to enable better decision-making. This type of analytics requires both specific tools and suitable methods for analysis. Machine learning is one of the most popular techniques used for this purpose (Kamilaris, Kartakoullis, and Prenafeta-Boldú, 2017). The next section defines machine learning and looks at the application of machine learning techniques in agriculture.

2.1.2 Machine Learning Applications in Agriculture

Artificial Intelligence (AI) is the process of simulating human intelligence using computer technology, and one of the subcategories of AI is machine learning. Mishra, Mishra, and Santra (2016) define machine learning as the process of giving knowledge to the machine through computer algorithms that improve automatically. It typically involves a learning process that aims to learn from experience by using a training set of data to perform a task. Machine learning methods are especially useful where the relationship between input and output features are not known or hard to obtain, and therefore it is ideal to model complex non-linear behaviours like the prediction of crop yield (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014).

Machine learning can broadly be categorised into two categories: supervised learning and unsupervised learning. In supervised learning, there is an associated response variable

for each predictor variable. Examples of supervised learning techniques include Artificial Neural Networks (ANNs), decision trees, regression analysis and Support Vector Machines (SVMs). In contrast, in unsupervised learning there is a vector of measurements but no set response variable, and the techniques are applied with the aim to better understand the relationship between observations (James et al., 2013). Hierarchical and k-means clustering are examples of unsupervised learning algorithms.

The application of machine learning techniques in agriculture is a growing trend (Mishra, Mishra, and Santra, 2016). This is because machine learning algorithms can help provide rich insights and assist farmers to make better decisions. Liakos et al. (2018)'s review shows that machine learning is used in various aspects of agriculture such as crop management, livestock management, water management, and soil management. In crop management specifically, it is used to help with yield prediction, weed detection, disease detection, crop quality, and species recognition, of which yield prediction and disease detection are the most popular applications. Various studies use data and images analysis tools to predict yield with the literature also encompassing a wide variety of crops such as wheat, soy beans, citrus, rice, and corn. These studies are further discussed in Section 2.3.

In summary, agriculture is undergoing a digital revolution that enables smart farming applications fuelled by data. Machine learning is an important technique that can be used to transform data into business knowledge with the ultimate goal of better decision-making. This study leverages the advancements in digital technology to evaluate the possibility of apple yield prediction by means of machine learning.

2.2 Apple Crop Management

Apples are one of the most important fruit crops worldwide – it is cultivated in at least 95 countries and 87 million tonnes of apples were produced in the world in 2019 (FAO, 2021; Li et al., 2019). In South Africa, approximately 900 000 tonnes of apples were produced in 2019 and apples were cultivated on 24 970 hectare (FAO, 2021). It is an important sector because it has the potential to make a significant contribution to economic development through job creation, employment, earning foreign exchange, rural development, and food security (Jafta, 2014). In 2019, the industry employed 30 213 people, supported an additional 120 853 dependents and contributed to approximately 63% of the total pome fruit turnover of R9 billion (Hortgro, 2019).

Apple production is complex. Managers of apple fruit crops need to find a balance between biological and economic factors when designing an orchard and implementing orchard management strategies. Producers are constantly looking for ways to optimise economic returns, and yield and quality have been identified as the most important factors that determine the success of an orchard (Bravin, Kilchenmann, and Leumann, 2009). Apple yield is a function of fruit number and size, and both may vary significantly from one season to the next (Manfrini et al., 2020; Bates, Morris, and Crandall, 2001). It is impacted by a vast number of factors, such as the cultivar type, rootstock, thinning practices, pruning, fertilisation, and environmental factors. A good understanding of the interplay between these factors and yield is essential for yield prediction and, therefore, it is the focus of this part of the literature review.

This section includes an introduction to apple production and a discussion on the factors that impact apple yield. The factors are organised into the following categories: orchard design, horticultural techniques, and environmental factors. The orchard design factors refer to decisions made prior to establishing the orchard, and include the site location, cultivar and rootstock selection, and orchard layout. Orchard management refers to the horticultural techniques that are applied throughout the lifetime of the orchard to ensure productivity and good yield. It includes factors such as thinning, pruning, fertilisation, and pest control. Lastly, the environmental factors are the climate-related factors that cannot be controlled by the grower.

2.2.1 Introduction to Apple Production

Commercial apple trees are composite biological units usually created by grafting. This means that each tree consists of a combination of rootstock, scion, and sometimes an interstem. Grafting involves taking a cutting or scion from a parent tree and physically placing it on a compatible rootstock so that the two plants fuse together as the tree grows.

Apple trees typically go through three phases of production over their lifetime. The first is a low production phase that is characterised by a gradual incline in yield. The second is a high production phase where the production volumes stabilise and the last is a gradual decline. The exact number of years that it takes for each phase varies widely in the literature. For example, Karami, Asadi, et al. (2017) consider the first and second phase to be 10 years each; Tahir, Johansson, and Olsson (2007) consider trees older than six years mature; and Goossens et al. (2017) consider trees mature at age four already.

Fruit growth of an individual apple starts after petal. It is a complex developmental process over a growing season that is affected by different environmental factors as well as orchard and tree management practices (Chaves et al., 2017). The five phenological stages of apple production are summarised as follows: bud formation, bud break, flowering, fruit growth, and fruit ripening (Karami, Asadi, et al., 2017). These phases are illustrated in Figure 2.1. The fruit originates from the base of the apple flower - after pollination and fertilisation, the seed cavity expands to form the fruit flesh. This expansion is first by means of cell division and later predominantly driven by cell expansion (Lakso and Goffinet, 2014). Apples bloom in spring and are harvested approximately 155 - 160 days after flowering, depending on the cultivar (Karami, Asadi, et al., 2017).

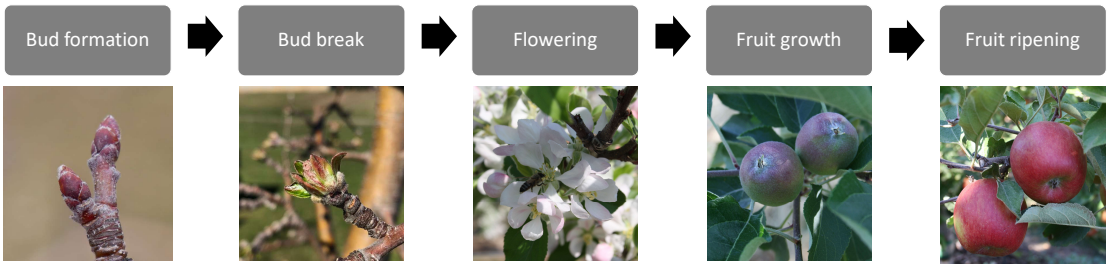


FIGURE 2.1: Phenological stages of apple formation adopted from Karami, Asadi, et al. (2017).

In South Africa, the apple harvesting season is roughly from February to May. Different cultivars are harvested at different times which helps with capacity planning. Furthermore, the exact harvesting period is also affected by climate conditions and, therefore,

differs between seasons. The blooming period is in spring during the months of September and October.

2.2.2 Orchard Design

Hester and Cacho (2003) highlight that many decisions need to be made prior to planting an apple orchard. These decisions constitute the design of the orchard and are critical to the economic viability of the orchard because it impacts the production potential of the trees. Each of the design decisions are discussed in subsections below.

Site

The first important decision that needs to be made when establishing a new orchard is the location, or site (Heinicke, 1975). Apple trees need full sunlight with at least 6 hours of sun per day. Besides general favourable environmental conditions for apple cultivation, the main factors to consider when choosing a site are frost, wind, and soil. The plant direction is also important as it influences light interception which is critical for apple growth (Lakso and Goffinet, 2014).

Cultivar

There are over 7 500 known apple cultivars, of which the following are the most prominent in South Africa: Golden Delicious, Granny Smith, Red Delicious, Gala, Fuji, and Cripps Pink (Hortgro, 2019). The choice of cultivar is important because it determines the fruit characteristics such as the colour, size, shape, taste, and firmness. It also impacts the biological pattern of the tree, like the blooming and harvesting period, and when the tree will first start to bear fruit. Commercial apple growers usually plant a variety of cultivars for risk dispersion, but also due to varying marketing requirements and to stagger harvest dates for efficient picking and packing operations (Hester and Cacho, 2003). In addition, some cultivars are more susceptible to pests and diseases than others and have different pollination requirements. Pollination is a key event in apple production, with the majority of cultivars depending on cross-pollination. Cross-pollination refers to the transfer of pollen from one plant to the flower of another genetically different plant.

Rootstock

Reig et al. (2018) argue that the choice of a suitable rootstock is equally important to the choice of cultivar for success in modern apple production systems. The choice of rootstock impacts the bearing characteristics of a grafted scion, influences the nutritional status of scions, and affects the flower development and fruit quality. Some rootstocks have also been seen to protect scions from diseases and abiotic stresses like frost, floods, and droughts (Reig et al., 2018). Most importantly, the rootstock is the dominant factor that controls the tree size and therefore a critical enabler for modern high-density apple orchards (Heinicke, 1975; Hester and Cacho, 2003).

Orchard Layout

Heinicke (1975) considers the tree density to be one of the most important factors influencing early production of apple trees. This is due to the fact that the spacing of trees have a direct impact on the light penetration abilities of the orchards. The orchard layout and tree architecture go hand-in-hand, because the tree architecture impacts the

density of trees that can be accommodated in the orchard.

Tree Architecture

The tree architecture also impacts light penetration, which is a determining factor for yield and orchard health (Anthony, Serra, and Musacchi, 2020). Middleton et al. (2002) state that the aim in orchard management is to create and maintain a tree form (height, spread, shape, and leaf area) that intercepts as much sunlight as possible and to ensure that the light reaches all parts of the canopy. The process in which trees are cut to achieve the desired architecture is called training. Even though trees are trained yearly by cutting shoots, it is important to choose a training system when designing an orchard, because training is most efficiently done when the trees are young and often require a physical support structure (Hester and Cacho, 2003). According to the Washington State University (WSU) Tree Fruit Research Commission, the following types of training systems are used for apple production: bi-axis, slender spindle, vertical axis, and V-system (WSU, 2021).

2.2.3 Orchard Management

Orchard management involves horticultural techniques such as training, pruning, and irrigation that are required for good health of the trees and optimal fruit production. It is a complex task as it involves the balancing of many different factors. A comprehensive review on orchard management is beyond the scope of this study, but an introduction is provided in the subsections below.

Training

As part of orchard management, training is the process of directing tree growth into a desired shape according to the architectural goal. The desired architecture or training system is usually decided on during orchard design, however the training process is a yearly effort and essential part of good orchard management (Hester and Cacho, 2003).

Tree Pruning

Pruning refers to the selective removal of a portion of a tree in order to correct or maintain the desired tree structure. It is also used to remove diseased or dead branches. Apple trees are usually pruned during late winter. In South Africa, pruning typically takes place in June, July or August, depending on where the orchards are located. Training and pruning are inter-related and need to be performed together to obtain one goal. According to Middleton et al. (2002), the aim is to obtain an orchard with midseason diurnal light interception of 60%, a leaf area index[1] of close to 2.0, trees of approximately 3 meters tall, and trees that are discrete units with tops that do not merge.

Thinning

Apple trees with heavy bloom typically produce 10 to 15 times more flowers that could result in double the amount of fruit than desirable for a good commercially-viable harvest (Lakso and Goffinet, 2014). Thinning is therefore required to concentrate the growth.

[1] In broadleaf canopies, the leaf area index is defined as the one-sided green leaf area per unit ground surface area. Note that it is a dimensionless quantity.

Apple thinning can be done by hand or chemically. It is an important practice to increase fruit size, improve fruit quality, and avoid tree breakage. The timing of thinning procedures is important. Lakso and Goffinet (2014) highlight the need for early and effective thinning, because the size of the fruit depends primarily on the number of cells in the fruit, which is a result of cell division occurring during the first few weeks after bloom.

Pest and Disease Control

Pests and diseases can be detrimental to an apple harvest. The major diseases in apples are bitter rot, white rot, marssonina blotch, and alternaria blotch; the major pests are spider mite, fruit moth, leafminer, aphid, and the hemipteran bug (Lee et al., 2007). It is important that producers are able to recognise pests and understand pest biology, because appropriate preventive measures are important for healthy and productive trees.

Irrigation

Most major apple producing regions depend on irrigation (Heinicke, 1975). Accurate water supply by means of irrigation is important for fruit productivity and quality – too little or too much water can lead to disorders such as bitter pit, root health diseases, and lower volumes and quality of fruit. The amount of water given at a time and also the timing of the irrigation is important, which highlights the need for a well thought-through irrigation schedule (WSU, 2021).

Soil and Nutrition

High quality soil and nutrition are critical success factors for apple yield and quality. This is because healthy trees depend on healthy roots to absorb water and nutrients from healthy soils. Soil quality can be defined as the soil's ability to support productive trees without negatively impacting the surrounding environment. The health of soil is influenced by the interplay between biological, physical, and chemical properties of the soil. Fruit tree nutrition is a specialist and complex topic that needs to be managed constantly, but the general aim is to fertilise trees based on the nutrient demand minus the supply in the soil (WSU, 2021).

Orchard Floor Management

Orchard floor management is also important. It includes topics such as orchard floor sanitation, ground cover crop, weed control, compost and soil. It is necessary to avoid unnecessary fungi and diseases that can inoculate healthy trees, and also reduce the amount of pesticides required for disease control (WSU, 2021). Ground cover crops are used as part of other orchard floor management practices for a variety of reasons, such as maintaining soil fertility, reducing weeds, and moderating soil temperature and moisture (Atucha, Merwin, and Brown, 2011). Heinicke (1975) mention that cover crops are an efficient method to control tree growth. It can be treated as a two-way system whereby the growth of the trees and cover crop are managed simultaneously to achieve the desired tree growth.

2.2.4 Environmental Factors

Besides the orchard design and management, apple production is also impacted by environmental factors that are not within the control of the grower. In general, apples

require cold temperatures in winter for dormancy (1000 to 1600 hours below 7 degrees Celcius), temperatures between 18 and 22 degrees Celcius for germination, and a certain number of heat units to germinate, grow, and mature (Karami, Asadi, et al., 2017).

Various studies have been performed to better understand the relationship between climatic conditions and apple production, of which many are focused on the impact of climate change. Li et al. (2019) studied the possible impact of climate change on the apple yield in Northwest China and considered different climatic variables such as temperature, precipitation, and frost. The study concludes that the main meteorological factors affecting yield are the total solar radiation during summer, evaporation during summer and autumn (April to September), and rainfall and minimum temperature in mid-April. In addition, the blossoming period was found to be especially sensitive to climatic conditions.

Lindén (2001) used discriminant and cluster analysis to identify the most important climatic factors associated with winter injury in Finland. The study found that the following variables had the biggest impact on yield: mean temperatures in September, January and February; minimum temperatures in December and February; maximum temperatures in December and January; precipitation in August and September; and the cumulative Growing Degree Days (GDD).

GDD is a common measurement for the number of heat units over a period of time in environmental analysis. It originates from the theory that different processes are activated at different thresholds. The base temperatures for apples vary slightly according to the cultivar. However, a commonly used base temperature is 7 degrees Celsius (Karami, Asadi, et al., 2017). Chaves et al. (2017) developed a model (using the Bon Verthalanfy statistical model and Euler integration method) to predict apple growth based on the cumulative GDD which shows physiological time. The relative error between predicted and observed fruit diameter was less than 3 percent in most cases.

In summary, temperature indicators, precipitation, sun hours, and the GDD are the most common factors used for apple yield prediction. There are contrasting views with regards to the relative importance of these factors and, therefore, this study aims to include as many of these factors as possible in the prediction model.

This concludes the discussion on apple crop management. This section shows that apple yield is affected by an interplay between multiple factors and feeds into the choice of predictor variables to include in the model. However, the decision does not only depend on the importance of the factors, but also on the data availability and prediction timeframe, which is further discussed in Chapter 3.

2.3 Crop Yield Prediction

Crop yield prediction is the short- or long-term prediction of the volume or quality of crop a harvest produces. It is important because it impacts resource requirements for harvesting and marketing, agricultural management strategies, and international crop trading (Liakos et al., 2018; Khaki and Wang, 2019; Kim et al., 2019). Yield prediction is a major challenge in agriculture. The issue is that predictions are often based on industry knowledge and years of experience in the field, and it is difficult to include the

many interrelated factors into the forecast (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014). Furthermore, the yield is impacted by farmer decisions such as the application of irrigation and pest control mechanisms, as well as uncontrollable factors that are difficult to foresee and take into consideration when forecasts are made.

Simple estimators on historical performance such as the average yield over a period are often used as forecasts (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014; Cheng et al., 2017). However, yield varies spatially and temporally with a non-linear behavior that causes large deviations from one year to the next. This emphasises the need for tools and data-driven models that can accommodate the complexity of the interrelated factors. These types of models, using statistical analysis, have been widely applied in recent years to gain a better understanding of the relationship between factors impacting yield.

This section starts with a brief introduction to the various machine learning techniques used to predict yield. It is followed by a summary of previous studies performed in the field. First, studies related to yield prediction of any type of crop, and thereafter studies specifically related to apple yield prediction, are discussed.

2.3.1 Machine Learning Techniques for Yield Prediction

A variety of machine learning techniques have been used and evaluated for their ability to perform crop yield predictions. These techniques treat the yield as an implicit function of the input variables. According to Mishra, Mishra, and Santra (2016)'s review, the following machine learning methods are the most popular for crop prediction: Artificial Neural Networks (ANNs), decision trees, regression analysis, clustering, Principle Component Analysis (PCA) and time-series analysis. This section introduces these methods to provide context for the remainder of the discussion on crop yield prediction. A technical overview of the methods that are applied in this study is provided in Chapter 4.

ANNs are computing systems that mimic the functioning of biological neural networks in the human brain. These networks model the relationship between a set of input parameters and output parameters with the use of highly interconnected processing elements. It has a node-link structure that typically consists of input, output, and hidden layers. Balducci, Impedovo, and Pirlo (2018), Cheng et al. (2017), and Khaki and Wang (2019) provide examples of how ANNs are used for crop yield prediction.

Decision tree methods involve segmenting the predictor area into a number of regions and performing predictions per area (James et al., 2013). In the simplest form, a decision tree is a collection of if-else statements that are applied to data to make a prediction. Two ensemble tree-based methods that are increasing in popularity are random forests and boosting models. With random forests, large number of decision trees are generated with slightly different characteristics by using random sampling of the training set (Breiman, 2001). With gradient boosting (boosting), trees are built sequentially in a stage-like fashion, meaning that each tree learns from the previous tree. Jeong et al. (2016) and Paudel et al. (2021) show how random forests and boosting can be used for crop yield prediction.

Linear regression is one of the most common techniques used in regression analysis. With this technique, the aim is to establish a relationship between two variables using a straight line. Multiple Linear Regression (MLR) is an extension of linear regression

that can be used when there is more than one predictor variable. The goal is to obtain a model that predicts as much variation in the response variable as possible using the predictor variables (James et al., 2013). Many crop yield prediction studies use a MLR model as a base model to compare with other models (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014; Balducci, Impedovo, and Pirlo, 2018).

Clustering involves grouping objects that are similar to each other using algorithms such as hierarchical or K-Means clustering. Paudel et al. (2021) and Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante (2014) show how clustering can be used for crop yield prediction. PCA is a common tool for dimensionality reduction and especially useful when dealing with a large number of predictor variables such as in the case of yield prediction. Lastly, time series analysis consist of methods that try to make sense of time series data, either by understanding the underlying context of the data points or by making forecasts (Montgomery, Jennings, and Kulahci, 2015). The next section takes a closer look at existing studies in the field of crop yield prediction.

2.3.2 Crop Yield Prediction Studies

This section considers yield prediction studies in general, meaning that the studies could concern the prediction of any type of crop, not just apples. The aim is to gain an overview of the field and also to identify possible opportunities for methods and practices that can be extended to the prediction of apple yield.

A number of studies have been performed with the aim to predict crop yield. Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante (2014) compared different machine learning and linear regression techniques in terms of their predictive ability for various crop types including soybean, tomato, potato, peppers, and beans. Temperature, irrigation, and crop type data were used as possible predictor variables. The following techniques were included: SVM, MLR, k-nearest neighbour, ANN and M5-prime regression trees. The M5-prime regression trees method delivered the best results. It was followed by the k-nearest neighbour techniques and SVM. The MLR model achieved the poorest overall performance.

Khaki and Wang (2019) designed a deep neural network approach to predict maize yield. The study used weather data, and the genotype and yield of 2267 maize hybrids in 2247 locations for 8 years. The model outperformed other popular methods such as regression trees and shallow neural networks. It resulted in a Root Mean Square Error (RMSE) of 12% of the average yield and 50% of the standard deviation for the validation data set. In addition to the method for prediction, the study also showed that the environmental data had a bigger impact on the yield than the genotype.

The use of satellite image processing is becoming increasingly popular for the prediction of crop yield (Kim et al., 2019; Jeong et al., 2016). The most commonly used index is the normalised difference vegetation index (NDVI) – an index for the normalised difference between the red and near-infrared signals. Russello (2018) used satellite images and convolutional neural networks to predict soybean crops. The model outperformed ridge regression, decision trees, and deep neural network models.

Jiang et al. (2004) built an ANN model for the prediction of wheat yield and compared the performance to a MLR model. Five predictor variables were included; four of the variables were derived from remote sensing data (NDVI, canopy surface temperature,

solar radiation, and water index) and the other variable is the average yield. The results show that the ANN model outperformed the MLR model with an average relative error of 3.5% compared to 11.5%. The ANN model was also found to be more stable and consistent than the MLR model.

Kim et al. (2019) used satellite images and meteorological and hydrological data to compare different AI models for the prediction of soybean and corn crop in the United States. The following models were compared: multivariate adaptive regression splines, SVMs, random forests, extremely randomised trees, ANNs and deep neural networks. The deep neural network delivered the best results. Interestingly, weather data only for July and August were included as these were found to have the most prominent impact on soybean and corn crop yield.

Jeong et al. (2016) compared a random forest model to a MLR model for wheat, maize and potato yield prediction. The random forest model outperformed the MLR model. The RMSE for the random forest model ranged between 6% and 14% of the average yield, compared to 14% – 49% for the regression model. The study used more than 10 predictor variables of which the majority are climate-related variables such as the average monthly temperature and precipitation. Other variables include irrigation, soil quality and fertiliser application. Fukuda et al. (2013) also showed that random forests can be used to estimate mango yield under different irrigation schemes.

Paudel et al. (2021) recently evaluated the performance of four machine learning techniques (ridge regression, SVMs, k-nearest neighbours regression and boosting) for crop yield prediction. The study combined machine learning with agronomic principles of crop modelling to create a baseline for large-scale crop yield prediction. The following crop types were included: potatoes, wheat, barley, sunflower, and beet. The boosting model delivered the best results overall and the early season predictions made by the combination model was comparable to forecasts in the European Commission's MARS Crop Yield Forecasting System. The model serves as a baseline that motivates the advantages of incorporating machine learning in crop yield prediction processes.

In summary, a number of different machine learning algorithms have been used for crop yield prediction. Amongst the most popular are ANNs and ensembled regression tree methods such as random forests and boosting. A couple of studies have compared different machine learning methods for the prediction of crops. The contrasting results between these studies accentuate the fact that there is no one machine learning model that works best for all types of yield prediction (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014; Khaki and Wang, 2019; Kim et al., 2019).

2.3.3 Apple Yield Prediction

The previous section considered crop prediction studies in general, whereas this section focuses on a more limited set of studies specifically concerned with the prediction of apple yield. Apple yield prediction studies can roughly be categorised into three groups. The first group consists of studies that consider environmental factors and crop type data to predict yield, the second are studies that incorporate image analysis and processing for apple detection and subsequently yield prediction, and the third are combinations of the first two groups.

Balducci, Impedovo, and Pirlo (2018) used historic data to predict yearly apple and pear harvests. The following factors were included: crop type, harvest year, geographical area, crop production amounts, temperature, rainfall, and fertilisation. Two models were compared: MLR and ANNs. The ANN model delivered superior results on both apple and pear crops. For apples, it resulted in an average prediction error of 9.19%, compared to 30.77% for the MLR model.

Li et al. (2019) predicted apple yield in the context of climate change. The study used grey relational analysis to assess 88 climatic factors in relation to meteorological yield of apples. Thereafter, a SVM model was used to make a quantitative prediction of the yield considering two climate change scenarios. The study showed that spring climate factors have the biggest impact on yield.

Stanley, Stokes, and Tustin (2000) considered the impact of environmental indicators on the early yield prediction of apple fruit size. The study concluded that there is a significant correlation between the final fruit weight and the accumulated GDD from full bloom to 50 days after full bloom.

A study performed by Kaack, Pedersen, et al. (2010) aimed to determine the relationship and interaction between fruit size, fruit weight, fruit quality, and climate factors in order to predict the optimal date to harvest apples. The study performed multiple regression analysis with forward stepwise selection of the climate factors. The results show that the fruit diameter is significantly affected by the degree days and evaporation potential; the fruit weight is only significantly impacted by the degree days; and that the fruit quality is affected by the degree days and relative humidity.

Cheng et al. (2017) performed early yield predictions with ANNs by using image analysis and tree canopy features. The following features were extracted from the images and acted as predictor variables: number of apples, total cross-sectional fruit area, cross-sectional foliage area, and cross-sectional area of small fruits. The results showed that the coefficient of determination (R^2 statistic) was 0.81, the mean absolute percentage error was 10.7% and the RMSE was 2.34 kg per tree. Linker (2018) predicted apple yield by processing night-time images. First, apples were detected and counted by analysing the images, and thereafter yield was predicted using a regression model. The overall yield estimation error was approximately 10% of the actual yield.

Table 2.1 gives a summary of the studies that are discussed. The majority of apple yield prediction studies consider one or two machine learning methods. Studies focusing on other (non-apple) crop types have shown that the best-suited machine learning method is dependent on the problem (Mishra, Mishra, and Santra, 2016). Therefore, there is a need for an objective evaluation of different methods for apple yield prediction.

Furthermore, past apple-related yield prediction studies mainly focused on either environmental data or image analysis. In addition to evaluating more than two machine learning methods, there is also a need to incorporate past yield performance with environmental factors to predict apple yield. In practice, most apple growers use simple yield estimators to forecast yield (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014), and there is a potential benefit in combining these estimators with machine learning methods.

TABLE 2.1: Summary of crop prediction studies

	Models Evaluated	**Best Model**	**Crop Type**	**Reference**
1	ANN, MLR	ANN	Wheat	(Jiang et al., 2004)
2	Random forests, MLR	Random forests	Wheat, potato, maize	(Jeong et al., 2016)
3	SVM, MLR, k-nearest neighbour, ANN and M5-prime regression trees	M5-prime regression trees	Soybean, tomato, bean, potato, peppers	(Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014)
4	Deep neural network, regression trees, shallow neural network	Deep neural network	Maize	(Khaki and Wang, 2019)
5	Random forests	Random forests	Mango	(Fukuda et al., 2013)
6	Multivariate adaptive regression splines, SVM, random forests, extremely randomised trees, ANN, deep neural networks	Deep neural network	Soy bean, maize	(Kim et al., 2019)
7	CNN, DNN, regression trees, ridge regression	CNN	Soybean	(Russello, 2018)
8	K-nearest neighbour, boosting, SVM, ridge regression	Boosting	Wheat, potato, barley, beet, sunflowers	(Paudel et al., 2021)
9	MLR, ANN	ANN	Apples	(Balducci, Impedovo, and Pirlo, 2018)
10	ANN	ANN	Apples	(Cheng et al., 2017)
11	MLR	MLR	Apples	(Kaack, Pedersen, et al., 2010)
12	MLR	MLR	Apples	(Linker, 2018)

Finally, existing literature on apple prediction have concentrated on medium to late yield prediction (Cheng et al., 2017). In particular, studies that perform yield prediction based on image detection are constrained to wait for apple formation. There are many benefits to using this approach, because prediction accuracy increases as the time to harvest decreases (Stajnko and Švagan, 2009). However, due to the capital-intensive nature of apple farming, early forecasts (4 to 6 months in advance) are important for budgeting and planning purposes and can therefore benefit growers.

2.4 Conclusion

The knowledge built in this chapter acts as a solid foundation for the remainder of the study. This study can be seen at an intersecting point between various research domains, namely crop yield prediction, the agriculture digital revolution, machine learning, and apple crop management. Figure 2.2 provides a conceptual view of these fields of study.

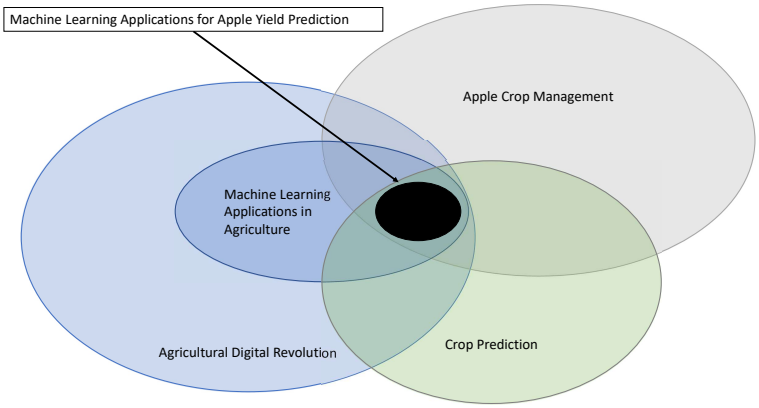


FIGURE 2.2: Conceptual overview of the literature review showing the different study areas and the positioning of the research topic.

The chapter started with a broad overview of the digital revolution in agriculture, helping to contextualise the study. Thereafter, a description of apple crop management was provided that included topics such as orchard design, orchard management, and environmental factors. This is essential for the choice and understanding of the predictor variables that are discussed in Chapter 3. Finally, crop yield prediction was discussed. This discussion was centred around previous studies in the field and is an important building block for the methodology discussed in Chapter 4. Before expanding on the methodology applied in this study, the next chapter describes the data.

Chapter 3

Data

This chapter covers a description and overview of the data. First, background to the data is provided and thereafter the selection of variables is discussed. This is followed by the data collection and preparation and, finally, an overview of the data is provided. The overview includes a summary of the data and trends analysis.

3.1 Background

This study uses apple production data from 2016 until 2020. The dataset includes data from 745 orchards and there are 2800 observations in total. For the purpose of this study, an orchard is defined as a block of trees that is in a demarcated geographical region and share similar properties. The production of the apple trees are grouped, measured and recorded on an orchard level. Each orchard consists of one apple variety, has the same plant date, and vary in size ranging from 0.2 to 11.27 hectares.

An observation is comprised of many independent variables and one dependent variable. The independent variables – properties that potentially impact the production of the apple orchard – are grouped into the following categories according to the type of data: production data, climate data, historic yield, and secondary data. The dependent variable is the annual apple production of the orchard measured in tonnes per hectare.

The data are collected from farmers in the Western Cape province of South Africa, which is the main apple producing region in the country contributing to approximately 80% of South Africa's total apple production. Due to confidentiality and the need for the data to remain anonymous, further details regarding the farms are not revealed.

Note that the prediction time-frame of the model impacts the type of data that can be included in the dataset. In South Africa, the approximate apple harvesting season is from February until May. The prediction time frame for the model is medium-term (approximately 4 to 6 months), which means that the yield is predicted in the preceding October before the harvest. The implication is that only data after the previous harvest and before the end of October can be taken into consideration.

3.2 Variable Description

In total, there are 21 independent variables and 1 dependent variable in the dataset. The variables are selected based on literature and previous studies, but also by taking into consideration the prediction time-frame and data availability. The dependent variable is the production yield of the orchard for a specific harvesting season. Due to the fact

that orchards vary in size, the yield is measured in tonnes (metric ton) per hectare.

Table 3.1 shows a summary of the independent variables. It includes the variable group, variable name, and a description. The dataset also includes other variables that are only used for descriptive purposes. Examples include the orchard ID, production year, and size of the orchard.

TABLE 3.1: Summary of independent variables including the variable group, name and description.

Group	**Variable Name**	**Description**
Production Data	FARM	The geographical area where the orchard is located.
	CULTI_GRP	The grouping of the cultivar.
	AGE	The age (in years) of the orchard based on the planting date of the trees.
Historic Yield	HIST_2_YEAR	The average yield for the orchard for the previous two years (Ton/Ha).
	HIST_1_YEAR	The previous year's yield for the orchard (Ton/Ha).
Climate Data	COLD_UNITS[2]	The number of cold units during the preceding Winter (May - August) before harvest.
	PRECIP	Total rainfall from 1 May until 31 October for the year preceding harvest (mm).
	GDD	Number of Growing Degree Days (GDD) for the preceding September and October to harvest.
	AVG_T_SEP	Average temperature for the September preceding harvest (degree Celsius).
	AVG_T_OCT	Average temperature for the October preceding harvest (degree Celcius).
	AVG_MAX_SEP	Maximum temperature for the September preceding harvest (degree Celsius).
	AVG_MAX_OCT	Maximum temperature for the October preceding harvest (degree Celsius).
	AVG_MIN_SEP	Minimum temperature for the September preceding harvest (degree Celsius).
	AVG_MIN_OCT	Minimum temperature for the October preceding harvest (degree Celsius).
	SUNH_SEP	Number of sun hours for the September preceding harvest.
	SUNH_OCT	Number of sun hours for the October preceding harvest.
Secondary Data	CCROP	Cover crop used in the orchard.
	PLANT_DIR	Plant direction of trees.
	ROOTSTOCK	Rootstock of apple trees.
	SPACING1	Between row spacing (m).
	SPACING2	Between tree spacing (m).

[2] Cold units refer to a metric that is used to measure a plant's exposure to chilling temperatures.

3.3 Data Collection and Preparation

The data are sourced from various sources, such as production and weather station reports, and formatted for further analysis. Only data related to commercial apple orchards are included, which means that trees used for cross-pollination are excluded from the study.

The following issues are identified in the original data that require specific action in order to prepare the data for analysis:

- Some categories within certain categorical variables were inconsistently spelled and needed to be standardised.
- The age of the trees needed to be derived from the planting and production date.
- Due to the sensitive nature of the data, certain variables needed to be changed to codes so that the data can remain anonymous.
- In order to ensure that each orchard shares the same properties, the orchard ID together with the cultivar and plant date needed to be used as a unique identifier for each orchard.
- The climate data are sourced from four weather stations. Each orchard is located within a farm and each farm is mapped to the closest weather station (with a maximum distance of 10km between farm and station) to obtain weather data per orchard.
- The climate data needed to be summarised and formatted before it could be used in the model.
- Some orchards are newly planted and therefore do not have historic yield data. The historic yield for these orchards are assumed to be 0.

The aim of the data preparation is to format the data into the desirable format for further analysis, which is conducted in Chapter 5. Table 3.2 shows a summary of the variables and types after preprocessing.

3.4 Data Summary

Prior to analysing and modelling data, it is good practice to gain a better understanding of the data by means of exploration. As mentioned in Section 2.2, the yield of an apple tree depends on a number of factors and differs per season. Table 3.3 shows the summary statistics for the yield variable in the dataset. The table includes common measures of location and statistical dispersion such as the minimum, median, mean, maximum, and interquartile range. The metrics suggest that there is a wide spread in yield, ranging from 0 to 155.98 tonnes per hectare.

Another way to better understand the yield characteristics is to look at the shape of the distribution of the yield, as shown in Figure 3.1. The large amount of orchards with a yield close to 0 are attributable to a notable proportion of young orchards in the dataset that are not yet in full production. If the young orchards are excluded, the distribution seems approximately bell-shaped with a slight upper tail. The large proportion

TABLE 3.2: Variable name and variable type after preprocessing

	Variable	Type
1	YIELD	numeric
2	CULTI_GRP	Factor w/ 15 levels
3	AGE	numeric
4	FARM	Factor w/ 7 levels
5	HIST_2_YEAR	numeric
6	HIST_1_YEAR	numeric
7	COLD_UNITS	numeric
8	PRECIP	numeric
9	GDD	numeric
10	AVG_T_SEP	numeric
11	AVG_T_OCT	numeric
12	AVG_MAX_SEP	numeric
13	AVG_MAX_OCT	numeric
14	AVG_MIN_SEP	numeric
15	AVG_MIN_OCT	numeric
16	SUNH_SEP	numeric
17	SUNH_OCT	numeric
18	CCROP	Factor w/ 11 levels
19	PLANT_DIR	Factor w/ 7 levels
20	ROOTSTOCK	Factor w/ 10 levels
21	SPACING1	numeric
22	SPACING2	numeric

TABLE 3.3: Summary statistics for the yield dependent variable

	Value
Min.	0.00
1st Qu.	35.51
Median	56.82
Mean	55.90
3rd Qu.	77.40
Max.	155.98

of non-bearing orchards is foreseen to have an impact on the modelling, especially the regression model, and therefore care should be taken in the treatment of the zero-yield values when the models are applied.

The correlation between two variables is an indication of the strength of the linear relationship between the variables. Refer to Figure 3.2 for the correlation between all the numeric variables in the dataset. The plot shows that there is a strong positive correlation between the yield and historic yield. Interestingly, the relationship between the yield and 2-year historic yield is stronger than the relationship between the yield and 1-year historic yield. As expected, there is some positive and negative correlation between climate indicators. Furthermore, there is a relatively strong correlation between the age and yield, precipitation and yield, and also between the age and spacing.

The relationship between the yield and age is further analysed in the next section. The correlation between the age and spacing possibly suggest a change in orchard architecture

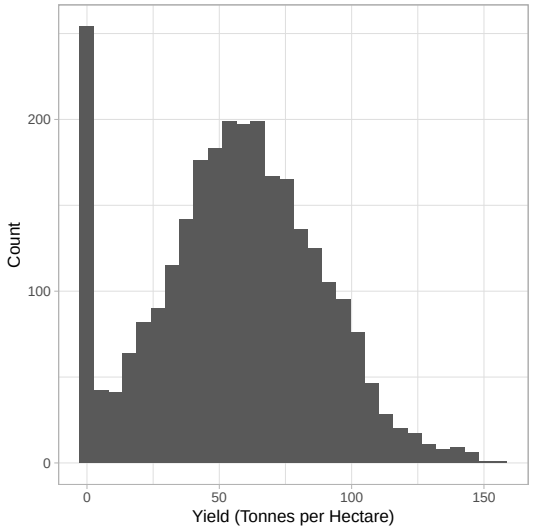


FIGURE 3.1: Histogram showing the distribution for the yield variable measured in tonnes per hectare.

over time. The fact that there is a strong positive correlation between the 1-year and 2-year historic yield is expected, since the one is a function of the other. The lack of correlation between the predictor variables and the yield indicates that a linear model might struggle to predict the relationship between input and output. It is, however, not an issue for the application of the non-linear models.

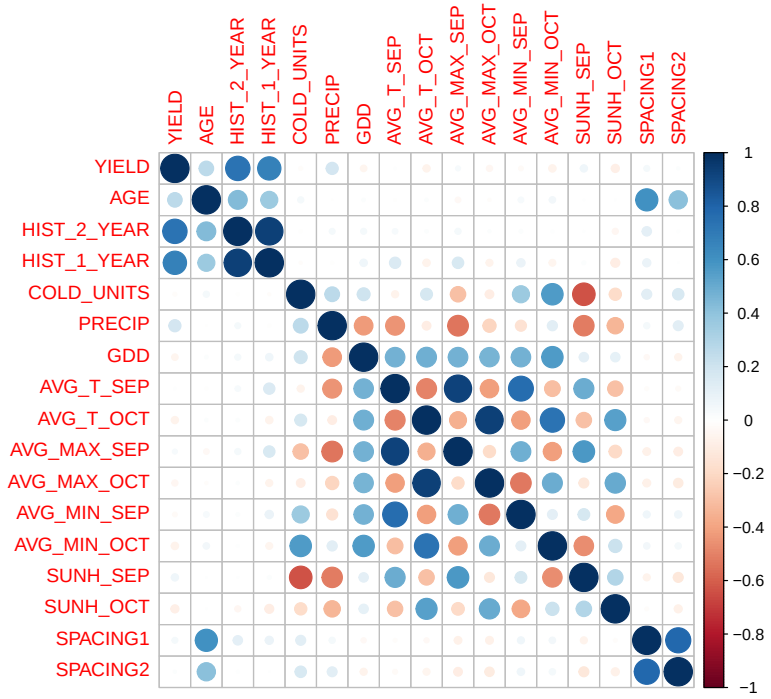


FIGURE 3.2: Correlation diagram showing the correlation between the numeric variables

3.5 Data Trends and Patterns

Another way to obtain a better understanding of the data is to look at trends and patterns. Seeing as the study is concerned with the prediction of yield, it is useful to look at the variability in yield across a number of different factors. All the independent variables have a potential impact on yield. However, in this section the focus is on the main factors, namely the age of the trees, the production year, cultivar type, and geographical location of the orchard. The climate variables are indirectly encapsulated as part of the production year.

Firstly, the relationship between the yield and age of the tree is analysed. This is depicted in Figure 3.3, showing the mean apple yield per age. The plot shows an increase in the average yield for the first 10 years, followed by a productive phase until the age of 20 and a very gradual decline thereafter. This general trend corresponds to the trends for apple tree production in the literature (mentioned in Section 2.2), although most studies consider trees full-grown at age 7 and do not highlight a highly productive period between 8 and 12 years as noticed in Figure 3.3.

When looking at the yield per age, it is also important to consider the distribution of samples across age groups. The average age of the orchards in the dataset is 20 years and 75% of the orchards are younger than 28 years. It seems that the drop in average yield at 17 years of age is due to a small sample (n = 7) at this age, rather than a general trend. The sample at 10 years includes 370 orchards and is considered representative. However, at this stage, it is not clear whether the spike around year 10 is due to the age of the tree or due to other factors like the cultivar mix and above-average seasons for the trees included in the samples. The fact that there is a clear period of incline in yield as the trees reach maturity could be problematic for a linear regression model and is further discussed in Section 5.1.

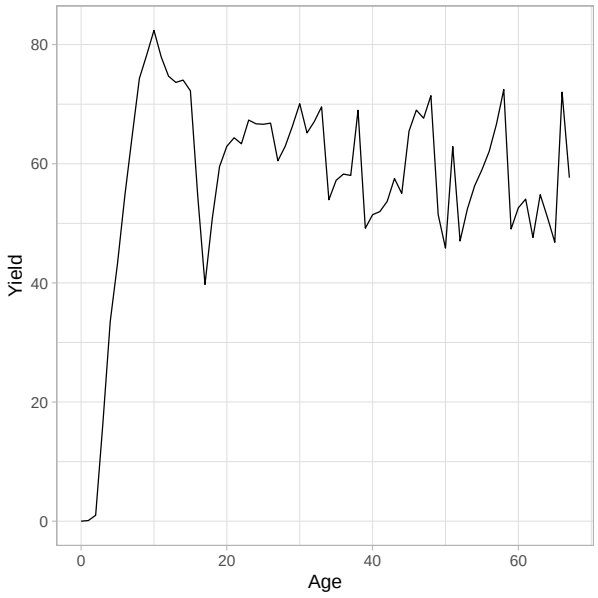
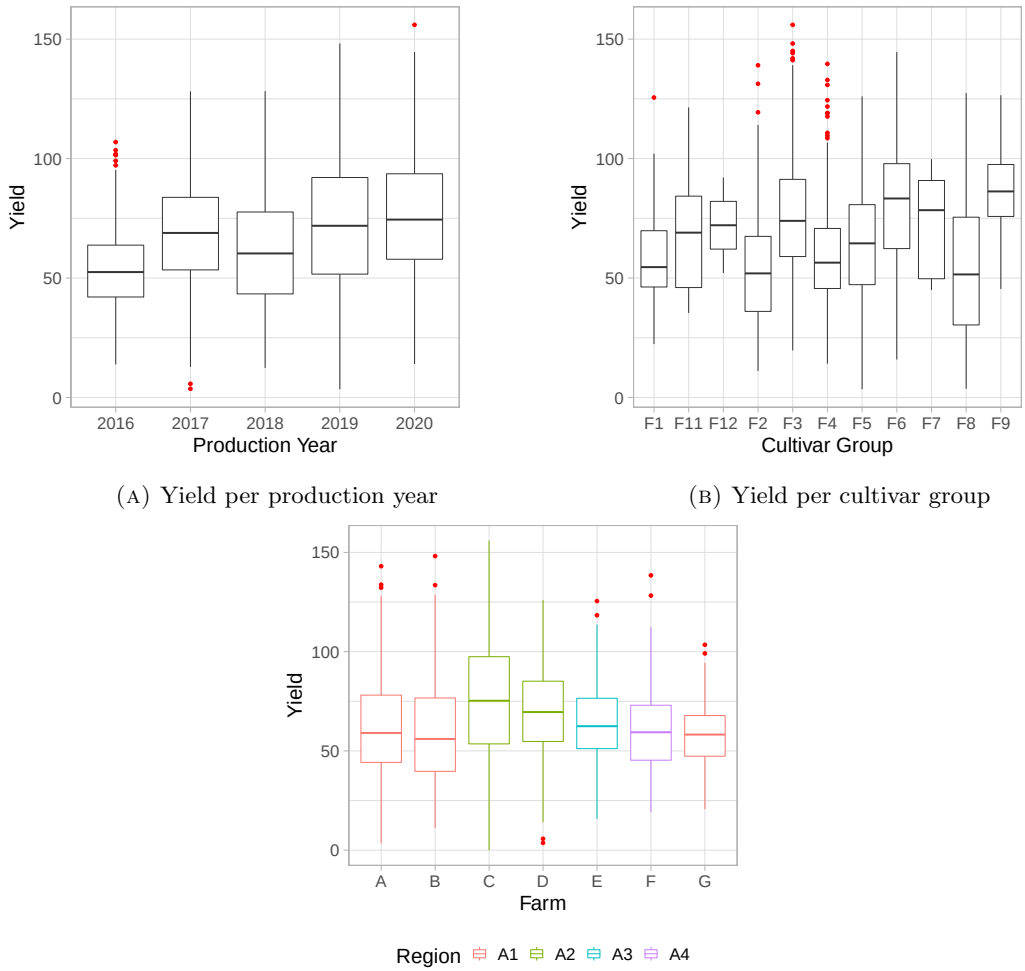


FIGURE 3.3: Average yield per age of trees

Secondly, the relationship between the yield and the production year, cultivar group and farm is considered. Figure 3.4 shows multiple boxplots that show the variability in yield

over these factors. Note that only mature orchards (older than 6 years) are included due to the large impact of young orchards' yield that is much lower than mature trees. Even though the data shows an incline in yield until 10 years, the average yield at 7 years is comparable to the average yield of mature trees and, therefore, the trees are considered mature from 7 years onwards. This corresponds to Goossens et al., 2017's definition of mature apple trees according to age.



(A) Yield per production year

(B) Yield per cultivar group

(C) Yield per farm

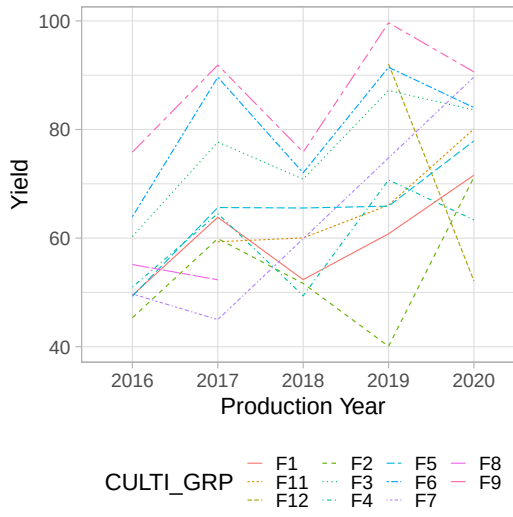
FIGURE 3.4: Boxplot statistics to show the variability in yield

Figure 3.4a shows how the yield of apple trees vary depending on the production year. It shows that the median yield fluctuates each year and also that the variability of yield differs each year. This is impacted by the independent variables that differ annually (such as the climate factors) and also by production practices (e.g. pruning and fertilization). In addition, the average yearly harvests depend on the orchards that are included in that production year. New orchards are planted each year and unproductive orchards are taken out, which means that the age distribution of the trees and the cultivar mix differ yearly. The impact of the age distribution is minimised in the plot by only considering mature trees, but the influence of the cultivar mix needs to be taken into consideration.

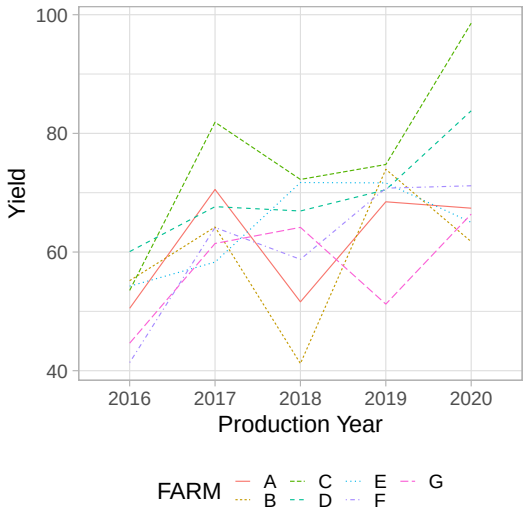
The variability in yield across cultivar groups is shown in Figure 3.4b. It is clear from the plot that both the median yield as well as the spread of yield varies across cultivar groups. New cultivars are continually developed and apple farmers aim to progressively move to cultivars that are more profitable. A higher potential yield is one of the main drivers for profitability.

Figure 3.4c shows the variability in yield across various farms. The climatic region, which is derived from the farm's closest weather station, is also shown by the colour of the boxes. Farms within the same climatic region are approximately within a radius of 20km from each other. As expected, due to the large impact of climatic factors, farms within the same region tend to deliver similar yield even though their cultivar mixes may differ.

As mentioned, apple yield is affected by the interplay of a number of factors and viewing only two factors at a time can be misleading. Figure 3.5 shows how the average yield varies per cultivar group, farm and production year. The figures show that the longitudinal trends differ depending on the cultivar group and farm. Note that again only mature trees are included to minimise the effect of the age distribution. Figure 3.5a shows how the yearly trends differ depending on the cultivar group. It is also evident from the plot that cultivar group F8 was discontinued in 2017 and that a new cultivar group was introduced in 2017 and 2019. Figure 3.5b shows that the yearly trends also differ largely depending on the farm. For example: Farm C's yield peaked in 2020 while Farm B's yield was the highest in 2019.



(A) Average yield per production year and cultivar group



(B) Average yield per production year and farm

FIGURE 3.5: Variability in yield per production year, cultivar group and farm

The trend analysis highlights that apple production yield is affected by an interplay of many factors. Visualising the data is helpful, but it is difficult to visualise more than three factors together. Therefore, more advanced machine learning techniques are required to both make sense of the data and to be able to predict yield.

3.6 Conclusion

In this chapter, the data were prepared, summarised and explored. The aim was to prepare the data for analysis and to gain a better understanding of the data through exploration. First, the data were introduced by defining the variables and discussing the background thereof. Thereafter, the data were summarised using summary statistics and the distribution of the dependent variable was explored. This was followed by looking at the relationship between the features and, finally, the trends were analysed using two and three factors concurrently.

This chapter provides a sufficient understanding of the data and acts as a solid foundation for further analysis in Chapter 5. The next chapter, Chapter 4, introduces the methodology that will be applied when the data are analysed in Chapter 5.

Chapter 4

Methodology

In Chapter 2, relevant literature studies for the prediction of apple yield were explored whilst Chapter 3 provided an overview of the dataset to be used for analysis. This chapter gives an outline of the methodology that is used during the application of the machine learning methods to predict apple yield. First, the modelling approach is discussed, which includes the data split, methods applied, and evaluation criteria. This is followed with a technical overview of Multiple Linear Regression (MLR), Artificial Neural Networks (ANNs) and tree-based methods. The aim of this chapter is to provide a solid foundation and reference point for the application of the methods, described in the next chapter.

4.1 Modelling Approach

The following approach is followed to select the most appropriate model for apple yield prediction. First, the data is split into a training and test set. Second, a variety of machine learning methods are applied to build a suitable prediction model using the training dataset. Lastly, the final models for each of the methods are evaluated using the test set. These three phases are depicted in Figure 4.1 and discussed in further detail below.

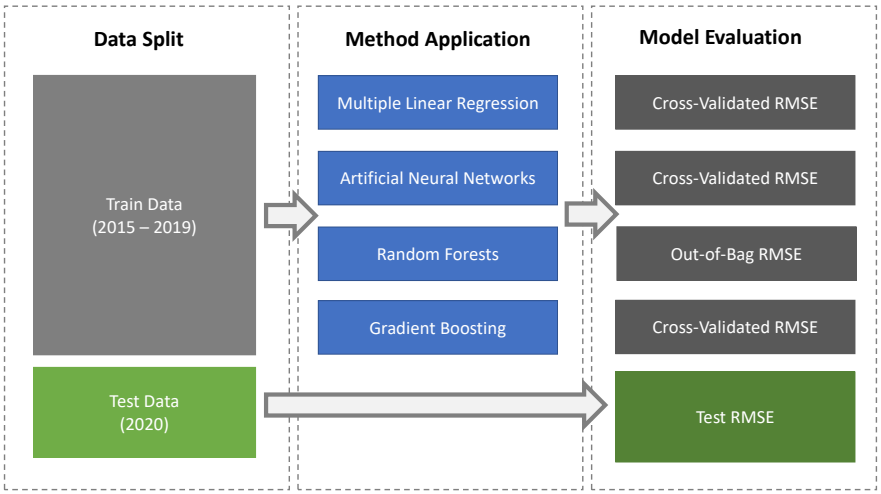


FIGURE 4.1: General approach for the application of machine learning methods

Phase 1: Data Split

During the first phase, the data are split into a training and test dataset. The training set is used to build the models and optimise hyperparameters, and the test set is kept aside and used to measure the model's ability to generalise. This split is crucial, because the generalisation error (measured using the test set) determines the overall quality of the model (Bruce, Bruce, and Gedeck, 2020). There are two common ways to split the data into a training and test set – it can be done randomly, or the most recent data can be kept aside as the test set. In this study, both approaches are used and the results compared.

First, the latest season's data (2020) are kept aside as the test set and data from 2016 to 2019 are used as the training set. This is similar to the approach followed in other yield prediction studies (Khaki and Wang, 2019; Paudel et al., 2021), and it also corresponds to how the model would be used in agricultural applications. This split is referred to as the 'chronological split' in the remainder of the study.

Secondly, the models are retrained and optimised using a random split to check if the model is able to generalise over different seasons. For this split, the well-known 80/20 split will be used whereby 80% of the data constitutes the training set and 20% the test set (James et al., 2013). This split is referred to as the 'random split' during the remainder of the study.

Phase 2: Method Application

The second phase includes the application of four machine learning methods: MLR, ANN, random forests and Gradient Boosting (boosting). MLR is included as a base model to compare performance. ANN, random forests and boosting are chosen because they have shown superior performance in crop prediction when compared to other models in studies referenced in Section 2.3 (Paudel et al., 2021; Balducci, Impedovo, and Pirlo, 2018; Cheng et al., 2017). The details of the methods are discussed in Sections 4.2, 4.3 and 4.4 of this chapter.

Learning, or training, is a fundamental capability of machine learning models. During this phase, the models have access only to the training set and learn by estimating certain parameters to minimise the error function. Many models depend on hyperparameters that need to be pre-specified when the model is trained. This phase also includes the choice of hyperparameters that are applicable to each of the methods.

Phase 3: Model Evaluation

The third phase involves the evaluation of the models. It is critical to use the same evaluation criteria to compare the performance of different models. The following measures are commonly used for regression analysis: Mean Squared Error (MSE), Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). The advantage of RMSE and MAE over MSE is that they measure the error in the same units as the base unit and are therefore more easily interpretable. In this case, RMSE is preferred and will be used to evaluate the models because this is a common measure in similar studies and also viewed by some as the most widely used metric for regression models (Bruce, Bruce, and Gedeck, 2020).

The formula to calculate the RMSE is shown in Equation 4.1 where n refers to the number of observations, f_i refers to the prediction of the i^{th} observation and o_i refers to the actual value for the i^{th} observation (James et al., 2013). In summary, it provides an indication of the average difference between the predicted and actual value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(o_i - f_i)^2} \tag{4.1}$$

The RMSE value is specific to the type of data that is used in the calculation. In this study, the following types of RMSE values are referred to: Train RMSE, Cross-Validation (CV) RMSE, Out-of-Bag (OOB) RMSE and test RMSE. The train RMSE is calculated using the data that are used to train the model. The CV and OOB RMSE values are used during hyperparameter tuning. These approaches rely solely on the training set and are used to estimate the test error. Table 4.1 shows which approach is used for which method in this study, because certain approaches are more suitable for some methods than others.

TABLE 4.1: Evaluation approach used per method during hyperparameter tuning

Method	**Evaluation Approach**
Multiple Regression	Cross-Validation
Neural Networks	Cross-Validation
Random Forests	Out-of-Bag
Gradient Boosting Machines	Cross-Validation

In this study CV refers specifically to k-fold cross-validation, which is a resampling procedure that involves randomly splitting the dataset into K groups (or folds) of approximately equal size. Each of the K groups then gets a turn to act as the validation set while the model is fitted to the remainder of the groups (*K-1*). The process is repeated K times and results in K estimates of the test error computed using the validation set. The overall error is the average of these estimates. Common choices for K are 5 and 10 because these values have shown to provide good estimates for the performance of models on an unseen dataset (James et al., 2013). Unless stated otherwise, this study uses $K = 10$. The OOB error is available only for bagging and random forest models due to the inherent nature of the models, described in detail in Section 4.4.

An important concept in machine learning is the bias-variance trade-off that refers to the dilemma of finding a balance between bias and variance to minimise the total error. Bias refers to the erroneous assumptions in the learning algorithm calculated by measuring the expected deviation from the true value of the function (Bengio, Goodfellow, and Courville, 2017). The variance of a model refers to the amount by which the estimated values will change if it was estimated using a different training set. A common term for a model that struggles to generalise is overfitting. Models that overfit have high variance and low bias. Underfitting, on the other hand, describes a model that does not depict the patterns in the training data well enough, resulting in low variance and high bias.

The concept of overfitting and underfitting is illustrated in Figure 4.2. The typical U-shape of the test error is due to the competing forces of bias and variance. As the model's capacity or complexity increases, the test error typically decreases up to a point

and then starts to increase when the model is starting to overfit on the training data. The difference between the test error and train error is commonly referred to as the generalisation gap (James et al., 2013).

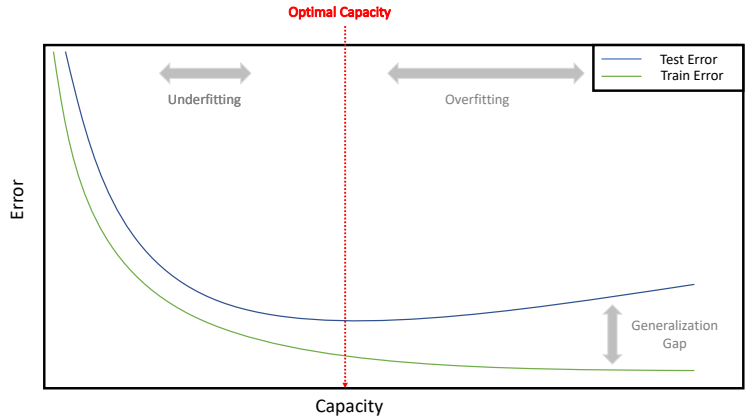


FIGURE 4.2: Typical relationship between the capacity of the model, and test and training error. When the model's capacity is increased beyond a certain point, the training error increases, which leads to overfitting and a bigger generalisation error (Bengio, Goodfellow, and Courville, 2017).

The aim of model selection is to find the simplest possible model that has a sufficient fit to the data and generalises well, and therefore it is critical to keep the bias-variance in mind during model evaluation (Du and Swamy, 2013). For this reason, care is taken to optimise parameters and hyperparameters using only the training set with appropriate error estimation techniques such as CV and OOB. The final evaluation, to determine the overall quality of the models, is based on the test RMSE. This is also the metric used to compare the performance of the final models obtained by means of each of the four methods.

4.2 Multiple Linear Regression

Multiple Linear Regression (MLR) is a statistical technique that uses a variety of explanatory variables to predict the outcome of a response variable. According to Kutner et al. (2005), it is one of the most widely used statistical methods. The aim of this technique is to obtain a model that predicts as much variation in the response variable as possible using the values of the predictor variables (James et al., 2013). This section provides an overview of MLR.

4.2.1 Model Specifications

The general form for a multiple linear regression model is depicted by (Kutner et al., 2005):

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + ... + \beta_p X_{ip} + \varepsilon_i \tag{4.2}$$

where:
$\beta_0, \beta_1, ..., \beta_p$ are regression parameters
$X_0, X_1, ..., X_p$ are observed constants
ε_i are independent $N(0, \sigma^2)$

$i = 1, ..., n$
n is the number of observations

More specifically, p represents the number of predictor variables, X_j indicates the j^{th} predictor variable, β_j is the regression coefficient that quantifies the relationship between the predictor variable and the response, and ε is the residual representing the difference between the predicted and the observed value. Each regression coefficient (β_j) can be interpreted as the change in Y relative to a one unit increase in X_j, holding all other predictors fixed (Kutner et al., 2005).

The simplified goal of regression analysis is to find the coefficients of the predictor variables. This is done by minimising the Residual Sum of Squares (RSS) loss function. Using the same notation as for RMSE, the function is defined as:

$$RSS = \sum_{i=1}^{n} (o_i - f_i)^2 \tag{4.3}$$

4.2.2 Model Fit

MLR models are trained to fit a dataset. The extent to which the model fits the data can be quantified using the R^2 statistic, also referred to as the coefficient of determination. It provides a relative measure of fit independent of the scale of the data, calculated as:

$$R^2 = 1 - \frac{RSS}{TSS} \tag{4.4}$$

where TSS refers to the *Total Sum of Squares* calculated as the sum of the squared difference between the observed value and the mean observed value in the dataset. In simple terms, the R^2 statistic can be interpreted as the proportion of variation in the response variable that is explained by the predictor variables. The range for the R^2 statistic is between 0 and 1, where 0 is a bad fit and 1 is a perfect fit. The adjusted R^2 statistic is similar, but it penalises complex models (measured by the number of predictor variables) and is therefore useful to compare models with different sets of variables.

4.2.3 Hypothesis Testing

A common question during regression analysis is whether there is a relationship between the response and at least one of the predictor variables. This is answered by testing the following hypotheses (James et al., 2013):

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0$$

$$H_A : \text{At least one } \beta_i \neq 0 \text{ where } i = 1, \cdots, p$$

In other words, the null hypothesis states that there is no relationship between the predictor variables and the response variable indicated by a zero regression coefficient. The null hypothesis is tested using the F-statistic, given by:

$$F_{stat} = \frac{(TSS - RSS)/p}{RSS/(n - p - 1)} \tag{4.5}$$

which, under the null hypothesis, follows an F-distribution with p and $n - p - 1$ degrees of freedom:

$$F_{stat} \sim F_{p,n-p-1}$$

Therefore, a large F-statistic reflects a large discrepancy between TSS and RSS, indicating that there is a relationship between at least one of the predictor variables and the response. The p-value – the probability of observing an F-statistic at least as extreme as F_{stat} under H_0 – is used to quantify the statistical evidence against H_0. A small p-value indicates that there is compelling evidence against H_0, meaning there is likely to be a relationship between at least one of the predictor variables and the response variable (James et al., 2013).

Similar hypothesis testing can be used to determine the significance of the relationship between each of the individual predictor variables and the response variable. In this case, the null hypothesis is that there is no relationship between the specific predictor variable and the response. The evidence against this null hypothesis is determined using the t-statistic, which is a metric used to compare the importance of variables in the model by measuring the extent to which a coefficient is far from 0 (Bruce, Bruce, and Gedeck, 2020). It is used in conjunction with its associated p-value, which again refers to the probability of observing a sample yielding at least such an extreme test statistic, under the condition that H_0 is true.

The smaller the p-value, the more unlikely it is to observe such an observation and the more evidence there is to reject H_0. In contrast, a big p-value indicates that it is likely to observe such a value and acts in support of H_0. The exact p-value cut-off for rejecting H_0 depends on the study. Commonly used cut-offs are 1% and 5% (James et al., 2013). It is important to note, when testing the null hypothesis that there is no relationship between the specific predictor variable and the response, the p-value shows only the partial affect of the variable given that all other variables are held constant. Therefore, variable selection should not be done based on individual p-values, but rather using specially developed variable selection techniques.

There are many different types of variable selection techniques. Regularisation techniques, such as lasso or ridge regression, can be used to shrink the coefficient estimates towards zero by penalising complex models. Lasso regression is particularly useful, because it has the ability to shrink coefficients to absolute zero. With this technique, the cost function changes to the following:

$$\text{Cost Function} = RSS + \sum_{j=1}^{p} |\beta_j| \tag{4.6}$$

Another variable selection technique is backward stepwise selection, which entails starting with a full model and then removing the least useful or significant variables one-by-one. Forward stepwise selection is similar, but instead of starting with a full model this method starts with an 'empty' model containing only the intercept and no predictor variables, and then adds the predictor variables one-by-one. The choice of predictor variables depends on the biggest improvement to the model.

The advantages of using MLR are that it is simple, easy to compute, and interpretable. The disadvantages are that it depends on a number of assumptions – such as linear relationships between the predictor and response variables; constant variance and independence of the residuals; and independence of predictor variables – that are often

not met by real-life problems. In this specific case of apple yield prediction, the data is skewed with a large number of zero values as illustrated in Figure 3.1. Alternative measures to address this when applying a MLR model are discussed in the next section.

4.2.4 Skewed Data Approaches

The fact that there is a notable proportion of orchards bearing no yield in the dataset is a cause of concern when applying a MLR model. This type of skewed data with a large number of zeros, also referred to as 'clumped at zero' or 'zero-inflated', is a common occurrence and not unique to apple yield prediction. Many studies have been done to find alternative ways to deal with this issue. A short summary of approaches are provided in this section. Seeing as the MLR model is used as a base model for comparison to non-linear models, a comprehensive review on dealing with the issue of skewed data in the context of MLR is out of scope for this study.

Min and Agresti (2002) surveyed models that have been proposed to cater for this type of data. The models are roughly grouped into the following types: tobit models, two-part models, sample selection models, compound poisson exponential dispersion models, and ordinal threshold models. The study concludes that a two-part model is a reasonable choice for numerous applications because it is simple to fit and also to interpret. An example of a two-part model is Duan et al. (1983)'s model that separates the modelling into two stages using two different equations. The first stage models whether the response is positive and the second the magnitude of the response.

This concludes the overview on MLR. The application of this approach in the context of apple yield prediction is discussed in Section 5.1.

4.3 Neural Networks

Artificial Neural Networks (ANN) are computational networks that are based on the functioning of the biological nervous system. It consists of multiple processing units that receive inputs and convert it to outputs according to a predefined function. These units are typically referred to as neurons, nodes or perceptrons and are connected to each other in a network structure (Smithing, 1999). Neural networks are applied for a variety of problems such as classification, pattern recognition, prediction, image analysis, and data mining.

There are many different types of neural networks. One of the most common types, and also the focus of this study, is a Multi-Layer Perceptron (MLP) network that is an ANN consisting of an input layer, one or more hidden layers, and an output layer (Smithing, 1999). The basic architecture of an MLP network (for a regression problem with one hidden layer) is shown in Figure 4.3. This type of network is used to learn non-linear functions and is a type of feedforward network with a one-way flow of information from the input to the output layer (Bengio, Goodfellow, and Courville, 2017). The input layer is used to receive external signals that are fed to the system. The hidden layer consists of neurons that are fully connected to the neurons in the input and output layers by means of a vector of parameters called weights. The output layer shows the output of the system. For regression analysis, it gives an estimate for the response variable.

The network in Figure 4.3 consists of p input neurons, m neurons in the hidden layer, and one neuron in the output layer shown as y. The input layer (x_i where $i \in \{1, .., p\}$)

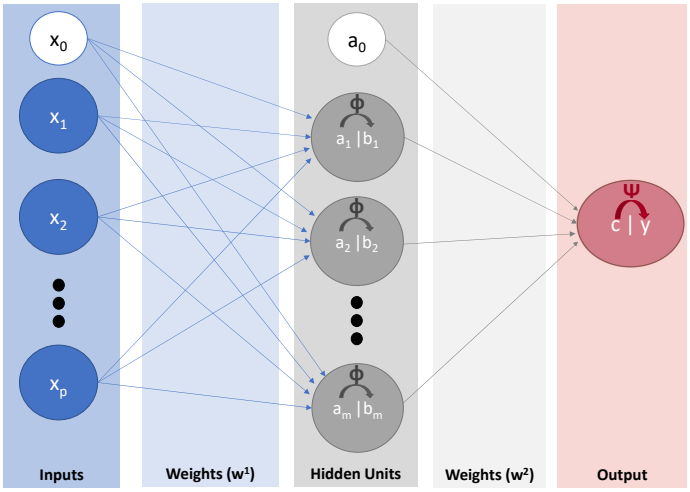


FIGURE 4.3: Basic architecture of an MLP network. The neurons in the input, hidden and output layers are represented by nodes and the weights are shown by the arrows connecting the nodes. The bias parameters are represented by x_0 and a_0 (Bishop, 2006).

is connected to the hidden layer by means of a vector called weights (w^1). The weight connecting unit i in the input layer to unit j in the hidden layer is denoted as w^1_{ij}, where $j \in \{1, .., m\}$. Similarly, the weights connecting the hidden layer to the output layer is referred to as w^2, where w^2_j refers to the weight connecting the j^{th} neuron in the hidden layer to the output layer. The flow into the hidden layer is denoted as a_j and the outflow, after applying the $\phi(.)$ activation function, is denoted as b_j. The inflow into the last layer is denoted as c and the output of the model, after applying the $\psi(.)$ activation function, is y. The bias parameters for the input and hidden layer are denoted as x_0 and a_0 respectively.

The input into the j^{th} neuron in the hidden layer (a_j) is a weighted linear combination of the input layer plus a bias term, constructed as follows:

$$a_j = \sum_{i=1}^{p} w^1_{ij} x_i + w^1_{0j} \tag{4.7}$$

The output from the j^{th} neuron in the hidden layer (b_j), is calculated by transforming a_j with an activation function $\phi(.)$:

$$b_j = \phi(a_j) \tag{4.8}$$

Similarly, the final output is the output from the hidden layer, transformed by the activation function in the final layer, given by:

$$y = \psi \left(\sum_{i=1}^{m} w^2_j b_j + w^2_0 \right) \tag{4.9}$$

The overall output from the various stages can be combined into the following function:

$$y = \psi\left(\sum_{i=1}^{m} w_j^2 \phi\left(\sum_{i=1}^{p} w_{ij}^1 x_i + w_{0j}^1\right) + w_0^2\right) \tag{4.10}$$

In summary, a neural network for a regression problem is the conversion of input variables (x_i) to a response (y) using a non-linear function that is controlled by a vector w of adjustable parameters (Bishop, 2006).

4.3.1 Neural Network Components

Neural networks consist of many components that are modelled as hyperparameters which need to be prespecified when training a model. As a result, many decisions need to be made during the design of a neural network. These decisions are typically made by means of trial and error, where hyperparameters are iteratively tuned and the performance evaluated. The various components are discussed in separate sections below.

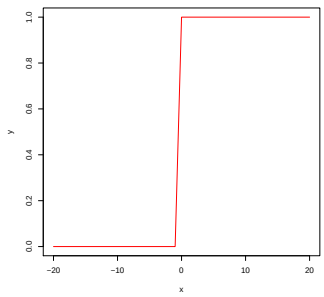
Network Topology

The network topology refers to the structure in which neurons are connected to each other. When designing a neural network, the number of layers and also the number of neurons in each layer need to be specified. MLP networks are generally organised into three types of layers – an input layer, one or more hidden layers, and an output layer. The number of neurons in the input and output layers are dependent on the structure of the data. The number of neurons in the hidden layer is typically decided by means of trial and error. This influences the complexity of the model – more neurons lead to higher complexity and an increased number of parameters (weights) that need to be estimated. The network topology is important because it has a significant impact on the performance of the model (Bengio, Goodfellow, and Courville, 2017). Too few neurons could lead to underfitting, whereas too many neurons could lead to overfitting and also increase the computational time.

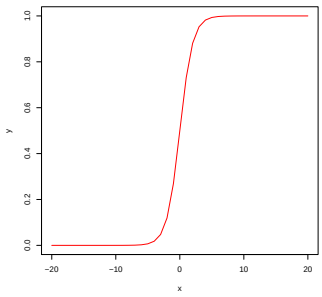
Activation Functions

The activation function is the processing mechanism or function that is used to transform input signals into an output signal at a neuron level. In Figure 4.3, $\psi(.)$ and $\phi(.)$ represent the activation functions which are defined per layer, therefore all units in a layer share the same activation function. Figure 4.4 shows some of the basic activation functions, other than the linear function, that are commonly used for this purpose.

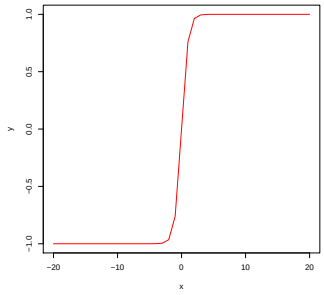
Examples of other activation functions are the leaky ReLU, parametric ReLU, softmax and swish. The choice of activation function depends on the nature of the data and the assumed distribution of the response. It is important that the activation function for the output unit gives an appropriate response for the error function. For regression models, a continuous numerical output is required as a response and therefore, the linear function is a good choice for the activation function in the output layer (Bishop, 2006). The ReLU function can also be used in the case where the response is limited to a positive value. According to Bengio, Goodfellow, and Courville (2017), the ReLU function is the most commonly applied activation function and a good default choice for the hidden layer. Glorot, Bordes, and Bengio (2011) also state that the ReLU function has been observed to perform better than the commonly used sigmoid and hyperbolic tangent functions.



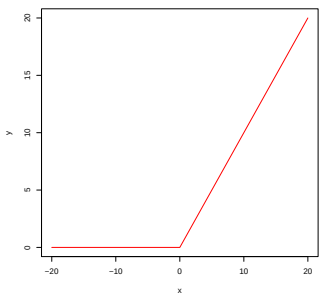
(A) Step function: $f(x) = 0$ if $x < 0$, 1 if $x \geq 0$



(B) Sigmoid function: $f(x) = \frac{1}{1+e^{-x}}$



(C) Hyperbolic tangent function: $f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$



(D) Rectified linear unit (ReLU): $f(x) = max\{0, x\}$

FIGURE 4.4: Basic neural network activation functions

Loss Function

The loss function is the objective function that the neural network aims to minimise during training by updating the weight parameters. The loss function reduces all aspects of the model into a single value that is used as the evaluation criteria when comparing models (Smithing, 1999). A common loss function used for regression is MSE (Bengio, Goodfellow, and Courville, 2017). There is a strong relationship between the activation function and the loss function and therefore a collective decision needs to be made with regards to these two components.

Optimisation Algorithm

Du and Swamy (2013) describe the optimisation algorithm as a set of learning rules that the network uses to find suitable network parameters. More specifically, this is the algorithm that determines how weights are updated to minimise the loss function. Most training algorithms use an iterative process whereby weights are updated during sequential steps. Each step typically consists of a two-stage process. First, the derivatives of the error function with respect to the weights are calculated by means of a process called backpropogation. Thereafter, the weights are adjusted using the derivatives and a prespecified learning rate. The most basic optimisation technique that is used to do this is gradient descent, which involves iterative weight updates by means of small steps in the direction of the negative gradient (Bishop, 2006).

Some of the more advanced adaptive methods include methods such as AdaDelta, AdaGrad, Adaptive Momentum Estimation (Adam) and RMSProp. Adaptive methods offer personalised learning by automatically updating the pace of learning, referred to as the learning rate. An obvious advantage of adaptive learning is that the learning rate does not need to be prespecified and tuned.

Epochs

Neural networks are trained by means of epochs, where one epoch refers to a complete run whereby all the training examples are presented and processed by the learning algorithm once (Du and Swamy, 2013). The general recommendation is to start with a small number of epochs and gradually increase while monitoring performance using a learning curve. A learning curve is a line plot with the epochs on the horizontal axis and the loss on the vertical axis. Refer to Figure 4.5 for an example. The optimal number of epochs is at the point where the loss converges whilst not overfitting. This point can be determined by using early stopping that is explained in Section 4.3.2.

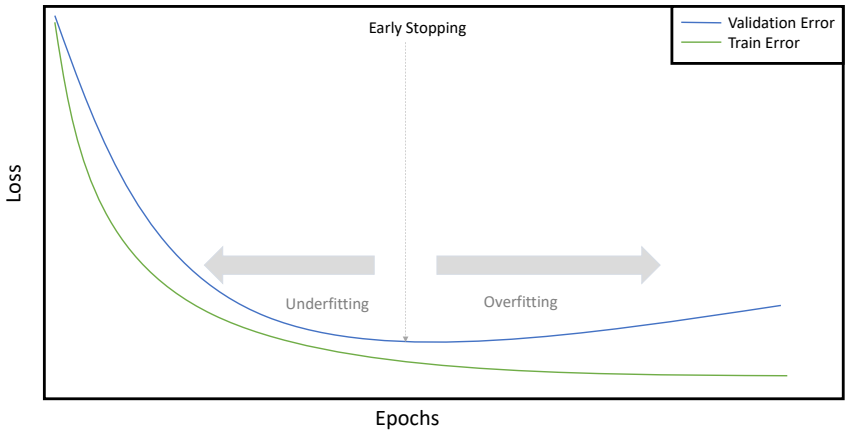


FIGURE 4.5: Illustration of a learning curve that is typically used to monitor the performance of a neural network. The plot shows the validation and test errors over the number of epochs. A typical early stopping point as well as areas of overfitting and underfitting are indicated.

Batch Size

The batch size refers to the amount of samples that are processed during each update iteration of the model. The batch size influences the speed at which the model is optimised. A minimum batch size is 1 and a maximum batch size equals the number of observations in the dataset. A large batch size learns quickly, but does not generalise as well on test data. A small batch size learns slower, but generalises better, which means it performs better on unseen data.

4.3.2 Regularisation

Neural networks are expressive models that can be used to learn complicated relationships between inputs and outputs. However, if training data is limited, these networks are prone to incorporate the noise in the training data, resulting in overfitting (Srivastava

et al., 2014; Bishop, 2006). Regularisation is a collection of techniques that are used to prevent overfitting by penalising complex models. These techniques are commonly used across many machine learning methods and are important because the best-fitting model is often a large model that has been regularised appropriately (Bengio, Goodfellow, and Courville, 2017). Three of the most common regularisation techniques for ANNs are discussed in separate subsections below (Du and Swamy, 2013).

Early Stopping

Early stopping is the most widely used form of regularisation for ANNs (Bengio, Goodfellow, and Courville, 2017). It is popular because it is unobtrusive, effective, and simple to implement. A common pattern observed during the training of a neural network is a U-shaped learning curve for the validation error as illustrated in Figure 4.5. The training error continues to decrease as the number of epochs increase, whereas the validation error typically decreases up to a point and then starts to increase as the network overfits, signifying the U-shape. Early stopping criteria can be used to automatically stop the training of the network when the error with respect to the validation set is at a minimum (Bishop, 2006). This prevents overfitting and also unnecessary computation.

Dropout

Dropout (Srivastava et al., 2014) is based on the theory that model combinations can be used to improve the performance of machine learning models. The key idea is to randomly drop units and their connections from a network during training. Dropout is a popular method because it is a computationally inexpensive and highly effective regularisation method that can be used to prevent overfitting and improve generalisation.

Weight Decay

Another well-known form of regularisation is to constrain the capacity of models by adding a parameter norm penalty to the objective function of the model. The L^2 parameter norm penalty (ridge regression) is one of the most common and simple types of penalisation that encourages weight values towards 0, but not exactly 0. It depends on a hyperparameter α that determines the extent of regularisation and controls the contribution of the norm penalty. For example: If α is set to 0, no regularisation is applied and the original objective function remains unchanged.

This concludes the neural networks overview. The next section explains tree-based methods.

4.4 Tree-Based Methods

Decision trees are powerful methods that model the relationship between features and potential outcomes in a tree-like structure. This section is focused on the application of decision trees to regression problems. The discussion starts with a brief description of a decision tree and then provides an overview of the following advanced tree-based methods: bootstrap aggregation, random forests and boosting.

4.4.1 Introduction to Decision Trees

The Classification and Regression Tree (CART) methodology was introduced by L. Breiman, J. Friedman, R. Olshen and C. Stone in 1984. In the simplest form, a decision tree is a collection of if-else statements that are applied to data to make a prediction.

Figure 4.6 shows a graphical representation of a simple decision tree applied to a partitioned space consisting of two dimensions (James et al., 2013). The partitioned feature space is shown on the left and the resulting decision tree on the right. In this case, a regression tree is built by splitting the feature space (x_1 and x_2) into four distinct and non-overlapping regions ($R_1, R_2, ..., R_4$). New observations are allocated to regions following the tree's logic. The prediction for a new observation is the average response of all training observations allocated to the same region.

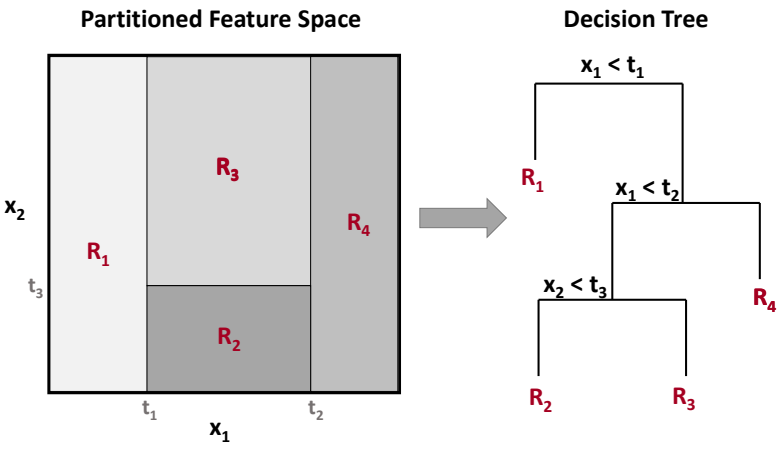


FIGURE 4.6: Illustration of a decision tree (right) corresponding to a two-dimensional partitioned space (left).

The main challenge in building decision tree models is to find the regions such that the RSS, depicted in Equation 4.3, is minimised (James et al., 2013). One way to do this is to test all possible partitions, but this quickly becomes infeasible for more complex problems than the illustrated example. Recursive binary splitting is a technique that was developed to assist with this type of optimisation. It is a top-down approach that involves starting at the top of the tree (using all observations) and successively splitting the tree into two new branches.

Recursive binary splitting typically results in complex trees that overfit on the training data. Another approach is to build a large tree and prune the tree back to a subtree of smaller size. With this approach, the challenge is to decide on the best possible subset of the tree without going through the unwieldy process of evaluating all the possible subsets. To assist with this decision, cost-complexity pruning can be used whereby a sequence of trees are considered that are indexed by a positive tuning parameter α. This is done by penalising the objective function with a term $\alpha|T|$ where $|T|$ refers to the number of terminal nodes in the subtree. In effect, the tuning parameter α is used to

determine the degree to which large trees are penalised.

The algorithm for cost-complexity pruning is summarised as follows (James et al., 2013):

1. Build a large tree using recursive binary splitting on the training data.
2. Apply cost-complexity pruning to obtain the best subset tree for each value of α
3. Use cross-validation to choose the tuning parameter, α, by calculating the CV MSE for each value of α.
4. Select the subset tree from Step 2, that corresponds to the chosen value of α in Step 3.

Decision trees are very useful, but suffer from high sampling variability and are prone to overfitting. A number of ensemble methods have been developed to produce better generalised performance. These methods are discussed in the next subsections.

4.4.2 Bootstrap Aggregation

Bootstrap aggregation or bagging, is an ensemble method that is commonly used in decision tree models. Bootstrapping, in general terms, refers to random resampling with replacement to create many simulated samples from one dataset. In the context of decision trees, bagging is used to decrease the variance that is typically associated with decision trees. It involves building multiple different decision tree models using bootstrapped subsets of a single training dataset. The final prediction is obtained by averaging the predictions across all the models (Boehmke and Greenwell, 2019). Figure 4.7 shows a graphical representation of the bagging process.

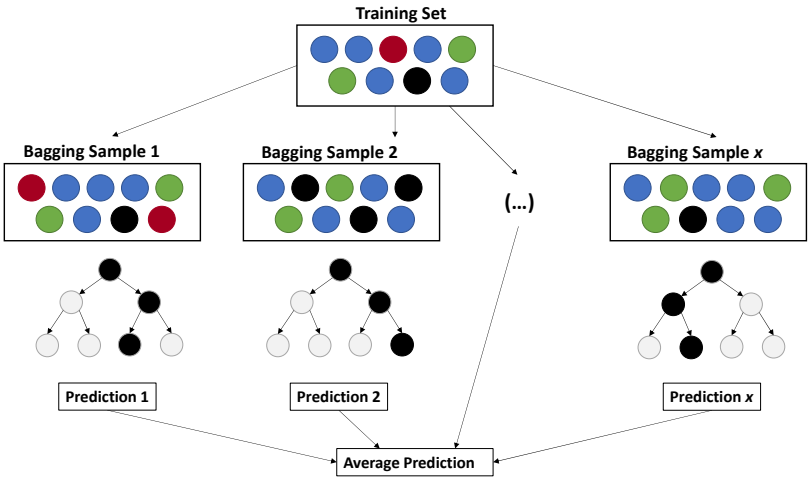


FIGURE 4.7: Graphical representation of the bagging process as it relates to a regression tree problem. The figure shows that x number of random samples (with replacement) are drawn from the dataset. Then, a decision tree is fitted to each of the samples, and the average prediction across all trees are used as the final prediction.

A very useful way to measure the performance of a bagging model is to consider the OOB error. This error stems from the fact that only a subset of the data is used when

building each model. The observations in the training set that are not used to train the individual model can therefore be used to test the predictions made by the model. For regression problems, the final OOB error can be obtained by taking the average error (eg. MSE or RMSE) across all models.

The number of trees that are built need to be prespecified when training a bagging model. A large number of trees does not result in overfitting, but could require unnecessary computation. It is common practice to determine the number of trees by plotting the OOB error against the number of trees. The idea is then to choose the number of trees at a point after which the error has converged. Also, note that bagged trees are grown deep and are not pruned, because the risk of overfitting is mitigated by growing multiple trees from the different subsets of the data. The implication is that each individual tree has high variance and low bias, but when the trees are combined, the variance is reduced.

4.4.3 Random Forests

The random forests technique (Breiman, 2001; Ho, 1995) is another ensemble method that builds upon the concept of bootstrap aggregation. With this technique, in addition to using bootstrapped samples to build the trees, only a random subset of predictor variables are considered during each split in the tree. This decorrelates the trees and causes the models to be less reliant on strong predictor variables.

The following main hyperparameters need to be considered when building a random forest model: the number of trees, the number of predictor variables that are considered during each split, the complexity of each tree, the sampling scheme, and the splitting rule. According to Boehmke and Greenwell (2019), the number of trees and the number of predictor variables considered have the biggest potential impact on the model. The number of trees need to be chosen large enough so that the error rate stabilises, and a common default for the number of predictors in a regression model is $\lfloor \frac{p}{3} \rfloor$, where p is total number of features in the dataset (James et al., 2013).

Furthermore, the complexity of the trees that are built can be controlled by tuning hyperparameters such as the minimum node size and maximum depth of each tree. The minimum node size is commonly used for this purpose and most implementations use a default value of 5 for regression analysis (Boehmke and Greenwell, 2019). The default sampling scheme is bootstrapping whereby a sample of the same size is drawn with replacement. However, the sample size can be reduced to decrease between-tree correlation. The general splitting rule for regression is the variance in the responses (Wright and Ziegler, 2017). Default values for the hyperparameters are a good starting point, but it is essential to find the optimal values by running a grid search and trying different combinations of the hyperparameters to find the best model.

4.4.4 Gradient Boosting

The gradient boosting algorithm, commonly referred to as boosting, is similar to bagging. The difference is that in bagging, trees are grown using random subsamples of the data, whereas in boosting the trees are grown sequentially using a modified version of the original dataset (James et al., 2013). Essentially, this means that the model learns by using the errors from the previous tree to build the next tree.

There are three main hyperparameters to specify when building a boosting model. The first is the number of trees. Unlike in bagging and random forests, it is possible to overfit with a boosting model, and therefore the number of trees need to be determined by looking at the CV error per number of trees. The second is the learning rate. Typical values for the learning rate are 0.01 and 0.001. A too small number can cause the model to learn slowly and require a large amount of trees, whilst too large a learning rate can result in a sub-optimal model (James et al., 2013). The learning rate depends on the problem and therefore different learning rates need to be tested to make an informed decision. The third is the interaction depth, which refers to the number of splits in each tree. This parameter controls the complexity of the trees that are built.

One benefit of applying ensemble decision tree methods is that they can provide estimates for variable importance. In general, the importance of a variable indicates how useful the variable is to make key decisions and is linked to a measurable score. In the context of regression trees, it is common to use an accuracy-based importance metric such as the reduction in variance or mean squared error associated with the removal of the feature.

4.5 Conclusion

In this chapter, the methodology was discussed. First, the modelling approach that is used during the application of the machine learning methods was discussed. It includes the split of the data into a training and test set, the application of methods, and the evaluation criteria. This was followed with an overview of MLR, ANN, and tree-based models. This chapter acts as a foundation and reference point for the application of these methods in the next chapter.

Chapter 5

Application

This chapter uses the knowledge in the previous chapter to model apple orchard yield via machine learning. First, the models are built using the chronological data split (see Section 4.1). The following machine learning techniques are applied in Sections 5.1, 5.2, 5.3 and 5.4 respectively: Multiple Linear Regression (MLR), Artificial Neural Network (ANN), random forests and boosting. Thereafter, in Section 5.5, the process is repeated using the random data split. Finally, in Section 5.6, the models are compared to determine the best-suited method and model for apple yield prediction.

5.1 Multiple Regression

In this section, the general approach described in Section 4.1 and the understanding of MLR discussed in Section 4.2 are used to build regression models. Figure 5.1 shows the steps that are performed to obtain the optimised MLR model for apple yield prediction:

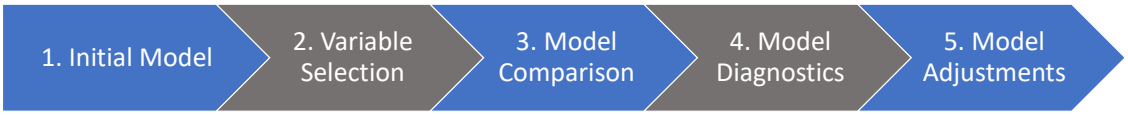


FIGURE 5.1: The five-step process to obtain the final MLR model.

Note that these steps are applied after the chronological data split into a training and test set. Recall that the 2016 to 2019 data constitute the training set and the 2020 data the test set. The presence of a large number of orchards with zero yield in the dataset, also referred to as clumping zeros or skewed data, is expected to be problematic when applying a MLR model. For the sake of consistency, the model is first applied on the data as it is, and thereafter in Section 5.1.5, alternative approaches and model adjustments are investigated to find a better-suited model. Each of the steps are discussed in further detail in separate sections below.

5.1.1 Initial Model

The first MLR model is referred to as a full model because it considers all 21 predictor variables listed in Table 3.1 to predict the yield. The outcome of the model is summarised in Table 5.1.

The small p-value shows that there is compelling evidence of a relationship between at least one of the predictor variables and the response variable. The R^2 statistics indicate a moderate fit of the model to the data with the model explaining approximately 70% of

TABLE 5.1: Summary of initial MLR model

Outcome	**Value**
p-value	< 2.2e-16
R^2	0.70
Adjusted R^2	0.69
CV RMSE	17.50

the variation in the yield. 10-Fold Cross-Validation (CV) is used to measure the performance of the model. The CV RMSE is 17.50; this is the metric that is used to compare this model to future models.

The output of the model also gives an indication on the partial significance of the variables as summarised in Table 5.2. Note that the categorical variables are converted to dummy variables when a MLR model is fitted, meaning that each category in the categorical variable is treated as a separate variable. For conciseness, the minimum p-value for categorical variables are included in the table. The table also shows a grouping of the variables according to p-values. For example: A grouping of <0.001 refers to a minimum p-value of less than 0.001, a grouping of <0.01 indicates a minimum p-value of less than 0.01 but more than 0.001. The most significant variables are shown in bold.

TABLE 5.2: Summary of the significance of variables based on the initial MLR model.

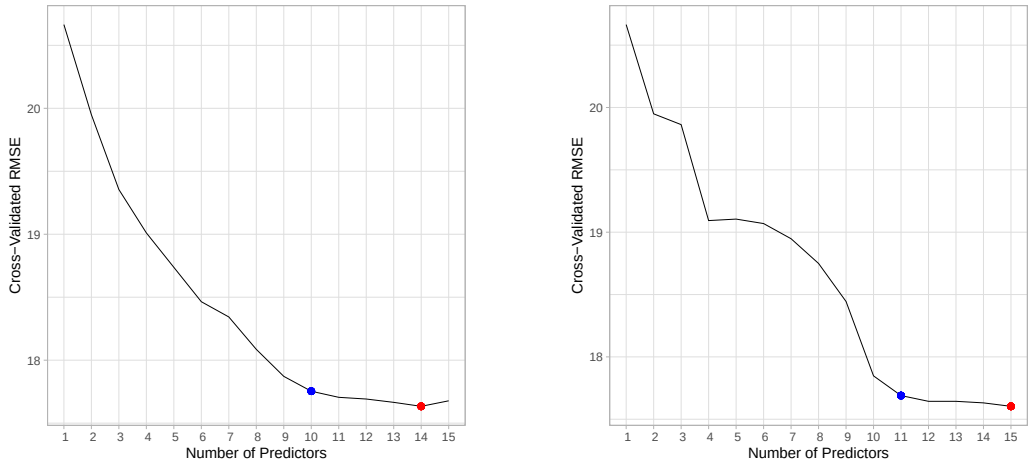
Variable	**Actual p-value**	**Minimum p-value**	**P-value group**
FARM		2.72e-05	<0.001
CULTIVAR_GRP		0.0011	<0.01
AGE	0.1438		
HIST_2_YEAR	<2e-16		<0.001
HIST_1_YEAR	0.1388		
COLD_UNITS	7.70e-08		<0.001
PRECIP	0.0086		<0.01
GDD	0.0001		<0.001
AVG_T_SEP	0.0002		<0.001
AVG_T_OCT	0.2176		
AVG_MAX_SEP	0.0007		<0.001
AVG_MAX_OCT	0.0001		<0.001
AVG_MIN_SEP	6.27e-05		<0.001
AVG_MIN_OCT	0.0003		<0.001
SUNH_SEP	5.31e-08		<0.001
SUNH_OCT	0.0231		<0.05
CCROP		0.0929	<0.10
PLANT_DIR		0.0329	<0.05
ROOTSTOCK		0.34	
SPACING1	0.1915		
SPACING2	0.304		

At this point it is tempting to drop variables with high p-values, but this is not advised because it refers to the partial significance when all the variables are included. A better strategy for choosing appropriate variables is to use variable selection techniques.

5.1.2 Variable Selection

In the previous section, a model was built using all possible predictor variables. However, it could be that some of the variables are not in fact associated with the response variable and if they are included in the model, it results in unnecessary complexity. This raises the questions of how many variables to include, and which variables to include in the model. The following variable selection techniques are applied to ease the selection process: forward selection, backward selection and lasso regularisation.

First, forward and backward stepwise selection are applied. These methods are used until the models contain 15 variables, because the error seems to flatten thereafter. For each number of variables, the model's performance is measured using 10-fold cross-validation. Figure 5.2 shows the CV RMSE per number of predictor variables for both methods.



(A) Forward selection results (B) Backward selection results

FIGURE 5.2: Forward and backward variable selection results showing the cross-validated RMSE per number of predictor variables.

When choosing a model, both the simplicity and the accuracy need to be considered, because complex models are more prone to overfitting and less interpretable. As such, for both methods two models are considered. The first is the model (indicated in red, later referred to as Forward Select 1 and Backward Select 1 respectively) with the lowest CV RMSE and the second (indicated in blue, later referred to as Forward Select 2 and Backward Select 2 respectively) is a model with fewer predictor variables at a point where the error starts to flatten. The reasoning behind the second choice is that in some cases a simpler model is preferred even though it means that the error is slightly higher. These four models are summarised in Table 5.3.

The third variable selection technique that is applied is lasso. With this technique, the choice of the tuning parameter, λ, is of utmost importance (Refer to Equation 4.6). Figure 5.3 shows the cross-validated MSE for different $\log(\lambda)$ values, and the number of variables related to each of the $\log(\lambda)$ values are also shown at the top of the plot. In this case, the log lambda value is chosen where the MSE is one standard deviation from the minimum MSE. This point is indicated by the blue line in Figure 5.3 – the $\log(\lambda)$ value is -2.62, the number of predictor variables with a non-zero coefficient is 51, and the CV RMSE is 17.77. Note that in this case each of the categories in a categorical variable are counted as separate variables due to the creation of dummy variables. If categorical

variables are considered as a whole, the number of variables used is 19. Interestingly, in this scenario, the two October temperature indicators were excluded.

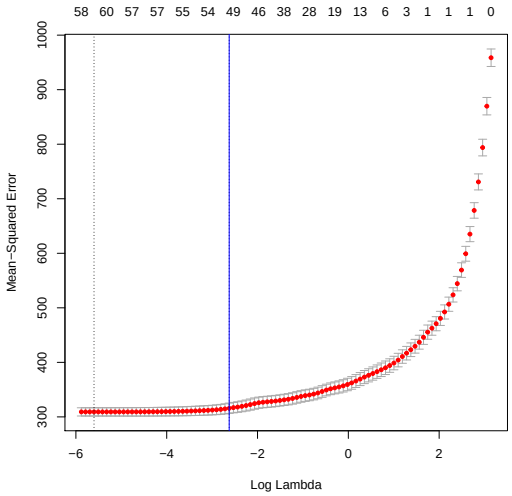


FIGURE 5.3: Lasso analysis showing the log lambda values and corresponding cross-validated MSE. The number of variables are also shown at the top of the plot.

5.1.3 Model Evaluation

Table 5.3 shows a summary of the five different models that resulted from the application of the variable selection techniques. The table shows the scenario description at the top, followed by the number of predictor variables included in the scenario and the corresponding CV RMSE. The predictor variables are also listed with a cross indicating whether the variable is included in the given scenario.

The bias-variance trade-off needs to be kept in mind when evaluating models. This means that there are two important considerations when choosing the final MLR model. The first is the resulting error measured using the CV RMSE, and the second is the complexity of the model that is indicated by the number of predictor variables. The risk of overfitting is mitigated to an extent with the use of cross-validation, but the complexity still needs to be taken into consideration because complex models are more difficult to interpret and more variables require a larger data collection effort.

Recall that the initial model with 21 predictor variables has a CV RMSE of 17.50. Considering this, Backward Select 2 is chosen as the final MLR model. This model contains 15 variables and has a CV RMSE of 17.57. It is chosen as the preferred final model because this model results in a CV RMSE that is only slightly higher than the full model, but contains 6 fewer variables.

When this model (Backward Select 2) is tested against the unseen 2020 test set, the RMSE is 22.02. The test error is higher than the cross-validated RMSE of 17.57. It is common that the test error is higher than the CV error, but the fact that the test error is 25% more than the CV error could indicate that the model overfits on the training data.

TABLE 5.3: Summary of variable selection models showing the number of parameters, the CV RMSE and the variables chosen for each scenario.

Variable Name	**Forward Select 1**	**Forward Select 2**	**Backward Select 1**	**Backward Select 2**	**Lasso**
Variable Count	11	14	11	15	15
CV RMSE	17.71	17.63	17.69	17.57	17.77
FARM	X	X	X	X	X
CULTIVAR_GRP	X	X	X	X	X
AGE	X	X		X	X
HIST_2_YEAR	X	X	X	X	X
HIST_1_YEAR					X
COLD_UNITS	X	X	X	X	X
PRECIP		X			X
GDD			X	X	X
AVG_T_SEP	X	X	X	X	X
AVG_T_OCT	X	X		X	
AVG_MAX_SEP	X	X	X	X	X
AVG_MAX_OCT	X	X	X	X	
AVG_MIN_SEP			X	X	X
AVG_MIN_OCT			X	X	X
SUNH_SEP	X	X	X	X	X
SUNH_OCT	X	X			X
CCROP					X
PLANT_DIR		X		X	X
ROOTSTOCK		X		X	X
SPACING1					X
SPACING2					X

To determine the final coefficients for the model, it is necessary to run the model on the full dataset (test and training set). A summary of the model is shown in Table 5.4 and the regression coefficients are shown in Appendix A.

TABLE 5.4: Summary of final MLR model

Outcome	**Value**
p-value	$<$ 2.2e-16
R^2	0.68
Adjusted R^2	0.67

5.1.4 Model Diagnostics

Linear regression models are based on the following assumptions: the response can be depicted as a linear combination of the predictor variables, the errors are independent and normally distributed and the error variance is the same for any set of predictor variables. When evaluating a model, it is important to check these assumptions; model diagnostic plots are commonly used for this purpose.

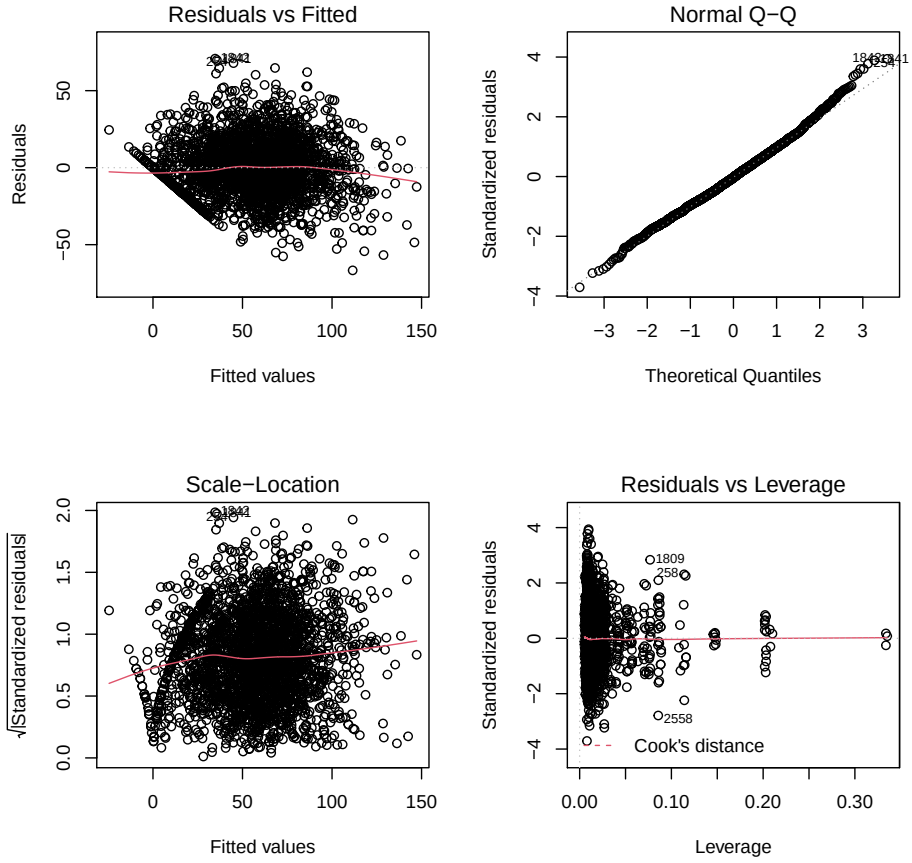


FIGURE 5.4: Model diagnostic plots for final MLR model

Figure 5.4 shows the model diagnostic plots for the final model. Firstly, the residuals versus fitted values plot can be used to check the linearity and homoscedasticity assumptions. The plot shows that the residuals are centered around 0 without an increase or decrease along the fitted values axis. However, the diagonal line on the left-hand side of the plot is a concern. After further investigation, it is noticed that this is related to the large number of non-bearing orchards in the dataset (illustrated in Figure 3.1). More specifically, these are cases where the model predicted a non-zero yield for orchards that had an actual yield of 0.

Secondly, the normal Q-Q plot can be used to check whether the errors are normally distributed. Ideally, for normally distributed residuals, the residuals should follow a straight line. In this case, the residuals follow a relatively straight line with a slight upward curve towards the top which shows that the errors are fairly normally distributed.

Thirdly, the residuals versus leverage plot can be used to identify any points with large influence. Points with high influence are outlier points that also have high leverage, where leverage refers to the impact that a point has on a model. Cook's distance is a function of standardised residuals and leverage, and is commonly used to measure influence. In this case, none of the points in the data are beyond Cook's distance.

In summary, even though the model seems to work reasonably well for the data, the diagonal patterns in the plots are a cause of concern and the current MLR model is

not the best-suited model for the problem. One option, that is investigated in the next section, is to look at alternative ways to treat the large number of zeros in the dataset. Another option is to consider non-linear methods such as neural networks and random forests.

5.1.5 Skewed Data Approach

As mentioned in Section 4.2.4, there are various approaches to deal with skewed data with a large number of zeros in the context of MLR models. In this case, a simple two-part method is followed. First, the dataset is split into two separate datasets - one consists of non-bearing orchards and another of bearing orchards. Thereafter, a zero yield is assumed for the non-bearing orchards and a MLR model is used to predict the yield for bearing orchards only.

Seeing as the bearing characteristics of trees are predominantly determined by the age of the tree, a simplified method according to age is used to categorise orchards – trees under the age of three years are considered non-bearing and the rest bearing. The cut-off is decided based on the fact that 87% of all non-bearing orchards in the dataset are under 3 years old and that more than 90% of orchards that are under three years old have a zero yield. Refer to Table 5.5 for a representation of zeros in the dataset. The first column shows the percentage of zeros in the age group and the second shows the percentage of zeros in the age group when compared to the total number of zeros in the dataset.

TABLE 5.5: The percentage of zero occurrences per age group and the percentage of zeros in the age group compared to the total number of zeros in the dataset.

Age	% Zeros in Age Group	% of Total Zeros
0.00	100.00	4.10
1.00	97.80	36.10
2.00	93.50	47.10
3.00	11.30	6.10
4.00	8.50	4.10
5.00	4.90	2.00
6.00	1.00	0.40

This assumption is supported by the results of a simple classification tree that uses the data to predict the bearing characteristic based on age. If a tree with two terminal nodes is built, the age cut-off is observed as 2.5 years whereby trees under 2.5 years are classified as non-bearing with a 1.5% misclassification error rate.

Two different approaches are used to build the MLR model for the bearing orchards. The first approach uses all 21 predictor variables and the second approach applies backward stepwise selection and then chooses the best subset model. In this case, the best subset model contains 14 predictor variables. The CV RMSE is 17.09 for the full model and 17.19 for the subset model. Seeing as the subset model is simpler (7 less predictor variables), and only has a slightly higher RMSE, the subset model is considered as the final MLR model for the bearing orchards.

Table 5.6 shows a summary of the outcome of the model. The test RMSE, for the two-part model considering both the bearing and non-bearing orchards, is also shown. Interestingly, the model fit is worse than the original MLR model, but the CV RMSE is better. This model is referred to as the two-part MLR model. Both this model and the original MLR model are considered and compared to the other models in Section 5.6.

TABLE 5.6: Summary of Two-Part MLR model

Outcome	**Value**
Number of predictors	14
p-value	$<$ 2.2e-16
R^2	0.60
Adjusted R^2	0.60
CV RMSE	17.19
Test RMSE	23.73

5.2 Neural Networks

In this section, an Artificial Neural Network (ANN) is designed to predict apple yield. Various combinations of hyperparameters are tested by means of a grid search to find the best performing model.

When building a neural network, one-hot encoding is used to convert categorical variables into binary indicators. This implies that the 21 predictor variables are converted to 61 variables. The large number of variables in comparison to the size of the dataset could pose a problem for the neural network, because it results in an even larger number of parameters that need to be optimised. It pre-empts the fact that the model could be prone to overfitting and highlights the need to apply regularisation techniques.

The general structure of the ANN is shown in Figure 5.5. The network consists of 61 neurons in the input layer representing the predictor variables, and 1 neuron in the output layer due to the fact that this is a regression problem. In addition, MSE is used as the loss function and the model is trained with the stochastic gradient descent algorithm using backpropagation and adaptive learning.

The general architecture can be altered by changing the number of hidden layers and the number of neurons in each hidden layer. Furthermore, the activation function in the hidden layer can be changed and regularisation techniques such as dropout, weight decay and early stopping can be added. The choice of these hyperparameters are made by means of a random grid search. Table 5.7 shows the parameter space used.

The number of units in the hidden layer are chosen with the goal to design a model that is complex enough to capture the relationships in the data, yet simple enough to generalise. Considering the limited size of the dataset in comparison to the number of predictor variables, care is taken to not make the model overly complex. As such, the considerations for the units in the hidden layer range between 4 and 32. A more complicated network structure consisting of two hidden layers was also tested, but performed worse than the single layer and was therefore excluded from the final parameter space for the sake of conciseness.

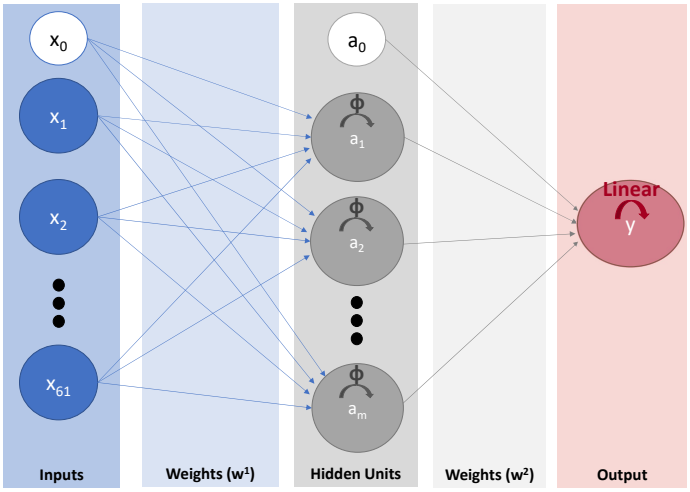


FIGURE 5.5: General architecture of the neural network model. The input layer contains 61 neurons and the output layer consists of 1 neuron using the linear activation function.

TABLE 5.7: Values for neural network hyperparameter grid search.

Hyperparameter	**Values**
Hidden layer Units	4, 8, 16, 32
Hidden layer activation function	Tanh, ReLU
L2 Regularisation	0, 0.01, 0.1
Dropout (hidden layer)	0, 0.1, 0.2
Stopping rounds	5
Stopping criteria	0.0001
Epochs	200

The tanh and ReLU activation functions are considered for the hidden layer because these functions have been seen to perform well as mentioned in Section 4.3. For the dropout, a range of values between 0 and 0.2 are considered, and for the regularisation hyperparameter, a range of values between 0 and 0.1 are included. The number of epochs is set high at 200 with conservative stopping criteria specified to enable early stopping.

Table 5.8 shows the 10 models resulting in the lowest CV RMSE. The model parameters and associated CV RMSE are shown. Figure 5.6 shows the CV RMSE and training RMSE for these 10 models. The model that performs the best is the model with 16 neurons, using the tanh activation function in the hidden layer, and no dropout.

Using the top model to predict on the test set yields an RMSE of 28.00. The big difference between the CV and test RMSE shows that the model is overfitting, even though overfitting is mitigated to an extent with the use of cross-validation. However, this could also point to a fundamental difference in the training and test set due to unique properties of the 2020 season. This is further discussed in Section 6.1.

TABLE 5.8: A summary of the top 10 neural network models showing the model hyperparameters and CV RMSE

Model	Activation function	Hidden neurons	Ridge lambda	Hidden dropout	CV RMSE
1	Tanh	16	0	0	14.7616
2	Tanh	32	0	0	15.0465
3	ReLU	32	0.01	0	15.1058
4	ReLU	32	0	0	15.144
5	ReLU	32	0	0.1	15.1898
6	ReLU	32	0.01	0.1	15.1961
7	Tanh	8	0	0	15.2247
8	ReLU	8	0.01	0	15.2282
9	ReLU	32	0	0.2	15.3234
10	Tanh	16	0.01	0	15.344

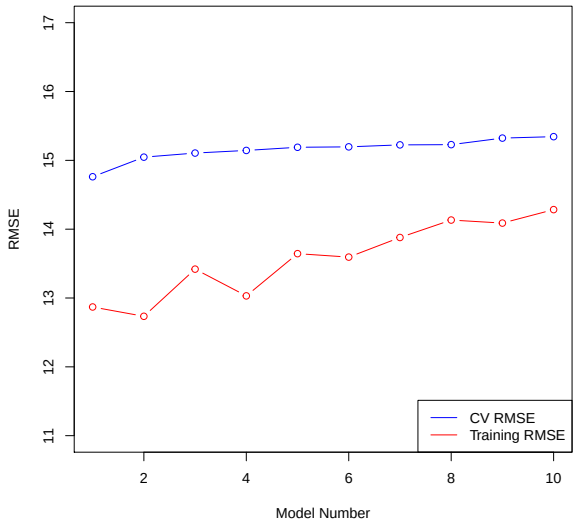


FIGURE 5.6: ANN performance: CV and train RMSE for top 10 models

5.3 Random Forests

In this section, an initial random forest model is fitted as a base model, and thereafter the hyperparameters are optimised by means of a grid search. Again, the models are trained using data from 2016 to 2019 and tested against the 2020 data.

There are three main hyperparameters to specify when building a random forest model:

- *ntrees*: The number of trees to build.
- *m*: Number of predictor variables to include in the set of features considered during each split.
- *min node size*: Minimum node size that controls the depth of the trees.

For the initial model, 200 trees are built and the default values for the other two variables are used (Liaw and Wiener, 2002; Boehmke and Greenwell, 2019). A summary of the variables and performance of this model are shown in Table 5.9.

TABLE 5.9: Summary of initial random forests model

Hyperparameters	
ntrees	200
m	5
min node size	5
Metrics	
OOB RMSE	12.81
% Variance explained	82.94%

The OOB RMSE error seems to start converging between 50 and 100 trees (Refer to Figure 5.7). However, 200 trees will be used because computation is not constrained.

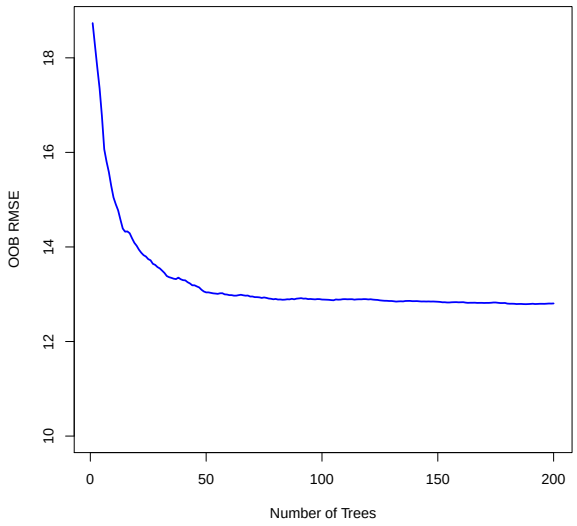


FIGURE 5.7: Random forests OOB RMSE versus number of trees

To find appropriate values for the other two hyperparameters, a grid search is performed. Due to the small size of the model and dataset, computation time is not a problem and it is possible to test a wide range of values. The default values are considered as approximate centre points for the choice of values. As such, values ranging from 2 to 15 are tested for the value of m; and to control the depth of the trees minimum node sizes of 1, 3, 5 and 7 are tested.

Figure 5.8 shows the OOB RMSE for each combination of hyperparameters. It is evident from the plot that the error decreases until at least 9 features are considered during each split. Also, in general, a lower minimum node size corresponds with a lower error. Table 5.10 shows a summary of the 10 models with the lowest OOB error. The best performing model has an OOB RMSE of 12.6 and the following hyperparameters: $m = 14$, *min node size* $= 3$. The proportion of variation explained by this model is 83.6%, and finally, when it is tested against the unseen 2020 test set, the RMSE is 21.52.

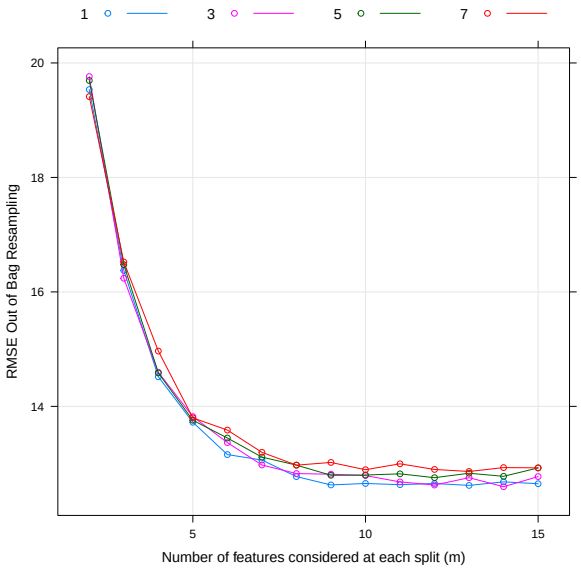


FIGURE 5.8: Random forests hyperparameter tuning results showing the OOB RMSE for each combination of hyperparameters.

TABLE 5.10: A summary of the top 10 random forest models showing the model parameters, OOB RMSE and R-squared statistic

m	min node size	OOB RMSE	R^2
14	3	12.60	0.836
13	1	12.62	0.836
12	3	12.63	0.836
9	1	12.63	0.837
11	1	12.63	0.836
15	1	12.65	0.835
12	1	12.65	0.835
10	1	12.65	0.836
11	3	12.68	0.835
14	1	12.68	0.834

5.4 Gradient Boosting Machines

In this section, the gradient boosting algorithm, commonly referred to as boosting, is applied. The following hyperparameters are tuned by means of a grid search: Number of trees, learning rate and interaction depth. Table 5.11 shows the values that are considered for each of the hyperparameters, whereby the default values are considered as an approximate mean.

Table 5.12 shows the 10 best boosting models and their hyperparameters. The model with the lowest CV RMSE of 13.33 is obtained with the following hyperparameters: 300 trees, interaction depth of 7 and learning rate of 0.05.

To test if the number of trees is optimum at 300, the CV RMSE is calculated per number of trees, while the interaction depth is kept at 7 and learning rate 0.05. Figure 5.9 shows that the point at which the CV RMSE is the lowest is in fact at 293 trees and, therefore, this is considered the final boosting model. The test RMSE for the model is 22.45.

TABLE 5.11: Values for the boosting model hyperparameter grid search.

Hyperparameter	**Values**
Number of Trees	100, 300, 500
Interaction Depth	1, 5, 7, 10, 13
Learning rate	0.01, 0.05, 0.1

TABLE 5.12: A summary of the top 10 boosting models showing the model hyperparameters and CV RMSE

Learning Rate	Interaction Depth	Number of Trees	CV RMSE
0.05	7	300	13.33
0.10	7	100	13.38
0.05	7	500	13.40
0.01	7	500	13.43
0.05	5	500	13.45
0.05	5	300	13.46
0.05	7	100	13.48
0.05	10	300	13.49
0.10	7	300	13.49
0.05	10	500	13.50

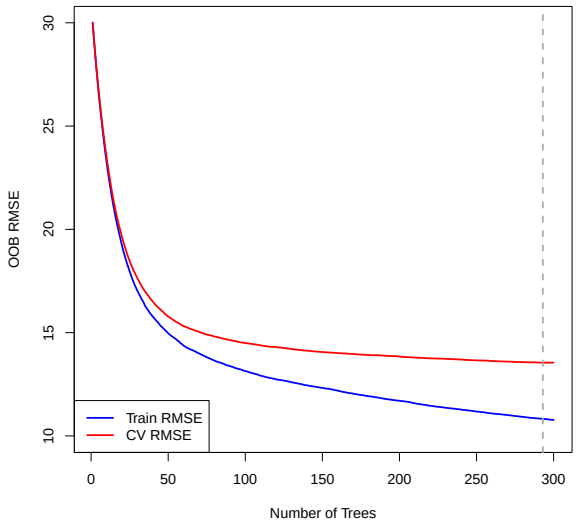


FIGURE 5.9: Boosting model train and CV RMSE versus number of trees

5.5 Random Data Split

To further evaluate the models and see how well the models perform across various seasons, the four methods are also used to build models based on a randomly selected training and test set. In this case, an 80/20 split is used whereby 80% of the data constitutes the training dataset and the rest the test set.

The models are retrained and optimised using the random training set and evaluated against the random test set. For conciseness, the entire process is not described in this document, but the final model parameters and results are presented below.

Table 5.13 shows a summary of the MLR and Two-Part MLR models. For both these types of models, full and subset models were built and the results compared to find the final model. The number of variables are also shown together with the results.

TABLE 5.13: Random data split: Summary of MLR Models

Outcome	**Value**
Number of predictors	13
p-value	$<$ 2.2e-16
R^2	0.68
Adjusted R^2	0.67
CV RMSE	18.11
Test RMSE	18.32

(A) MLR model

Outcome	**Value**
Number of predictors	15
p-value	$<$ 2.2e-16
R^2	0.61
Adjusted R^2	0.61
CV RMSE	17.71
Test RMSE	17.40

(B) Two-Part MLR model

The neural network model is retrained using the same general architecture and applying a grid search on the same parameter search space shown in Table 5.7. The final model parameters and associated CV and test RMSE values are shown in Table 5.14.

TABLE 5.14: Random data split: Summary of the neural network model showing the model parameters, CV RMSE and test RMSE

Model	Activation function	Hidden neurons	Ridge lambda	Hidden dropout	CV RMSE	Test RMSE
1	Tanh	32	0	0	15.66	15.50

Finally, the random forests and boosting models are also retrained and a grid search applied on the same hyperparameter search space. Tables 5.15a and 5.15b show the final model hyperparameters and associated performance for each of the methods.

TABLE 5.15: Random data split: Summary of tree-based models showing model hyperparameters and associated performance

Hyperparameters	
ntrees	200
m	10
min node size	1
Metrics	
OOB RMSE	13.36
Test RMSE	14.14

(A) Random forests model

Hyperparameters	
ntrees	300
interaction depth	7
learning rate	0.05
Metrics	
CV RMSE	14.09
Test RMSE	14.52

(B) Boosting model

For all models, the test performance using the random split in the data is significantly better compared to the performance against 2020's data. In the next section, the performance of the models are compared and discussed - the results of both data splits (chronological and random) are considered.

5.6 Machine Learning Model Comparison

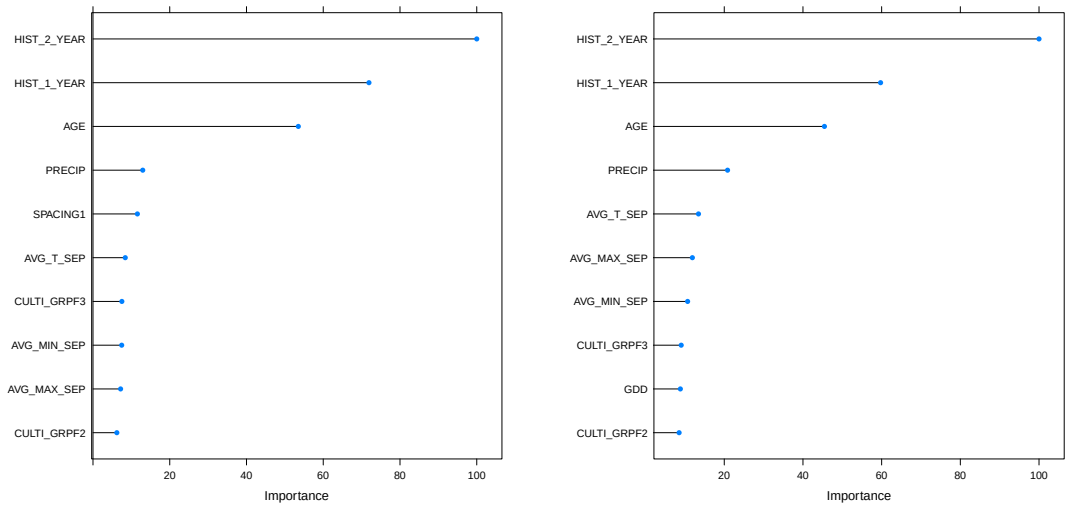
Table 5.16 shows a summary of the results for the five final machine learning models. The validation and test RMSE values for both the chronological and random split are shown. Note that the 'Validation RMSE' refers to either the CV or OOB error.

For the chronological data split, the neural networks model is the worst performing model with a test RMSE of 28.00, and the random forest model is the best with a test RMSE of 21.52. For the random data split, the MLR model performed the worst with a test RMSE of 18.32 and the random forest model performed the best with a test RMSE of 14.14. Overall, the random forest model is the best performing model with the lowest test RMSE for both data splits. This model is therefore considered as the final and best-suited model for this case of apple yield prediction.

TABLE 5.16: Summary of the machine learning models that shows the validation (CV or OOB) RMSE and test RMSE for both the chronological and random split of data.

	Chronological Split RMSE		**Random Split RMSE**	
Model	**Validation**	**Test**	**Validation**	**Test**
MLR	17.57	22.02	18.11	18.32
Two-Part MLR	17.19	23.73	17.71	17.40
Neural Network	14.76	28.00	15.66	15.50
Random Forest	12.60	21.52	13.36	14.14
Boosting	13.33	22.45	14.04	14.52

One advantage of random forest models is that they can indicate variable importance. The 10 most important variables based on the impurity and permutation-based measures are shown in Figures 5.10a and 5.10b respectively.



(A) Importance based on impurity

(B) Importance based on permutation

FIGURE 5.10: Variable importance for the final random forest model using the chronological split and two different measures for importance

The three most important variables in both cases are the 2-year historic yield, the 1-year historic yield and the age of the trees. Interestingly, the September temperature indicators are considered the most important climate variables.

The large discrepancy between the test and validation errors when the chronological data split is used is worrying. It raises questions regarding the ability of the model to be used to generate future yield predictions for a specific year and, therefore, it is further investigated and discussed in Section 6.1. All the models seem to generalise well when the random split of data is used.

5.7 Conclusion

The aim of this chapter was to apply various machine learning methods to determine the best-suited model for apple yield prediction. The following techniques were applied: MLR, ANN, random forests and boosting. First, the methods were applied using the chronological data split and thereafter, the process was repeated for the random data split. Overall, the random forest model was identified as the best performing model with the lowest average test RMSE. The ANN model was the worst performing model, followed by the MLR model. In the next chapter, the results are further discussed and the study concluded.

Chapter 6

Discussion and Conclusion

This chapter discusses the results and concludes the study. First, the poor generalisation performance on the 2020 dataset is investigated. Thereafter, the final machine learning model obtained by means of the random forests method is compared to simple estimators for yield. This is followed by a discussion on the implication of the results, limitations of the study, and recommendations for future work.

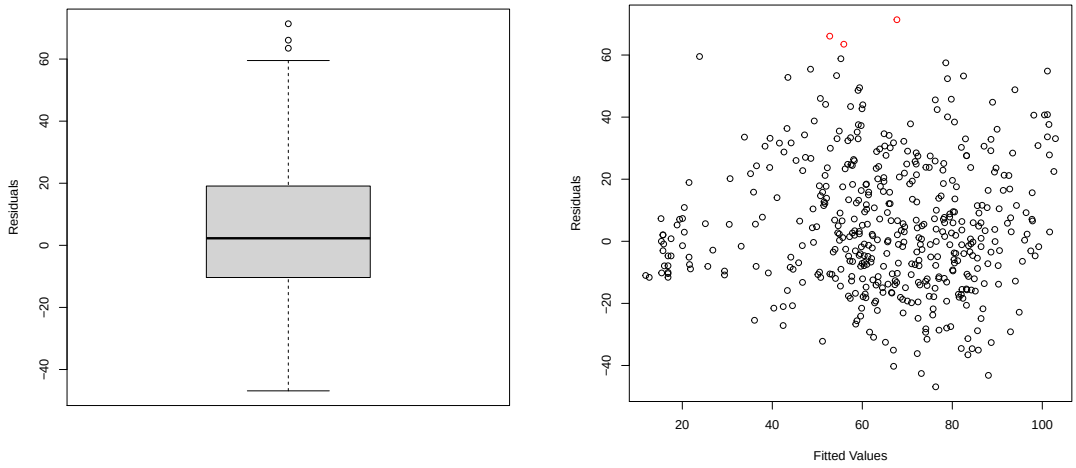
6.1 2020 Performance Investigation

The large discrepancy between the best random forest model's CV RMSE and test RMSE (12.60 versus 21.52) when the chronological data split is used is concerning, because it raises questions regarding the model's ability to predict future seasons' harvests. It points to an issue with either the model or with the data. On the one hand, models that are too complex could lead to overfitting, meaning that they fit the data too well and capture the noise in the data instead of only the signal. On the other hand, data consisting of too few observations or relevant features could be insufficient in capturing the underlying relationships in the data. Also, an important assumption in supervised learning applications is that the testing and training data are representative of the same population.

The model's performance under the random data split can be used as an indication of the model's ability to capture the relationship in the data. The random forest model performs well when using the random split, with a test RMSE of 14.14 and only slightly better CV RMSE of 13.36. The fact that hyperparameters are optimised using the OOB error also helps to mitigate the risk of overfitting. Therefore, the poor generalisation performance when the chronological data split is used seems more attributable to the data, and specifically 2020 data, than the model.

A closer look at the residual values can help further investigate and test this assumption. Figure 6.1a presents a boxplot of the residuals. The fact that the residuals have a positive median and are skewed to the right shows that the model generally predicted a lower yield than the actual 2020 yield. The data investigation in Section 3.5 shows that the 2020 season had the highest average yield over the 5 years. The fact that the range of yields in the training set differs to the test set could make it difficult for the model to make accurate predictions, perhaps explaining the large error. This is especially pertinent due to the fact that the model considers the 2-year historic average as the most important predictor variable.

Figure 6.1b shows the residual versus fitted values. Even though the estimates are spread fairly evenly along the x-axis, there is a slight funnel shape indicating that the model's predictions are worse for higher yield predictions. The plot also shows that there are a



(A) Residuals boxplot for final random forests model (B) Residuals versus fitted values for final random forests model

FIGURE 6.1: Residual plots for final random forests model

few extreme residual values that are larger than 60 (indicated in red), meaning that the model predicted a yield that is more than 60 ton/ha lower than the actual yield. Upon further investigation, we notice that these are orchards for which the difference in the 2-year historic yield and the 2020 yield is very large.

The model's reliance on the 2-year historic yield could also be problematic in the case of young orchards where there is a clear increase in the yield until the tree reaches full production. To investigate the impact on this, the average absolute residuals are plotted against the age group in Figure 6.2. It is evident from the plot that the model performs the worst at predicting yield between the ages 6 to 8. This is possibly due to the fact that those are typically ages at which the yearly yield varies the most, making it difficult to predict using a limited set of explanatory variables with a strong dependence on the 2-year historic yield.

To test whether the poor 2020 performance is unique to the 2020 season or applicable to any specific year, the model is retrained using a more regular season (2017) as the test set. When the 2017 data is used as the test set and the model retrained, the OOB RMSE is 13.40 and the test RMSE is 16.54. The fact that the difference between the errors is notably smaller means that the model is better suited to predict the 2017 season than 2020. Interestingly, very different hyperparameters are chosen for this scenario with the number of parameters considered at each split being 12 and a minimum node size of 1.

Furthermore, the poor 2020 generalisation performance could also indicate that the model is not using enough or appropriate explanatory variables and, therefore, the model is struggling to find a generalised relationship between the input and the output. In this case, the choice of explanatory variables largely depends on data availability, which is influenced by the type of data required and the timing of when data is available. Firstly, apple production is dynamic and there are additional factors that contribute to apple yield for which data are not available, such as orchard management practices and tree-specific microclimate. Secondly, apple yield is impacted by factors and unforeseen events that are not yet known at the time of prediction, such as droughts, pests and diseases, and untimely rain.

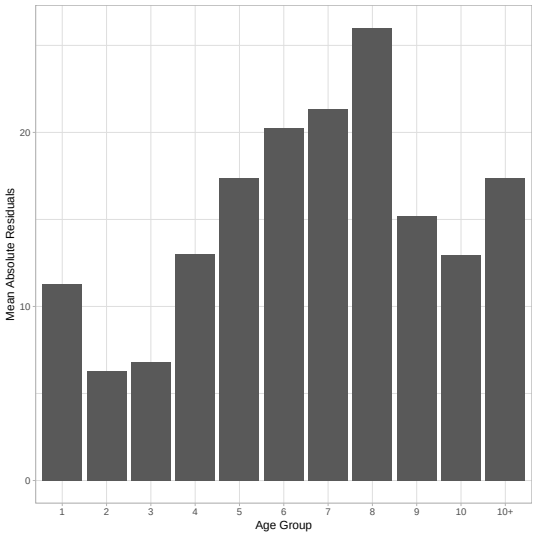


FIGURE 6.2: Average absolute residuals per age group (in years) for final random forests model

In summary, the model's poor generalisation performance on the 2020 dataset is attributable to the data used in this study, highlighting the role of seasonality in apple production. The assumption that the training and test set belong to the same population is questionable in this case. Bengio, Goodfellow, and Courville (2017)'s advice – that more training examples are required if a model has reached optimal capacity but still struggle to generalise – seems applicable to this scenario. Also, additional predictor variables can potentially improve performance. These two opportunities are expanded on in Section 6.4. The good performance when the random split is used indicates that the model is more suitable to make long-term predictions over a couple of years when the seasonality of one specific year plays a less prominent role.

6.2 Model Evaluation

In this section, similar to what was done in Paudel et al. (2021)'s work, the final model is evaluated by comparing it to other more simple and intuitive approaches. First, the standard deviation is considered and thereafter simple estimators.

The standard deviation is the benchmark that any model needs to beat, because it indicates the amount of error that naturally occurs in the response variable. In this case, the standard deviation for apple yield in the dataset is 31.5. This is comparable to the error that would have been observed if we used the average yield as the prediction for all orchards. The test RMSE for the model is far lower than the standard deviation and, therefore, the model outperforms this benchmark.

Simple yield estimations based on averages are commonly used to predict apple yield in practice (Gonzalez-Sanchez, Frausto-Solis, and Ojeda-Bustamante, 2014; Cheng et al., 2017). This can be done by either predicting the yield per orchard using the historic performance of the orchard, or by using averages per age and cultivar. The 2020 test RMSE for these methods are shown in Table 6.1 and compared to the final random forest model.

TABLE 6.1: Summary of simple average models.

Model	**2020 Test RMSE**
2-Year historic yield	27.93
Age norms	28.52
Age per cultivar norms	28.11
Final random forests model	*21.52*

The methods are described as follows:

- **2-Year historic yield:** The forecast for each orchard is the average of the two previous years' actual yield for the same orchard.
- **Age norms:** The forecast for the orchard is based on the age of the orchard. The forecast is the average yield of all the orchards in the training set that are the same age as the orchard.
- **Age per cultivar norms:** The forecast for the orchard is based on the age and cultivar of the orchard. The forecast is the average yield of all the orchards in the training set that are the same age and in the same cultivar group as the orchard.

In summary, even though the final model has a fairly high test RMSE when using the 2020 split in the data, the model outperforms other simple and intuitive approaches. This illustrates the benefit of adding additional factors when forecasting yield, and also motivates the need to apply machine learning techniques to incorporate these factors into the predictions.

6.3 Implications of Results

The aim of this study is to build a machine learning model to predict apple yield. However, besides the yield prediction, the model can also be used to better understand the relationship between the input and output variables. Naturally, the model results lead to a series of questions, such as how the model can be used, what it shows about the relationships in the data and which variables contribute the most to apple yield prediction. These questions are discussed below.

How can the model be used?

- The model can be used as a base model for yield predictions, and then updated as the time to harvest decreases and more explanatory variables become available.
- The model can be used to predict the average yield over a number of years. This could be helpful, for example, when modelling future yield to do capacity planning.
- The model serves as a starting point for future optimisations that include additional features, more data or other methods.

Which variables contribute the most to apple yield prediction?

In both the multiple regression models and tree-based models, the 2-year historic yield is the most important predictor variable. Other variables that are considered important

according to the random forest model are the one year historic yield, age, precipitation, September temperature and the cultivar group (specifically F2 and F3).

What can we learn from the results?

- Apple harvests vary largely per season.
- Early yield prediction (4 - 6 months in advance) is difficult because of the lack of explanatory variables available at that point in time.
- Machine learning models have the potential to outperform simple estimator models based on averages.
- The random forest model is the best performing model for this specific scenario of apple yield prediction.
- The neural networks model is the least-suited model for this specific scenario of apple yield prediction.
- The 2-year historic yield of the specific orchard has the most predictive power compared to the other predictor variables. Therefore, yield predictions per orchard are better than averages per cultivar.
- The type of cultivar, the age of the orchard and the temperature in September (preceding the harvest) are also important indicators for the yield.
- Due to the variability in yield between orchards, apple yield should be predicted per orchard.

6.4 Limitations and Recommendations

Limitations are a natural and unavoidable part of any scientific research project. It refers to characteristics of the methodology or design that impacted the research results and findings. The following limitations were exposed during this study:

- Methods applied – The choice of the four machine learning methods was made according to available literature and previous studies. However, in addition to these methods, there are a number of other methods that could have been considered. Examples include support vector machines and time-series analysis.
- Predictor variables – In this study, the choice of predictor variables is largely driven by the data available at the time of prediction. It limits the type of data that can be used and causes many predictor variables that potentially impact the yield to be excluded.
- Size of dataset – The model is built using a fairly small dataset consisting of 5 years of production data. More training data would help to improve the prediction accuracy of the model. This is especially important for the prediction of apple yield that varies largely per season and is impacted by many different factors.

The limitations, as well as renewed insights into the field of apple yield prediction, lead to the following recommendations or opportunities for future research:

- Repeat the study with a larger dataset containing more observations and additional predictor variables (e.g. image-related data, fertilizer application data, water-related data, soil data and the timing of various orchard management practices).
- Apply additional machine learning models for apple yield prediction and compare to the final random forest model.
- Use machine learning techniques to build an apple prediction model with a shorter prediction time-frame.
- Capture additional orchard management data that can be fed into the model.

In summary, this model serves as a baseline that can be improved by incorporating additional data sources, more predictive features and using different machine learning algorithms. It adds value by acting as a starting point or pilot project that can be used for further optimisations.

6.5 Concluding Remarks

The prediction of apple yield is important as it impacts resource requirements across the entire apple supply chain, yet it is a challenging task because the yield is impacted by many inter-related factors. Machine learning methods can be used to identify complicated relationships between input and output variables. As such, this study aimed to build a machine learning model for apple yield prediction. It focused specifically on early yield prediction, meaning that the prediction is in the October preceding the harvest.

The study started with a sound literature review extending across the fields of apple crop management and crop yield prediction in the context of the agricultural digital revolution. Thereafter, the data were explored with the use of summaries and trend analysis. This was followed by the methodology chapter that included the modelling approach and a technical overview of the methods applied.

Four different machine learning methods were considered for apple yield prediction in this study: multiple linear regression, neural networks, random forests and boosting. These methods were applied and hyperparameters were optimised with the aim of finding the best possible model per method. Thereafter, the models were compared to find the best-suited model and method for this specific scenario of apple yield prediction.

The model application process was repeated twice using two different datasets: First, using the chronological split – where the data for 2016 - 2019 were used as the training set and the 2020 data as the test set – and then using the random split, whereby 80% of the data were randomly selected as the training set and the remainder used as the test set. Overall, the random forest model performed the best with a test RMSE of 21.52 and 14.14 using the chronological and random split respectively. When the chronological data split was used, the neural network model performed the worst, and the multiple linear regression model had the worst performance when the random split was used.

The final random forest model performed better than simple estimators for yield, showing that machine learning methods have the potential to add value to apple yield predictions. The models did however struggle to generalise when the chronological split is used, which is attributable to the uniqueness of the 2020 data, but also highlights limitations to the current study. These limitations are centred around the limited data used

and therefore recommendations for future work include the use of additional predictor variables and more data observations. In addition, the application of additional machine learning methods and the use of a different prediction time-frame are also suggested.

This study is different to previous studies because it evaluated more than two machine learning methods for apple yield prediction and focused on early yield prediction. The evaluation of multiple methods is valuable because some methods are better suited for certain problems than others, and early yield prediction is important for capacity planning and budgeting.

The overall research objective of this study is achieved because the best-suited machine learning model for early apple yield prediction is identified as the random forest model. Furthermore, the sub-objectives, stated in Section 1.3, are also achieved:

1. The state of the agricultural digital revolution has been established.
2. The factors impacting apple yield have been summarised.
3. The methods for crop yield prediction have been defined.
4. The data are understood.
5. Machine learning methods have been applied and compared.

In conclusion, this study proposed the application of machine learning methods for apple yield prediction. Four different machine learning methods were applied and compared, and the random forest model was identified as the best-suited model for the specific scenario. The study contributes towards a more comprehensive view of apple yield prediction and the model serves as a baseline model for further optimisation.

Appendix A

Final Multiple Linear Regression Model

Table A.1 shows a summary of the regression coefficient estimates, standard error, t-values and p-values for the final MLR model.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	7721.16	880.35	8.77	0.00000
HIST_2_YEAR	0.81	0.02	53.55	0.00000
AVG_T_SEP	-49.05	3.63	-13.51	0.00000
SUNH_SEP	719.50	41.52	17.33	0.00000
FARMB	6.24	1.30	4.81	0.00000
FARMC	28.74	1.82	15.79	0.00000
FARMD	20.99	1.95	10.75	0.00000
FARME	2.64	2.78	0.95	0.34235
FARMF	3.68	3.51	1.05	0.29518
FARMG	-0.08	2.95	-0.03	0.97928
AVG_MIN_OCT	-205.54	22.75	-9.04	0.00000
GDD	13.03	1.44	9.06	0.00000
AVG_MAX_OCT	-207.74	23.01	-9.03	0.00000
AVG_MIN_SEP	-184.19	21.05	-8.75	0.00000
COLD_UNITS	0.06	0.01	8.14	0.00000
CULTI_GRPF10	-18.41	4.76	-3.87	0.00011
CULTI_GRPF11	6.20	4.41	1.41	0.15983
CULTI_GRPF12	-5.40	5.25	-1.03	0.30315
CULTI_GRPF13	-17.48	7.17	-2.44	0.01490
CULTI_GRPF14	-19.26	10.68	-1.80	0.07144
CULTI_GRPF15	-2.04	5.76	-0.35	0.72346
CULTI_GRPF2	-4.57	1.84	-2.49	0.01293
CULTI_GRPF3	2.82	1.84	1.53	0.12589
CULTI_GRPF4	-4.28	2.02	-2.12	0.03377
CULTI_GRPF5	0.01	2.14	0.00	0.99661
CULTI_GRPF6	0.75	2.15	0.35	0.72652
CULTI_GRPF7	2.90	2.99	0.97	0.33306
CULTI_GRPF8	-2.18	3.15	-0.69	0.49015
CULTI_GRPF9	3.55	3.14	1.13	0.25928
AVG_MAX_SEP	-168.27	20.52	-8.20	0.00000
AGE	-0.12	0.04	-3.13	0.00178
PLANT_DIRNOORD_SUID	0.18	1.96	0.09	0.92673
PLANT_DIRNOORD_WES	1.02	2.04	0.50	0.61643
PLANT_DIROOS_WES	3.49	2.28	1.53	0.12604
PLANT_DIRSUID	-5.11	6.51	-0.78	0.43318
PLANT_DIRSUID_WES	0.05	8.39	0.01	0.99566
PLANT_DIRWES	5.91	5.85	1.01	0.31219
AVG_T_OCT	6.41	2.28	2.81	0.00503
ROOTSTOCKGEMENG	6.41	8.33	0.77	0.44147
ROOTSTOCKM111	-1.39	9.50	-0.15	0.88379
ROOTSTOCKM25	3.47	8.67	0.40	0.68882
ROOTSTOCKM7	7.32	8.30	0.88	0.37801
ROOTSTOCKM793	3.90	8.20	0.48	0.63481
ROOTSTOCKMM106	-1.64	11.52	-0.14	0.88674
ROOTSTOCKMM109	7.53	8.27	0.91	0.36270
ROOTSTOCKSaailing	2.72	8.99	0.30	0.76210
ROOTSTOCKSAAILING	3.43	8.27	0.41	0.67861

TABLE A.1: Regression coefficients and associated p-values for the final MLR model

Bibliography

Anthony, Brendon, Sara Serra, and Stefano Musacchi (2020). "Optimization of Light Interception, Leaf Area and Yield in "WA38": Comparisons among Training Systems, Rootstocks and Pruning Techniques". In: *Agronomy* 10.5, p. 689.

Atucha, Amaya, Ian A Merwin, and Michael G Brown (2011). "Long-term effects of four groundcover management systems in an apple orchard". In: *HortScience* 46.8, pp. 1176–1183.

Bacco, Manlio et al. (2019). "The digitisation of agriculture: a survey of research activities on smart farming". In: *Array* 3, p. 100009.

Balducci, Fabrizio, Donato Impedovo, and Giuseppe Pirlo (2018). "Machine learning applications on agricultural datasets for smart farm enhancement". In: *Machines* 6.3, p. 38.

Bates, Richard Pierce, Justin R Morris, and Philip G Crandall (2001). *Principles and practices of small-and medium-scale fruit juice processing*. 146. Food & Agriculture Org.

Bengio, Yoshua, Ian Goodfellow, and Aaron Courville (2017). *Deep learning*. Vol. 1. MIT press Massachusetts, USA:

Bishop, Christopher M (2006). *Pattern recognition and machine learning*. springer.

Boehmke, Brad and Brandon M Greenwell (2019). *Hands-on machine learning with R*. CRC Press.

Bravin, E, A Kilchenmann, and M Leumann (2009). "Six hypotheses for profitable apple production based on the economic work-package within the ISAFRUIT Project". In: *The Journal of Horticultural Science and Biotechnology* 84.6, pp. 164–167.

Breiman, Leo (2001). "Random forests". In: *Machine learning* 45.1, pp. 5–32.

Bronson, K and I Knezevic (2016). *Big Data in Food and Agriculture, Big Data & Society*.

Bruce, Peter, Andrew Bruce, and Peter Gedeck (2020). *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*. O'Reilly Media.

Chaves, Bernardo et al. (May 2017). "Modeling fruit growth of apple". In: *Acta Horticulturae* 1160, pp. 335–340. DOI: 10.17660/ActaHortic.2017.1160.48.

Cheng, Hong et al. (2017). "Early Yield Prediction Using Image Analysis of Apple Fruit and Tree Canopy Features with Neural Networks". In: *Journal of Imaging* 3.

Du, Ke-Lin and Madisetti NS Swamy (2013). *Neural networks and statistical learning*. Springer Science & Business Media.

Duan, Naihua et al. (1983). "A comparison of alternative models for the demand for medical care". In: *Journal of business & economic statistics* 1.2, pp. 115–126.

FAO (2021). *FAOSTAT Apple Production Data*. URL: http://www.fao.org/faostat/en/#data/QC (visited on 03/20/2021).

Fukuda, Shinji et al. (2013). "Random Forests modelling for the estimation of mango (Mangifera indica L. cv. Chok Anan) fruit yields under different irrigation regimes". In: *Agricultural water management* 116, pp. 142–150.

Glorot, Xavier, Antoine Bordes, and Yoshua Bengio (2011). "Deep sparse rectifier neural networks. volume 15 of". In: *Proceedings of Machine Learning Research*.

Gonzalez-Sanchez, Alberto, Juan Frausto-Solis, and Waldo Ojeda-Bustamante (2014). "Predictive ability of machine learning methods for massive crop yield prediction". In: *Spanish Journal of Agricultural Research* 12, pp. 313–328.

Goossens, Yanne et al. (2017). "Life cycle assessment (LCA) for apple orchard production systems including low and high productive years in conventional, integrated and organic farms". In: *Agricultural Systems* 153, pp. 81–93.

Heinicke, Donald Richard (1975). *High-density Apple Orchards: Planning, Training, and Pruning*. 458. Agricultural Research Service, US Department of Agriculture.

Hester, Susan M and Oscar Cacho (2003). "Modelling apple orchard systems". In: *Agricultural systems* 77.2, pp. 137–154.

Himesh, S et al. (2018). "Digital revolution and Big Data: a new revolution in agriculture". In: *CAB Rev* 13.21, pp. 1–7.

Ho, Tin Kam (1995). "Random decision forests". In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE, pp. 278–282.

Hortgro (2019). *Key deciduous fruit statistics*. URL: https://www.hortgro.co.za/.

Issad, Hassina Ait, Rachida Aoudjit, and Joel JPC Rodrigues (2019). "A comprehensive review of Data Mining techniques in smart agriculture". In: *Engineering in Agriculture, Environment and Food* 12.4, pp. 511–525.

Jafta, Asanda (2014). "Analysing the competitiveness performance of the South African apple industry". PhD thesis.

James, Gareth et al. (2013). *An Introduction to Statistical Learning: with Applications in R*. Springer. URL: https://faculty.marshall.usc.edu/gareth-james/ISL/.

Jeong, Jig Han et al. (2016). "Random forests for global and regional crop yield predictions". In: *PLoS One* 11.6, e0156571.

Jiang, Dong et al. (2004). "An artificial neural network model for estimating crop yields using remotely sensed information". In: *International Journal of Remote Sensing* 25.9, pp. 1723–1732.

Kaack, K, H Lindhard Pedersen, et al. (2010). "Prediction of diameter, weight and quality of apple fruit (Malus domestica Borkh.) cv.'Elstar'using climatic variables and their interactions". In: *European Journal of Horticultural Science* 75.2, p. 60.

Kamilaris, Andreas, Andreas Kartakoullis, and Francesc X Prenafeta-Boldú (2017). "A review on the practice of big data analysis in agriculture". In: *Computers and Electronics in Agriculture* 143, pp. 23–37.

Karami, Mokhtar, Mehdi Asadi, et al. (2017). "The Phenological Stages of Apple Tree in the North Eastern of Iran". In: *Computational Water, Energy, and Environmental Engineering* 6.03, p. 269.

Khaki, Saeed and Lizhi Wang (2019). "Crop Yield Prediction Using Deep Neural Networks". In: *Journal of Frontiers in Plant Science* 10, p. 621.

Kim, Nari et al. (2019). "A Comparison Between Major Artificial Intelligence Models for Crop Yield Prediction: Case Study of the Midwestern United States". In: *International Journal of Geo-Information* 8, p. 240.

Kutner, Michael H et al. (2005). *Applied linear statistical models*. Vol. 5. McGraw-Hill Irwin Boston.

Lakso, Alan and Martin Goffinet (2014). "Apple Fruit Growth". In: *Proceedings from the Empire State Producers Expo*. Department of Horticulture, NYSAES, Cornell University.

Lee, SoonWon et al. (2007). "A report on current management of major apple pests based on census data from farmers." In: *Korean Journal of Horticultural Science & Technology* 25.3, pp. 196–203.

Li, Meirong et al. (2019). "Possible impact of climate change on apple yield in Northwest China". In: *Theoretical and Applied Climatology* 139, pp. 191–203.

Liakos, Konstantinos G et al. (2018). "Machine learning in agriculture: A review". In: *Sensors* 18.8, p. 2674.

Liaw, Andy and Matthew Wiener (2002). "Classification and Regression by randomForest". In: *R News* 2.3, pp. 18–22. URL: https://CRAN.R-project.org/doc/Rnews/.

Lindén, Leena (2001). "Re-analyzing historical records of winter injury in Finnish apple orchards". In: *Canadian Journal of Plant Science* 81.3, pp. 479–485.

Linker, Raphael (2018). "Machine learning based analysis of night-time images for yield prediction in apple orchard". In: *Biosystems engineering* 167, pp. 114–125.

Manfrini, L et al. (2020). "Innovative approaches to orchard management: assessing the variability in yield and maturity in a 'Gala'apple orchard using a simple management unit modeling approach". In: *European Journal of Horticultural Science* 85.4, pp. 211–218.

Middleton, Simon et al. (2002). "The productivity and performance of apple orchard systems in Australia". In: *Compact Fruit Tree* 35.2, pp. 43–47.

Min, Yongyi and Alan Agresti (2002). "Modeling nonnegative data with clumping at zero: a survey". In:

Mishra, Subhadra, Debahuti Mishra, and Gour Hari Santra (2016). "Applications of Machine Learning Techniques in Agricultural Crop Production: A Review Paper". In: *Indian Journal of Science and Technology* 9 (38).

Montgomery, Douglas C, Cheryl L Jennings, and Murat Kulahci (2015). *Introduction to time series analysis and forecasting*. John Wiley & Sons.

Paudel, Dilli et al. (2021). "Machine learning for large-scale crop yield forecasting". In: *Agricultural Systems* 187, p. 103016.

Reig, Gemma et al. (2018). "Horticultural performance and elemental nutrient concentrations on 'Fuji'grafted on apple rootstocks under New York State climatic conditions". In: *Scientia Horticulturae* 227, pp. 22–37.

Russello, Helena (2018). "Convolutional neural networks for crop yield prediction using satellite images". In: *IBM Center for Advanced Studies*.

Shankarnarayan, Vinay Kellengere and Hombaliah Ramakrishna (2020). "Paradigm change in Indian agricultural practices using Big Data: Challenges and opportunities from field to plate". In: *Information Processing in Agriculture*.

Smithing, N (1999). *Supervised Learning in Feedforward Artificial Neural Networks*.

Srivastava, Nitish et al. (2014). "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1, pp. 1929–1958.

Stajnko, Denis and Vanja Švagan (2009). "Improvement of apple yield forecast accuracy with additional sampling". In: *Proceedings. 47th Croatian and 7th International Symposium on Agriculture. Opatija. Croatia*. Vol. 821, p. 826.

Stanley, CJ, JR Stokes, and DS Tustin (2000). "Early prediction of apple fruit size using environmental indicators". In: *VII International Symposium on Orchard and Plantation Systems 557*, pp. 441–446.

Tahir, Ibrahim I, Eva Johansson, and Marie E Olsson (2007). "Improvement of quality and storability of apple cv. Aroma by adjustment of some pre-harvest conditions". In: *Scientia horticulturae* 112.2, pp. 164–171.

United Nations, World Water Assessment Programme (2003). *Water for People, Water for Life: The United Nations World Water Development Report: Executive Summary*. Unesco Pub.

Wright, Marvin N. and Andreas Ziegler (2017). "ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R". In: *Journal of Statistical Software* 77.1, pp. 1–17. DOI: 10.18637/jss.v077.i01.

WSU (2021). *Tree Fruit Pruning and Training Systems*. http://treefruit.wsu.edu/orchard-management/pruning-and-training-systems/. [Online; accessed 23-March-2021].