USING MACHINE LEARNING BASED EMULATORS FOR SENSITIVITY ANALYSIS OF PROCESS-DRIVEN BIOPHYSICAL MODELS

A Thesis submitted by

David B Johnston

BAgrSc (Hons), BAppSc

For the award of

Doctor of Philosophy

2022

ABSTRACT

Sensitivity Analysis (SA) is a versatile and well-established tool used in the development and application of computer models. Although considered an integral part of the modelling process in multiple disciplines, its use in the development of process-driven biophysical models is relatively rare. One contributing reason for this lack of use is the computational burden associated with performing SA on complex models. Literature reports examples of the use of emulators, or metamodels, as an approach for reducing the computational burden of complex models, but there are no reports of using machine learning based emulators for undertaking SA of the underlying process-driven biophysical models. This doctoral thesis explores the potential of machine learning emulators (MLEs) in reducing the computational burden of performing SA on process-driven biophysical models. Firstly, a new method is developed that confirms that the variable importance indices of MLEs are comparable to the sensitivity indices produced by the commonly used Morris and Sobol methods. This provides the confidence upon which to proceed with investigating further the role that MLEs might play in reducing the computational burden of SA. Secondly, three different machine learning (ML) algorithms are used to generate MLEs of the APSIM-NextGen chickpea model to evaluate if some MLEs are better suited to the task of emulating process-driven biophysical crop models. The MLEs were assessed on accuracy of predicted values and the computational effort required to develop the MLEs themselves. The emulators based on random forest models were shown to produce the most accurate predictions, but also required the most computational effort to develop and train. Thirdly, two MLEs are used to undertake SA of all 22 input parameters of the MLEs, as well as a selected subset of six input parameters linked to the phenology of the crop. These analyses required more than 40 million simulations to be run. The MLEs were assessed based on their speed of execution, and on the Morris and Sobol indices produced. The impressive computational speed of the MLEs was quantified in comparison to the speed of the process-driven biophysical model. Some discrepancies were also noted between the results generated by the two types of MLE, so no firm conclusions could be made about the sensitivities of the underlying process-driven model. This work is at the juncture of the fields of process-driven biophysical model development, agronomy, plant physiology, machine learning emulators, and global sensitivity analysis. The

outcomes of this work have implications for model development and model application in all these disciplines. Firstly, the Morris method remains a more computationally efficient choice, when compared with the development and use of MLEs, for the screening of importance of parameters of process-driven models. Secondly, the results show that, while both Morris and Sobol analyses produce very similar results across different MLEs, the discrepancies indicate that great caution is needed if interpreting these results as a way of understanding the underlying process-driven model and its input-output sensitivities. The results suggest that by using the computational efficiency of an MLE, SA of large-scale simulation experiments becomes more feasible, and this can contribute to efficiency gains for scientific research. The SA of enhanced forms of simulation experiments produced by hybrid models, which use the outputs of process-driven models and combine these with other sources of data to create new forms of ML based agro-ecological models, is suggested by this research as a direction that could be perused to advance agro-ecological modelling. This work has demonstrated how applied research in these areas, when combined, can better serve the needs of researchers and modelling practitioners alike.

CERTIFICATION OF THESIS

This Thesis is the work of David B Johnston except where otherwise acknowledged, with the majority of the authorship of the papers presented as a Thesis by Publication undertaken by the Student. The work is original and has not previously been submitted for any other award, except where acknowledged.

Principal Supervisor: Assoc. Professor Keith Pembleton

Associate Supervisor: Professor Ravinesh C. Deo

Associate Supervisor: Dr Neil Huth

Student and supervisors' signatures of endorsement are held at the University.

ACKNOWLEDGEMENTS

I would like to acknowledge and express my appreciation of the following people and organisations:

I owe a debt of gratitude and special thanks to my supervisory team:

- Associate Professor Keith Pembleton for his wise counsel, guidance, patience, and prodding motivation which all assisted me completing this thesis.
- Professor Ravinesh Deo for his teaching and guidance on machine learning and his excellent editorial skills.
- Dr Neil Huth for his guidance in the design of the APSIM system and the design and testing of sensitivity analysis experiments, and for his patience in persevering with me through some very slow periods of progress.

The University of Southern Queensland for allocating me a PhD place, for their material support and guidance in undertaking this candidature.

This research has been supported by an Australian Government Research Training Program Scholarship.

This research has been supported by a Queensland Government Advance Queensland PhD Top-Up Scholarship.

Acknowledgment is made to the APSIM Initiative which takes responsibility for quality assurance and a structured innovation programme for APSIM's modelling software, which is provided free for research and development use (see www.apsim.info for details).

Dean Holzworth and Drew Holzworth for their encouragement and support while providing software expertise and enhancing features in the APSIM-NextGen framework in support of my studies.

I also wish to thank Dr Allan Peake of CSIRO Toowoomba, for kindly providing the base APSIM-NextGen chickpea simulation configuration and the guidelines for simulating chickpea production in each of the Australian chickpea growing regions. These guidelines and concepts formed the basis of the APSIM experimental design used in this study.

A special thanks to my family members who so graciously provided formatting and proofreading assistance at crucial times.

Finally, my deepest thanks and appreciation goes to my loving wife, Joan, who has faithfully and lovingly provided her support to let me pursue my goals over many years. This goal would never have been attainable without your constant encouragement, patience, and support. I thank you. To my son, Tim, I express my appreciation and thanks for all the time that you have allowed me to be missing from family activities in order to complete these studies.

STATEMENT OF CONTRIBUTION
FOR MATERIALS SUBMITTED FOR PUBLICATION

Paper 1:

> Johnston, DB, Pembleton, KG, Huth, NI & Deo, RC 2022, **[in review]** 'Evaluation of the effectiveness of using machine learning emulators for sensitivity analysis of process driven models: A case study using a model of chickpea phenology', *Stochastic Environmental Research and Risk Assessment (SERRA).*

Student, David Johnston, contributed 85% to this paper. Collectively, Keith Pembleton, Neil Huth and Ravinesh C Deo, contributed the remainder.

Paper 2:

> Johnston, DB, Pembleton, KG, Huth, NI & Deo, RC 2022, **[in review]** 'Comparison of machine learning methods emulating process driven crop models', *Environmental Modelling & Software*.

Student, David Johnston, contributed 85% to this paper. Collectively, Keith Pembleton, Neil Huth and Ravinesh C Deo, contributed the remainder.

Paper 3:

> Johnston, DB, Pembleton, KG, Huth, NI & Deo, RC 2022, **[in review]** 'Sensitivity analysis of process driven biophysical models using machine learning emulators', *Environmental Modelling & Software*.

Student, David Johnston, contributed 85% to this paper. Collectively, Keith Pembleton, Neil Huth and Ravinesh C Deo, contributed the remainder.

TABLE OF CONTENTS

ABBREVIATIONS

AI	artificial intelligence
ANN	artificial neural network
APSIM	Agricultural Production Systems Simulator
APSIM-NextGen	Agricultural Production Systems Simulator - Next Generation
ARES	adaptive regression splines (MATLAB package)
CI	confidence interval
COE_{LM}	coefficient of efficiency (Legates-McCabe index)
DAS	days after sowing
DD	degree days (thermal time in heat units)
DOY	day of year
DSSAT	Decision Support System for Agrotechnology Transfer
EE	elementary effects
EET	elementary effects test
ESW	extractable soil water
GSA	global sensitivity analysis
KNN	k-mean nearest neighbour
LSA	local sensitivity analysis
MAE	mean absolute error
MARS	multivariate adaptive regression splines
MB	mean bias
ML	machine learning
MLE	machine learning emulator
NASA	National Aeronautics and Space Administration
PAWC	plant available soil water
R^2	coefficient of determination
RF	random forest
RMSE	root mean squared error
SA	sensitivity analysis
S_i	Sobol primary effects index
S_{ij}	Sobol secondary effects index
S_T	Sobol total effects index
SOM	self-organising map
SVN	support vector machine

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Background

Developing process-driven biophysical models is a time consuming and expensive task. Over the past fifty years, agricultural models have progressed from being simple fertiliser versus yield calculator models and price by area profit calculators, to highly complex tools that integrate knowledge across multiple science disciplines. These approaches have provided ever greater levels of detail in applications for improved agricultural practices (Jones et al. 2016). Models vary from the simple regression equations, for example, the recommendations derived for fertiliser application rates for predicted yield levels, and the effect of time of sowing on the yield, to the complex representation of farming systems such as the APSIM modelling system (Holzworth et al. 2014; Holzworth et al. 2018), DSSAT (Jones et al. 2003), STICS (Brisson et al. 2003) and the CropSyst (Stöckle et al. 2003) modelling systems. In the Australian context, the preeminent tool for modelling agricultural systems is the APSIM framework (Holzworth et al. 2014; Holzworth et al. 2018). Much of the scientific knowledge built into the sub-models that comprise these complex modelling systems in agriculture, is based on the findings of carefully controlled experiments which have supported the development of mechanistic models of the processes involved (Jones et al. 2016). For example, research into the phenological process involved in legume crop growth and development by Robertson et al. (2002) underpins the APSIM-NextGen chickpea model. More complex processes have been modelled from data gathered to describe functional relationships, with empirical equations developed to approximate the underlying complexity. Examples of these models include the well-known Penman-Monteith (Allen et al. 1998) equation used for the calculation of reference evapotranspiration and the Priestley-Taylor (Priestley & Taylor 1972) functional equation used for the prediction of potential evapotranspiration. Alternative approaches for producing models that simulate the world around us are continuously being developed, tested, and improved. Currently, however, there are no viable alternatives that are ready to replace process-driven modelling systems in their entirety. The development of

process-driven biophysical models and modelling systems continues to be an area of significant scientific research.

The aforementioned process-driven models have evolved in their ability to describe biophysical processes providing an ever increasing level of detail, while researchers have also combined more layers of models to simulate ever larger and more complex agricultural systems (Brown et al. 2014). As a result of this increased complexity, the number of parameters required to define the simulations in these systems, and the computational burden to run these simulations, has continued to increase. Another problem inherent with this ever-increasing complexity is that the verification of model integrity and accuracy of model outputs is also increasingly challenging (Huth & Holzworth 2005).

Sensitivity analysis (SA) is a widely accepted technique used in model verification activities. A number of prominent leaders in a range of science, mathematics, policy, and economic disciplines agree that the inclusion of SA in systems-based modelling must become a best practice standard, but they are also acutely aware of the discrepancy between the theory and the practice of model verification and validation (Razavi et al. 2021). Advances in computational capacity, whether it be due to improvements in hardware performance or software algorithms, are now facilitating new trends in SA research, including, but not limited to response surface surrogates, polynomial chaos expansions (Liu & Choe 2021), and the grouping of parameters into success or failure sets (Bachoc et al. 2020). However, the tools and techniques developed in one discipline of research are not readily taken up in the other disciplines (Razavi et al. 2021). This has been attributed to a range of reasons including, inconsistent terminology and historical bias. Another aspect is the computational burden of generating and analysing large data sets, an issue identified as one of the significant barriers to the uptake of applying comprehensive SA to process-driven modelling systems (Razavi et al. 2021). The advances made in the rapidly expanding areas of artificial intelligence (AI) and machine learning (ML) offer efficiency gains for the processing and analysis of the large data sets (Gollapudi 2016; Ramstein et al. 2019; Dokic et al. 2020; Saiaa et al. 2020). Although ML methodologies are ideally suited to investigating problems involving large data sets, there is currently very limited published research into any approaches

for applying SA based on ML. The use of ML for developing new crop models and approaches for agro-ecological modelling is being reported in recent studies (refer to Chapter 2, e.g., Deo and Şahin (2015b); Kouadio et al. (2018)). However, these approaches generally avoid or bypass the use of the process-driven biophysical modelling systems, such as APSIM, which are widely used in Australia and global contexts. The power, versatility, and proven fit-for-purpose standing of process-driven modelling systems is unlikely to be replaced in the short-term (Holzworth et al. 2015). If SA is to truly become an integral part of simulation model development and validation, then SA of process-driven biophysical modelling systems must remain a focus for research. This doctoral research thesis aims to investigate the application of rapidly developing ML methods to the existing problems of parametrisation of biophysical crop models, and to the increasingly important issue of new methods required for SA of process-driven modelling systems.

1.2 Research aim

While much ML research is focused on developing new modelling approaches using new or enhanced data sources, this research looks at the application of the developing area of ML to the existing and increasingly important problem of applying SA to process-driven modelling systems. The aim of this research is to assess if ML based emulators can be used in reducing the computational burden of performing SA on process-driven biophysical models. A review of relevant literature will explore the current state of knowledge around combining process-driven modelling approaches with ML modelling approaches.

1.3 Outline of thesis

This thesis is a seminal research work presenting an investigation into the practicalities, applicability, benefits, and limitations of using ML-based emulators built to perform SA of process-driven biophysical crop models. In this work the methods are applied to the APSIM-NextGen chickpea model as an example. In Chapter 2, the most relevant literature is reviewed that describes the following aspects of this research work:

- Latest approaches in process-driven biophysical model development and relevant testing of applications;
- The evidence-based arguments for sensitivity analysis to be included in the development, validation, and application phases of the lifecycle of process-driven biophysical models;
- A review of the most common sensitivity analysis approaches used in biophysical model development, including the Morris and the Sobol methods, is undertaken;
- The role of emulators in addressing the computational burden is discussed; and
- A discussion on ML algorithms currently used in agricultural applications is presented.

Chapter 3 is devoted to the development of a method that allows the sensitivity measures produced by the Morris and the Sobol methods to be compared, via a cross validation approach, with the measures produced by a ML based emulator of the process-driven biophysical model. This initial step was required to verify that the MLEs can produce sensitivity analysis that is comparable to the analysis produced by standard techniques used on the underlying process-driven model. Chapter 4 considers three different ML models and further assesses their suitability, in terms of computational performance and the accuracy of their predictions, as emulators of the process-driven biophysical APSIM-NextGen chickpea model. The MLEs were developed for six APSIM outputs and trained on seven chickpea production locations. Accuracy of predictions was assessed for spatial and temporal variations in input values. In Chapter 5, the two emulators derived from Chapter 4 which were assessed as suitable for further investigation, were used to perform the Morris and the Sobol analyses on both the full set of 22 input variables used to develop and drive the emulators, as well as a selected subset of six input variables. The computational efficiency of the emulators has been demonstrated in this chapter, along with the comparative accuracy of the two MLEs. Some limitations of this approach for the sensitivity analysis of process-driven models have also been identified. In Chapter 6, the most significant findings from Chapters 3 to 5 are brought together to summarise the results, and observations made in respect to the

feasibility, applicability, and the limitations of using ML emulators to assist in the sensitivity analysis of process-driven biophysical models are presented. The practical implications of this study are discussed in light of the evidence derived from results presented earlier in the thesis.

CHAPTER 2: LITERATURE REVIEW

2.1 Scope of review

It is generally acknowledged that biophysical models provide a valuable tool in the study of the components and interactions of environmental, biological, and man-made systems. Modelling of agricultural production systems had its beginnings as far back as the 1940's and 1950's (Jones et al. 2016) and continues to expand even today in its complexity and focus of application. There has been a progression from modelling detailed functional processes and relationships, to crops, to whole farm systems and then further to include off farm environmental, economic and policy modelling (Robertson et al. 2015; Jones et al. 2016). This progression has been matched with an increase in the complexity of the modelling systems and the number of input parameters required to drive the models (Holzworth et al. 2015). Regardless of the simplicity or complexity of the model, the model outputs need to be tested and validated with real datasets (Montesino-San Martin et al. 2018). Low level modelled processes are generally well specified and validated, but as models evolve in size and complexity, processes and sub-systems will interact with each other and affect the performance of the whole system (Hieronymi 2013). Given that the performance of a model as a component in a larger modelling system are likely to be affected by the interactions with other components in that system, testing of complex models and the modelling systems that they operate in, is an ongoing challenge. It is important to establish appropriate methods with which to test the quality of the modelling system as a whole (Bellocchi et al. 2010; Kersebaum et al. 2015). In this thesis, the term 'testing' and 'tested' refer to the processes of reviewing data and models for correctness, while the term 'validated' is used to refer to data and models that have been tested and assessed to be correct or produce correct outputs. Testing may not result in the assessment of validated.

There are many statistical (linear and non-linear) methods that have been developed to assist in the development and testing of models. One approach which has been widely utilised is that of sensitivity analysis (SA). In its most simple form, SA tests the amount of change in a given output value for given change in a single input

parameter. In practice, with potentially hundreds of input parameters in a crop model, the computational requirement for testing all levels for all input parameters against all other input parameter values quickly becomes impossible. For example, if 5 parameters are varied across 6 levels each, then this would result in 6^5 = 7,776 simulations. If this increased to 20 parameters at 6 levels each, then there would be 3.6 x 10^{15} simulations; an infeasibly large number of simulations to run for most complex biophysical models on easily accessible computing platforms. This creates a challenging modelling environment, one where, even with a good level of expertise and due care when using the crop model, modelling outputs cannot always be assumed to have been comprehensively validated during model development (Montesino-San Martin et al. 2018).

The problem of evaluating model validity is common across all types of models from all science disciplines. Thus, research undertaken into model sensitivity and uncertainty analysis in disciplines other than agricultural systems modelling may benefit the development methodologies for crop models. Of particular interest is the rapidly advancing field of machine learning (ML). Modern computing power and new techniques present the possibility that some new analysis methodologies might be available and applicable to crop model development. This literature review will provide an outline of the significant role that agro-ecological modelling plays within science research, and the position that the Australian Agricultural Production Systems Simulator holds within this space on a global basis. A brief review of the history and role of biophysical crop modelling then leads into an outline of some of the critical steps involved in model development and testing, highlighting the critical need for, but often the absence of, the application of SA. The two most applied SA methods in the field of biophysical modelling are summarised. Computation burden of applying SA is identified as one of the main hinderances to its adoption, and the area of model emulators and metamodels is reviewed for possible solutions to the problem. The potential for ML emulators to be part of the solution is identified. To put this relatively recent discipline into context with respect to agro-ecological modelling, a brief review is made of ML concepts and how they are already being applied in this area of science. Three selected ML algorithms, which are considered most suitable for developing ML emulators of process-driven biophysical crop models, are then briefly introduced. The lack of peer reviewed literature looking at

ML emulators of process-driven biophysical models, and how such emulators might perform in reducing the computational burden associated with performing SA, is highlighted as a gap in the current body of knowledge.

2.2 Overview of agro-ecological modelling

Agro-ecological modelling is used widely throughout the world for a wide range of applications including: grower and farmer advice (Carberry et al. 2002; Peake et al. 2014), production performance analysis (Hochman et al. 2016), policy planning and assessment (Brennan et al. 2008), research into resource usage and conservation (Qureshi et al. 2013; Kersebaum et al. 2015), plant breeding (Casadebaig et al. 2016), and climate change effects and adaption research (Asseng et al. 2013; Luo et al. 2016; Pembleton et al. 2016; Shukr et al. 2021). The Agricultural Production Systems Simulator (APSIM) (McCown et al. 1995; Keating et al. 2003; Holzworth et al. 2014) is Australia's dominant agricultural systems modelling platform (Robertson et al. 2015). In 2002 a review by Donatelli et al. (2002) found APSIM and Decision Support System for Agrotechnology Transfer (DSSAT) (Jones et al. 2003) to be the two most referenced agricultural modelling systems in the world. In a later review of the grains industry in Australia (Robertson et al. 2015) it was reported that ~95% of grains industry simulation modelling in Australia was with the APSIM framework. Development of the APSIM modelling system remains ongoing with much effort now going into an updated software infrastructure called APSIM-Next Generation (APSIM-NextGen) (Holzworth et al. 2018).

2.3 Biophysical crop modelling

Biophysical modelling allows mechanistic descriptions of biological and physical processes to be used to provide quantitative predictions of functional outcomes. These mathematical models are usually based on known physical, chemical, or biological control processes occurring in crop production systems. The mathematical equations which describe them are developed from measured data generated from controlled experiments conducted in laboratories, glasshouses, or farm fields. These processes often have many model parameters associated with them and their

functional limits are generally well understood. Examples of these processes include photosynthesis, nutrient and water uptake by roots, translocation within plants. Complex processes, such as radiation use efficiency and crop evapotranspiration, are generally based on empirical functions which have been developed by modelling relationships based on observed data. These empirical functions often involve greater degrees of approximation than do the mechanistic model functions and can be the source of greater model uncertainty (Jones et al. 2016). Development and improvement of biophysical models requires areas of uncertainty, wherever they are in the model, to be identified and understood before resources are expended on generating and collecting data sets whose purpose is to aid in further model development.

2.4 Model development and testing

One of the many changes being implemented as part of the development of APSIM-NextGen is the requirement that crop models be developed in a common Plant Modelling Framework (PMF) (Brown et al. 2014). The PMF takes a high-level view of plants as whole organisms and defines the major organs, structures and processes that are common across almost all plants. Different plants are modelled, or defined, by using different parameter settings for the various components of the framework. This framework is an attempt to acknowledge the commonality in the processes and growth of plants and in the way crop models represent them. By using a common framework, the modelling system developers can achieve a much greater reuse of high quality, maintainable code and consistent functional interfaces (Holzworth et al. 2014; Holzworth et al. 2018). Model developers and users can rely on consistent approaches to biological and physiological processes, which can assist with more consistent data sets for calibration, validation and ultimately parameter sets for model use. As well as requiring all new crop models that are developed for APSIM-NextGen to be designed and implemented in the PMF, the APSIM Initiative has also specified that models must have substantial validation data sets and have validation testing sets set up before they will be considered for inclusion in a released production version of APSIM- NextGen

(https://apsimnextgeneration.netlify.app/development/science/testing/, retrieved December 2021).

Although it is considered 'best-practice' for SA to be a fundamental part of the analysis of model performance ((Saltelli & Annoni 2010; Plischke et al. 2013), the reported use of SA is relatively scarce in crop modelling literature. Other science disciplines, such as Hydrology modelling, have a much greater reported use of the techniques. It is not immediately clear why this is the case, though Bellocchi et al. (2010) suggests that the high heterogeneity of input data and the number of influencing input parameters in biophysical models, which leads to very large data set requirements for SA, could act as a disincentive to modellers following the recommended best practices. Almost all reviews of SA acknowledge the computational burden that performing such analysis carries with it (Christopher Frey & Patil 2002; Iooss & Lemaître 2015; Norton 2015; Stanfill et al. 2015; Razavi et al. 2021). Computational efficiency is one of the key measures that needs to be considered in any assessment of SA for biophysical models.

2.5 Sensitivity analysis

Sensitivity analysis is acknowledged and recommended as the best practice by international agencies, such as the National Aeronautics and Space Administration (NASA) and national regulatory agencies, for the audit, validation, and application of scientific models (Saltelli & Annoni 2010; Plischke et al. 2013). While published literature demonstrates the application of a few SA methodologies to biophysical models, and specifically agricultural systems models, SA is not observed to be widely applied in the development processes of crop models. Indeed, Razavi et al. (2021) argue that despite SA research and practice gaining significant momentum in a range of science disciplines over the past few years, its benefits and true potential has not been realised. This lack of application of best practice methodologies can be attributed to a number of factors, with the computational expense of performing any single analysis probably chief amongst them (Bellocchi et al. 2010). With complex models having many tens, if not hundreds of input parameters, identifying the key inputs to analyse and knowing which SA methodologies are most suitable for

addressing which model data or model design questions compounds the issues of having to run potentially millions of simulations to perform a thorough analysis.

Sensitivity analysis can be broadly grouped into two categories based on the number of input factors being analysed and the scope of values that are tested. Local sensitivity analysis (LSA) takes the form of one or a few input factors being varied around a nominal point in the problem space. Local minima and maxima of functional relationships can be identified, but its application is limited to answering very specific questions (Saltelli & Annoni 2010). Global sensitivity analysis (GSA) is the more generally applied approach where the influences on some output value are assessed across an entire, or global, problem space (Saltelli et al. 2000).

There are a wide range of statistical approaches that can be used when approaching the problem of performing SA. Pianosi et al. (2016) reviewed a range of SA approaches used for analysis of environmental models. These approaches, which are also applicable for the SA of biophysical crop models, ranged from the relatively simple and computationally efficient Elementary Effects Test (EET) methods which are most applicable for qualitative screening of input parameters, to the much more computationally demanding variance-based analysis approaches which are more suitable for the detailed analysis of a smaller number of input parameters. Both of these are GSA techniques, and each approach has its strengths and limitations. Of the EET approaches, the Morris method (Morris 1991) and its revised version (Campolongo et al. 2007) have gained a level of popularity as shown by its use in a range of simulation experiments (Casadebaig et al. 2016; Sarrazin et al. 2016; Pardon et al. 2017; Jaxa-Rozen & Kwakkel 2018). For variance-based SA approaches, the most established method is that of Sobol' (Sobol' 1993). Razavi and Gupta (2015) and Yang (2011) both acknowledge the prevalent use of the Morris method, and the variance-based analysis method of Sobol' (Sobol' 2001) in the environmental sciences. These two SA approaches, the Morris method and Sobol method, are tried and tested approaches which have been applied in area of agro-ecological modelling (Ascough II et al. 2004; DeJonge et al. 2012; Iooss & Lemaître 2015; Pianosi et al. 2016; Thorp et al. 2020).

2.5.1 Morris method: Elementary Effects Test

The Morris method (Morris 1991), later enhanced by Campolongo (Campolongo et al. 2007) is also called the Elementary Effects Test (EET) (Saltelli & Annoni 2010). It is often used for computationally efficient screening of larger numbers of input parameters to determine their importance for a target output. Analysis can be used to determine which input factors may be considered as having effects which are (a) negligible, (b) linear and additive, or (c) non-linear or involved in interactions with other factors (Campolongo et al. 2007), and it has been found to be robust and suitable as a SA screening tool for environmental models (Sarrazin et al. 2016). Two sensitivity measures are computed by this method: μ, which assesses the overall influence of the input factor on the output, and σ, which estimates a factor's level of non-linearity and/or interactions with other factors. The required sample size to reliably estimate the input factors' contribution to the output value is generally 10 to 100 times the number of input factors (Pianosi et al. 2016). The Morris method has been used by APSIM crop modellers (Casadebaig et al. 2016; Pardon et al. 2017). The method has limitations in that it assumes linear response functions for input parameters and that input parameters are independent of each other. It also does not reveal second or higher order interactions between input parameters.

2.5.2 Sobol method: Variance decomposition approach

The Sobol method (Sobol' 1993) is a variance decomposition method of SA. This approach of SA breaks down variation in an output value to attribute the effects to variations in individual input parameters and interactions between input parameters. The default analysis produces two sensitivity indices, the primary effects index (S_1) and the total effects index (S_T) for each input parameter. The S_1 index is a measure of that input parameter's influence, by itself, on the value of the output as a proportion of the total variation in the output. The Sobol SA method also allows the evaluation of higher-order sensitivity indices which demonstrate the effects of parameter interactions. The S_T index is a measure of the effect that the input parameter has, both by itself and in combination with all other input parameters, on the value of the output as a proportion of the total variation of the output parameter. The value of S_T is most comparable to the measure associated with μ of the Morris

method, although it is expressed as a proportion of the output's variation while μ is expressed in the units of the output parameter. Sobol' has a much higher computational cost than the Morris method. While there are no strict guidelines as to the sample size required to obtain reliable sensitivity indices, Pianosi et al. (2016) suggests that the sample size should be approximately 1000 times the number of input factors, or greater. Apart from a demonstration paper (Stanfill et al. 2015) where the Sobol method is compared to an emulator's performance for SA of selected characteristics of APSIM-wheat, the lack of papers published detailing the use of these methods for the analysis of APSIM models points to a lack of SA being used in crop model development and application. This highlights the breadth of the research gap in this area.

2.6 Emulators and metamodels

In this thesis, the term 'algorithm' is used to refer to the logic or design of a method, while 'model' is used to refer to a functioning implementation of the code developed by applying the algorithm to a given set of data. 'Emulator' is used to refer to a model that has been developed to emulate the functioning, or part thereof, of a more complex model.

The issue of the computational burden imposed by performing SA is not new. One approach that is quite widely reported in literature for the improvement of computational efficiency when undertaking computer simulation experiments or analyses is the use of emulators (also referred to in the literature as metamodels and surrogate models). The specific terminology used to refer to these tools is largely dependent upon the science discipline in which the research work was undertaken. Razavi et al. (2012) presented a comprehensive review of the surrogate, or emulator models used in the science discipline of water resources. Generally, the purpose of these emulators was to simplify an original complex model or function and produce statistically consistent outputs with less computational effort. The trade-off usually being the limiting of input options or range of values and to some extent the accuracy of the analyses. While Razavi et al. (2012) noted the use of artificial neural networks (ANN) and support vector machines (SVM) for the generation of surrogate

models, a special issue of The European Journal of Agronomy (Volume 88, 2017) 'Uncertainty in Crop Model Predictions', which summarised the 'current' research being conducted into uncertainty and SA of crop models, does not include any references to machine learning (ML) based emulators. This indicates, at least in part, the lack of, or slow transference of knowledge and techniques between different science disciplines. Data generation and data analysis are key components of the development, validation, and application of complex models. The advancement of computing power and ongoing developments in Artificial Intelligence (AI) and ML research, especially around big datasets that are required to improve complex process driven models, presents a field of potential alternate approaches for data generation and analysis that might yield computational gains and time savings for the developers and end-users of simulation modelling systems. At this time ML emulators have not been used to explore improving the efficiencies of conducting SA for biophysical crop modelling. A potential approach to improving the computational efficiencies of conducting SA is to develop emulators to accurately predict the outputs of biophysical models. Then, taking the inputs of the process driven model that are of interest to the SA, to use these as the inputs to the ML based emulators to predict the output target values. Analysis methods, such as the Sobol method, require very large numbers of simulations to be evaluated. If these simulations can be evaluated by ML emulators in a fraction of the time taken to undertake the same simulations using the process driven models directly, then gains should be possible in reducing the computational burden of undertaking the SA. Alternatively, ML models can analyse which input parameters are of most significance in calculating the output values. If these input parameter importance indices can be shown to be comparable to standard statistical indices of importance, such as Morris or Sobol indices, then additional simulation runs might not be required to analyse the sensitivity relationships between input and output parameters.

2.7 Overview of machine learning concepts

It is beyond the scope of this thesis to explore the considerable volume and depth of knowledge that constitutes the research areas of AI, and the more specialised subarea of ML. The intention in this thesis is to use 'off-the-shelf' existing ML algorithms,

or tools, and test if they can be used to improve the efficiencies of SA of biophysical crop modelling. To this end a general overview of where these algorithms sit within the ML landscape is appropriate.

Machine learning is a term used to describe the ability of computer systems to 'learn' to perform new functions without having a program written to specifically solve the provided problem (Buchanan & Miller 2017). There are three main categories of ML algorithms: (a) supervised learning, (b) unsupervised learning, and (c) reinforcement learning. Supervised learning algorithms use labelled data sets of discrete or continuous variables to 'learn' what input values create what output values. The models produced can be used to make predictions of output values. Training and testing of these models requires large sets of data that contain both the inputs and the matching output values. Some of the most commonly encountered ML algorithms that can be used for generating supervised learning models are artificial neural networks (ANN), random forests (RF) and support vector machines (SVM). Emulators of process driven models, as developed in this research, will fall into the category of supervised learning models.

Unsupervised learning algorithms are used to develop classification and clustering tools. Data need to be of a discrete, categorical nature with associated properties. There is no concept of a generated output from the input values. Algorithms which fall into this category include self-organising maps (SOM) and k-mean nearest neighbour (KNN). Tools built on unsupervised learning models are especially useful in pursuits such as data mining and image analysis (Buchanan & Miller 2017). The third category of ML algorithm, reinforcement learning, is a more specialised branch of ML which is utilised primarily in robotics and gameplay (Kaelbling et al. 1996; Goodfellow et al. 2016).

When considering performing SA on process driven models, the computational burden of generating large data sets is often one of the primary concerns. Data are the key to ML models (Jordan & Mitchell 2015; Gollapudi 2016). Their emergence in the computing and mathematical communities has, in part, been driven by a need to manage, manipulate and analyse huge amounts of data and extract useful features, patterns and correlations between covariates and a target variable. They are referred

to as ‘data intelligent’ tools because of their focus and reliance on data and the information embedded therein. Statistical models can summarise large datasets (Liu et al. 2015), but they operate in a very rigid conceptual framework in comparison to the data-driven, data-intelligent and unprogrammed framework of ML models. It is these features – ability to work with huge datasets, and an ability to be driven by data features rather than a predetermined mathematical program, that make ML approaches of particular interest for research in the area. The ability to generate an ML based emulator for a given set of process driven model’s input values, and have it accurately predict the output target values, is a key requirement in potentially using ML emulators for undertaking SA of process driven models. The use of ML models to assist in SA should be a candidate for research in this space.

2.8 Application of ML in agro-ecological modelling

Machine learning and AI are not uncommon in agricultural applications. Remote sensing imagery is becoming commonly used in large scale crop yield prediction (Bocca & Rodrigues 2016; Kuwata & Shibasaki 2016; Stas et al. 2016), biomass estimation (Ali et al. 2015; Ali et al. 2016) and crop health monitoring (Behmann et al. 2015). Machine learning algorithms are also being used for the creation of new predictive models for agriculture. Rainfall prediction models (Acharya et al. 2013; Deo & Şahin 2015b), evaporation prediction (Deo & Şahin 2015a; Deo et al. 2015), drought (Deo & Şahin 2015a; Deo et al. 2017) and solar radiation estimates (Şahin et al. 2014) have all been modelled. These applications have a close fit to the potential application of ML algorithms in crop model SA and uncertainty analysis as they are creating models which emulate complex biophysical systems. From the literature reviewed, a few classes of ML algorithms have been the focus for developing these emulators. Specifically, artificial neural networks (ANN), extreme learning machines (ELM), random forests (RF) and multivariate adaptive regression splines (MARS). The ELM is an advanced form of ANN, so it will be included under literature relating to ANNs. Examples of models developed for each are given below under their relevant grouping. As this is an emerging and rapidly changing field of study, enhanced versions of these approaches and, indeed, wholly new approaches, are being researched and reported constantly.

2.9 Selected ML algorithms

2.9.1 Artificial Neural Networks

Artificial neural networks (ANN) have been one of the most utilised ML models in biological and environmental systems research where they have been used to predict outputs such as yield and biomass (Shastry et al. 2016; Ghimire et al. 2018; Sanikhani et al. 2018; Nettleton et al. 2019). Originally designed to imitate the functioning of neurons in a brain, ANNs represent some of the earliest ML algorithms. They are invariably 'black box' models which are trained on input data and automatically self-calibrate to classify or predict output values, the internals of the ML model generally not being able to be observed by a user of the system.

2.9.2 Multivariate Adaptive Regression Splines

The Multivariate Adaptive Regression Splines (MARS) algorithm is considered to be a form of both regression analysis and ML, as the resulting model is generated, tested and refined based solely on the data set being used generate the model. This method was developed by Friedman (1991a), and further described in Friedman and Roosen (1995). The technique uses recursive partitioning of response functions, or splines, by introducing new basis functions to progressively improve a model of high dimensional data interactions to predict an output value. The MARS algorithm generates a mathematical equation that produces a continuous model with continuous derivatives. This equation explains the interactions between explanatory and target variables. Evaluation of this equation allows for the generation of surface response plots showing input parameter interactions as well as calculating the importance of input parameters in the generation of an output value. The MARS algorithm has previously been used to create environmental models. For example, it has been used in modelling nitrate flux under potatoes (Fortin et al. 2014) and modelling evaporative loss from farming systems' soil and water bodies (Deo et al. 2015).

2.9.3 Random Forest

There are many examples in agricultural research of Random Forest (RF) models being utilised. They have been used for yield forecasting (Kouadio et al. 2018; Feng et al. 2019; Feng et al. 2020; Obsie et al. 2020; Guo et al. 2021), soil models (Gebauer et al. 2019; Hussein et al. 2020) and analysis of remote sensing imagery (Belgiu & Drăguţ 2016; Dahms et al. 2016). They are one of the most widely used forms of ML frameworks, with Cravero and Sepúlveda (2021) finding that they are the second most referenced technique for analysis of big data in agriculture. Random forest algorithms use ensembles of decision trees to classify or predict outcomes. Decision trees enable great flexibility in the types of data that can be classified and analysed, and the inherent ability to evaluate alternate decisions at many nodes across multiple trees, hence the name 'random forest', provides a robustness to the algorithm to cope with imperfect or noisy data. The outcomes of the ensemble of decision trees are statistically evaluated using a method called 'bagging' to limit the influence of noise in the data and to avoid overfitting of the model (Biau & Scornet 2016). The RF algorithm has been included in this study because of its popularity with researchers for use in agricultural and environmental modelling.

2.10 Summary of the knowledge gaps and conclusions

Agricultural production systems simulation, of which crop models are an integral part, is extensively used for scientific agricultural research, food and fibre production planning and forecasting, climate change adaptation planning, policy assessment and planning, risk assessments from farm scale to international negotiations, to name but a few. Accurate, robust, and well tested models are critical to the trustworthiness of these simulations. Although SA of the models is recognised as best practice for the model's development, testing and application processes, it is not a routinely used aspect of the lifecycle of biophysical crop models. There appears to be no single answer as to why this is the situation. Undertaking a global SA (GSA) of a crop model is a complex task. Many modelling systems are not well designed for conducting the multiple iterative simulations required for GSA. In addition, to conduct such analyses, most GSA analysis requires programming and computational skills on the part of the model users, a skill set that many modellers do not have.

Most modern crop models which run in agricultural production systems simulation environments, such as APSIM-NextGen, are complex models that require large numbers of input parameters. This means that there are extremely large numbers of potential combinations of input parameters to be assessed when performing SA. The computational overheads of running and analysing the output of such large numbers of simulations required to thoroughly analyse a model's performance quickly becomes unworkable. One option for reducing the computational burden of running large numbers of simulations is the generation and use of emulator models which approximate the functioning of the original complex model, generally with much lower dimensionality, or a narrower functional range of input values. This approach has been used in other science disciplines, for example the area of hydrology and water resources, but a similar level of adoption has not been seen in crop modelling. Additionally, the rapidly expanding area of AI and ML has not been adequately assessed for its potential application to the area of SA of process driven models. There is a knowledge gap in this area as little, if any, ML has been applied to SA for process driven biophysical model development, or any process driven models. Machine learning algorithms that can efficiently create an emulator to minimise the computational burden of running the number of simulations required for SA offer a potential path forward to facilitate the use of SA as 'best practice' in model development. Artificial Intelligence and ML techniques are being used in other areas of agricultural research, but not currently in model testing and development.

2.11 Research questions

The overarching research question for this study is: Are there ML algorithms that can be used to perform sensitivity and uncertainty analysis effectively and efficiently on process-driven biophysical crop models?

This broad question can be broken down into task focused questions:

1. When using a ML emulator of a process-driven biophysical crop model, are there features of the emulator, such as input variable importance, which are comparable to the Morris or the Sobol indices generated by running SA on the process-driven biophysical model itself?

2. Which ML algorithm, e.g., MARS, ANN, or RF, produces the most accurate emulator, and at what computational cost? What are the advantages and disadvantages of each algorithm for producing an emulator?
3. When using a ML derived emulator, can SA be performed using the Morris and the Sobol methods, or some other analysis method, which gives comparable results to results obtained when run on the process-driven biophysical model itself? What considerations need to be considered when utilising these approaches?

From these research questions, the research objectives are:

1. Develop a method that allows the indices of Morris, Sobol and MLEs parameter importance to be compared to assess if MLEs report comparable SA indicators as the standard traditional methods.
2. Compare different ML algorithms to assess which are best suited to the role of generating MLEs for the process-driven models. These will be assessed on their accuracy of predicted values and the comparative computational burdens of generating the ML models.
3. Perform SA on the MLEs using the Morris and the Sobol methods to test the speed of performing the analysis and compare the consistency of results between the different MLEs.

CHAPTER 3: PAPER 1 - Evaluation of the effectiveness of using machine learning emulators for sensitivity analysis of process driven models: A case study using a model of chickpea phenology

Preamble

The purpose of this research chapter is to demonstrate if the input variable importance indices generated by machine learning models are correlated with the indices generated by the statistical methods of Morris and the Sobol. The Morris method is often used as an efficient screening tool to establish which input parameters are most important and which have no, or little effect, on output values. The Sobol method uses variance decomposition analysis to calculate which variables contribute most to the output's variance, and the interactions between different input variables on the output values. In addition to the input importance indices, machine learning models, such as the MARS approach, can generate matrices of data which record the individual and combined contribution to the output values, of two input variables at fine scale across the variables' value ranges, These data sets can then be plotted as surface response curves showing the interaction of the input variables to the value of the output target. What is not evident from literature is how these machine learning model indices compare with the indices generated by traditional statistical analysis methods.

3.1 Introduction

Sensitivity analysis is concerned with the mathematical and statistical analysis of how the variations in input parameter values contribute to the variations in output values of a simulation model. It is an analysis tool that is applicable to modelling in general across a wide range of science disciplines and can be used in research which utilises anything from simple mathematical models to complex multi-model systems. Agricultural systems models, such as APSIM-NextGen (Agricultural Production

Systems sIMulator – Next Generation (Holzworth et al. 2018)), are complex assemblies of process based biophysical models requiring large numbers of input parameters. Development and validation of individual sub-models, as well the validation and calibration of the whole of the modelling system consumes significant amounts of resources (Jones et al. 2016). By using SA to identify the most and least important model inputs determining a target output - for example, maximum and minimum temperatures, but not twilight, for emergence duration - the efficiency and accuracy of the model development and validation processes can be improved (Ascough II et al. 2004; Archontoulis et al. 2014; Norton 2015). There is a responsibility for both model developers and researchers to be as efficient as possible in producing their results, both in terms of money and time expenditure. Also, in practical terms, the amount and type of the computing resources required to perform the simulation tasks should be kept to a minimum. Work has previously been done on the use of emulators (also referred to as metamodeling in the literature) for applications seeking to reduce the dimensionality of complex mathematical models and for sensitivity analysis (Ratto et al. 2007; Ratto et al. 2012; Razavi et al. 2012; Villa-Vialaneix et al. 2012). Dynamic emulation modelling was formalised by Young and Ratto (2009); Castelletti et al. (2012) for the purpose of assisting in the SA of high order simulation models and potentially even replace the complex model with a reduced-order parametrically efficient emulator. The detailed procedural framework developed is undoubtedly robust and thorough, but the lack of its widespread adoption in disciplines outside the fields of expertise of the founding developers potentially highlights, at least a perceived, issue of complexity.

Despite the field of SA being a data focused discipline, data centric machine learning approaches have also not been adopted by practitioners of SA. Lack of familiarity with many of the advanced SA methods, their calculation requirements and the interpretation of their results has hindered the adoption of formalised and rigorous SA practices by the developers of process driven models, which includes biophysical crop models. Razavi et al. (2021) discuss six key areas, identified by a multidisciplinary group of SA researchers and practitioners, which highlight the benefits and challenges of advancing the integration of SA into modelling methodologies across a range of disciplines. The advancement of sophisticated agricultural systems modelling environments, such as APSIM-NextGen, provides an

opportunity for standardised SA methods to be made available to model developers and modelling practitioners via simple intuitive interfaces. Development of efficient and easily implemented software tools may assist in addressing the underlying problem of the discipline of simulation modelling not adopting best practice principles with respect to the consistent use of SA. It is the goal of this research to assess a data centric methodology that has the potential to address this shortcoming.

Pianosi et al. (2016) reviewed a range of SA approaches used for analysis of environmental models. These approaches, which are also applicable for the SA of biophysical crop models, ranged from the relatively simple and computationally efficient Elementary Effects Test (EET) methods which are most applicable for qualitative screening of input parameters, to the much more computationally demanding variance-based analysis approaches which are more suitable for the detailed analysis of a smaller number of input parameters. Each approach has its strengths and limitations. Of the EET approaches, the Morris method (Morris 1991) and its revised version (Campolongo et al. 2007) have gained a level of popularity as shown by its use in a range of simulation experiments (Casadebaig et al. 2016; Sarrazin et al. 2016; Pardon et al. 2017; Jaxa-Rozen & Kwakkel 2018). The required sample size is generally 10 to 100 times the number of input factors (Pianosi et al. 2016). The Morris method, with its focus on determining which input factors may be considered as having effects which are (a) negligible, (b) linear and additive, or (c) non-linear or involved in interactions with other factors (Campolongo et al. 2007), has been found to be robust and suitable as a SA screening tool for environmental models (Sarrazin et al. 2016). It computes two sensitivity measures: μ, which is the standardised effect on an output variable of a positive or negative change of an input variable, also known as the elementary effect (Morris 1991), and σ, which estimates a factor's level of non-linearity and/or interactions with other factors. For variance-based SA approaches, one of the established methods is the Sobol method (Sobol' 1993). The Sobol method is a global sensitivity analysis approach based on variance decomposition, where the total variance of an output parameter's value is decomposed into component variances from individual parameters and their interactions. It does, however, come at the expense of a high computational cost. While there are no strict guidelines as to the sample size required to obtain reliable sensitivity indices, Pianosi et al. (2016) suggests that the sample size should be

approximately 1000 times the number of input factors, or greater. It should be noted that, while the two SA approaches, the Morris method and Sobol method, are tried and tested approaches (Ascough II et al. 2004; Iooss & Lemaître 2015; Pianosi et al. 2016), SA is not limited to these approaches and investigating potential efficiency gains through alternative methods is a worthwhile endeavour.

The advancement of computing power and ongoing developments in Artificial Intelligence (AI) and Machine Learning (ML) research, especially on big datasets required to improve complex process driven models, presents a field of potential alternate approaches for data generation and analysis that might yield computational gains and bring in time savings for the developers and end-users of simulation modelling systems. The field of ML has contributed many different data centric tools to a wide variety of problem applications. Data driven methods, as represented by ML algorithms, enable computers to produce useful outputs by 'learning' patterns and relationships in input data without being hardcoded with predetermined rules to solve the problems directly. The iterative steps of process driven model development – develop/test/refine – involves data generation and data analysis at all stages. Villa-Vialaneix et al. (2012) compared eight techniques to produce metamodels of a biophysical application. The aim of their study was the assessment of the reduction in the computational expense of generating model outputs. The focus of our research, in contrast, is the investigation of the potential of improving the efficiency of undertaking SA directly, rather than the generation of outputs per se. This has the effect of shifting the focus from primarily assessing the accuracy of the emulator outputs, to being able to review how the sensitivity indices have been calculated. It has been noted the 'black-box' approach of many ML algorithms do not allow the necessary verification that the emulator is 'credible' (Castelletti et al. 2012). Certainly, for SA, the ability to review the calculation process is desirable. As the generation of ML models can be computationally less demanding than running of very large numbers of simulations using process driven models, gains in computational efficiencies of obtaining SA results might be able to be realised by developing the MLEs instead of running the SA directly using the process driven models.

The ML tool selected for this research needed the ability to have its outputs compared against the outputs of the Morris and Sobol methods in terms of SA. The Multivariate Adaptive Regression Splines (MARS) algorithm (Friedman & Roosen 1995), as implemented in Adaptive Regression Splines – MATLAB ARES toolbox (Jekabsons 2016), was selected for use in this analysis. Technically, the MARS method is a data driven regression technique that has some essential features of ML methodologies, that being the ability to configure itself to produce useful target outputs from given inputs without being specifically programmed to produce these outputs. It can, in this respect, be considered a 'data-intelligent ML' approach. As part of the MARS algorithm, input parameters are assessed for their impact on the value of the output variable. This is reported as the input variables index of importance. It is these indices that will be compared to the Morris and Sobol indices as a method of assessing the sensitivity of the input parameters. The MARS method has previously been used to create emulators for environmental models (Fortin et al. 2014; Deo et al. 2015). In the process of generating an emulator, this method produces an accurate analysis of the input variable importance and the input variable interactions. Its intuitive outputs, particularly its input importance indices, can readily be utilised for the purposes of SA. Additional functionality allows the emulator to generate surface plots of the input parameter interactions.

This chapter assesses a new approach to an old problem by investigating how ML methodologies might be implemented for the SA of complex process driven models. This is not to say that the traditional statistical techniques used for SA are not adequate for the task at hand, but the issue of computational burden is a problem that warrants research to identify potential gains in efficiency. Using a biophysical model as an example, limitations that might be encountered and potential benefits resulting from the use of ML algorithms are identified. The focus of this research is not the parameter SA per se, but rather the comparison of the results between the different approaches and their relative efficiency of execution. The objective of this chapter is to establish, by demonstration, whether the parameter importance indices of an MLE can provide the same, or comparable information, as the indices produced by the traditional SA methods of Morris and Sobol, and if they do, is the computational burden of producing these indices reduced compared to the established methods.

This will be done in context of a chickpea model in APSIM-NextGen. To date, there has been little published research into ML approaches that might be used as alternatives to, or in addition to, traditional statistical approaches for SA analysis of biophysical models. Alternative methodologies for sensitivity analysis will be required if the developers of simulation models are to use such techniques as data centric ML when developing, testing, and evaluating their models. If MLEs are shown to provide measures which are comparable to those provided by the current statistical approaches of Morris and Sobol, then the data centric ML techniques can potentially provide such alternative methodologies for the SA of process driven models. The research presented in this chapter will therefore explore the potential role that data centric ML techniques can have in the sensitivity analysis of the biophysical crop model APSIM compared to the existing statistical approaches of Morris and Sobol.

3.2 Methodology

Three approaches to SA are compared in terms of computational efficiency, as measured by the number of model evaluations (simulations) that are required to produce stable predictions of the influence indices, the order of the predicted influence indices, and a comparison of the level of confidence that can be placed in the influence index values is provided.

3.2.1 Computing environment

All simulations and data analyses were performed on an Intel Core-*i*7 7600U CPU 2.9 GHz based computer with 16 GB RAM running Microsoft Windows 10 operating system. The APSIM version used was APSIM-NextGen (vers 2020.02.05.4679) (Holzworth et al. 2018). The APSIM-NextGen chickpea model, a redevelopment of the APSIM-Classic chickpea model and based on research by Robertson et al. (2002), was used as the crop model. Built-in features of the APSIM-NextGen User Interface were used to configure and run both the Morris and Sobol analyses. The built-in feature of the APSIM-NextGen User Interface for configuring and executing factorial simulation experiments was used to generate the data sets of

input and output values. These data sets, stored and transferred in comma separated values (CSV) file format, were then used for the MARS analysis. The MARS analysis was performed using the ARES package (Jekabsons 2016) for MATLAB® (ver. R2017b). This analysis was undertaken on the same Windows based computer that was used to run the APSIM simulations.

3.2.2 Simulation configuration

The phenology sub-model was the target of the SA testing for this simulation experiment, with seven inputs of phenological importance selected for analysis against each of seven model outputs. The input parameters are listed in Table 3-1, along with the input value ranges used. Values were based on default values used in the chickpea model, with value ranges being established to ensure crops grew to maturity in each of the simulated locations and for each of the sowing dates used. These input parameters and value ranges were used for both the Morris and Sobol methods. Factorial experiments were defined and run to generate the data used to create the MARS emulators. The same set of input parameters were used with discrete values defined for each parameter, also detailed in Table 3-1. The seven output variables from the crop model that were chosen for sensitivity analysis were: (1) Days from sowing to emergence (EmergenceDAS), (2) Days from sowing to flowering (FloweringDAS), (3) Days from sowing to crop maturity (MaturityDAS), (4) Crop biomass at harvest (kg/ha) (Biomass), (5) Weight of harvested grain (kg/ha) (GrainWt), (6) Weight of reproductive organs (kg/ha) (PodWt), and (7) Count of potential reproductive plant nodes (/m2) (NodeCnt).

Table 3-1. Crop model input parameters used for factorial experiments and sensitivity analysis.
Parameters for the ASPIM-NextGen chickpea's phenology sub-model are listed with descriptions, input values and input value ranges.

Input Variable Name	**Description**	**Values Used as factors for experiments**	**Range (min and max values) used for sensitivity analysis**
BaseTemp	Temperature below which phenological development ceases	0, 2, 4, 6 ^{o}C	0.0 – 6.0 ^{o}C
OptTemp	Optimum temperature for phenological development	24, 26, 28, 30 ^{o}C	24.0 – 30.0 ^{o}C
MaxTemp	Temperature above which phenological development ceases.	34, 36, 38, 40 ^{o}C	35.0 – 40.0 ^{o}C
OptTempDD	Thermal time accumulated at optimum temperature	25, 26, 27, 28 ^{o}Cd	25.0 – 28.0 ^{o}Cd
CriticalPhotoperiod	Photoperiod below which phenological development ceases	6, 7, 8 hours	6.0 – 9.0 hours
OptPhotoperiod	Optimum Photoperiod for phenology development	16, 17, 18, 19 hours	16.0 – 22.0 hours
Twilight	Sun's angle below horizon which still produces photoperiod response	-2^{o}, -4^{o} , -6^{o}	-2.0^{o} to -6.0^{o}

3.2.3 Morris method

The Morris method (Morris 1991) is an EET that can be used for the analysis of input parameters' influence on the value of a model's output. Campolongo et al. (2007) further enhanced the method to improve its robustness in situations where effects can return both positive and negative values. The values produced by the Morris method are the estimate of the mean of the distribution of the absolute values of the elementary effects (EE) defined as μ^*, and the standard deviation of the distribution of values (σ). The standardised effect of a positive or negative Δ change of an input variable is calculated using Eq. (3.1).

$$EE_i(X) = [y(x_1, x_2, \dots, x_{i-1}, x_i + \Delta, x_{i+1}, \dots, x_k) - y(x)]/\Delta \quad (3.1)$$

where Δ is magnitude of step, which is a multiple of 1/(p - 1); p is the number of 'levels', or values, over which the variables can be sampled.

For each input variable, the mean (μ) and standard deviation (σ) of the set of EEs are calculated using Eq. (3.2) and Eq. (3.4).:

$$\mu_i = \frac{\sum_{n=1}^{r} EE_n}{r} \quad (3.2)$$

$$\mu^*{}_i = \frac{\sum_{n=1}^{r} |EE_n|}{r} \quad (3.3)$$

$$\sigma_i = \sqrt{\frac{1}{r} \sum_{n=1}^{r} (EE_n - \mu_i)^2} \quad (3.4)$$

The strength of the relationship between the *i*-th input variable and the output response due to all first- and higher-order effects that are associated with that variable is assessed by the sensitivity index μ_i (Campolongo & Braddock 1999).

Campolongo et al. (2007) develop the use of μ^*, the mean of the distribution of absolute values of the EE_i, as given in Eq. (3.3). When μ^*_i is high in comparison to other variables, this input variable has a stronger influence on the output value. Conversely, a variable with a low μ^*_i value has lesser influence associated with it as the same Δ change causes a relatively small change in output (King & Perera 2013). The variance (spread) of the finite distribution of the EE_i values, denoted by σ_i, is calculated by Eq. (3.4). The greater the value of σ_i, the greater the indication of possible interactions with other variables and/or that the variable has a non-linear effect on the output (Campolongo & Braddock 1999).

To simplify comparisons against the other SA methods, the $\mu*$ values are converted from the default output value's units to a normalised range of values (0 – 1) (Eq. 3.5). The values of σ were used to calculate the 95% confidence intervals for each of the $\mu*$ values.

$$\mu*_{Normalised} = \frac{\mu*}{\left(\mu*_{max} + CI_{\mu*max}\right)} \tag{3.5}$$

where $\mu*$ is the value produced by the Morris method, $\mu*_{max}$ is the maximum value of $\mu*$ for the set of input parameters being analysed and $CI_{\mu*max}$ is the 95% confidence interval of the maximum $\mu*$ value.

The APSIM-NextGen user interface provides a feature for Morris analysis. This feature utilises R (R Core Team 2020) libraries to generate the Monte Carlo randomised data sets and to perform the statistical analysis of the EET. For this analysis, the R package utilised was *sensitivity: Global Sensitivity Analysis of Model Outputs* (Iooss et al. 2020) and the function used was *morris: Morris's Elementary Effects Screening Method*. As the number of simulations executed increase, the value of $\mu*$ becomes more stable and converges towards the true mean value of the population of simulations. The value of $\mu*$ was plotted against the number of simulations executed to show the rate of convergence for selected output values. The computational requirements of the Morris method are N = n * (p + 1), where N is the total number of simulations required, n is the number of sets of simulations that the user requests to establish the value of $\mu*$, and p is the number of input parameters being evaluated. For comparison against the other SA methods, these sets of

simulations are simply reported as N, the total count of simulations that are required to be executed. In this analysis 520 simulations were performed to calculate the parameter importance by the Morris method. Based on the findings of Campolongo et al. (2007), who demonstrated empirically that the sensitivity measure calculated by the Morris method can be used as a comparison measure against the total effects index produced by the Sobol method, the index values base on $\mu*$ were chosen for comparison against the Sobol total order index values. This approach was also used by Sarrazin et al. (2016) for their comparative study which included the Morris and Sobol methods.

3.2.4 Sobol method

The Sobol method (Sobol' 1993) is a variance decomposition method of SA. This approach of SA breaks down variation in an output value, calculated by function V(Y) Eq. (3.6), to attribute the effects to variations in individual input parameters and interactions between input parameters.

$$V(Y) = \sum_i v_i + \sum_{i<j} v_{ij} + \sum_{i<j<k} v_{ijk} + \cdots + \sum_{1\ldots p} v_{1\ldots p} \quad (3.6)$$

where v_i is the amount of variance due to the i-th parameter X_i, and v_{ij} is the amount of variance due to the interaction between parameter X_i and X_j. The sensitivity of single parameter or parameter interaction, i.e., Sobol''s sensitivity indices of different orders, is then calculated based on their proportional contribution to the total variance V by using Eq. (3.7) for first-order indices, Eq. (3.8) for second-order indices, and Eq. (3.9) for the total-effects indices:

First-order index $$S_i = \frac{v_i}{V} \quad (3.7)$$

Second-order index $$S_{ij} = \frac{v_{ij}}{V} \quad (3.8)$$

Total-effects index $$S_{Ti} = S_i + \sum_{j \neq i} S_{ij} + \cdots \quad (3.9)$$

where S_i measures the sensitivity from the main effect of X_i, S_{ij} measures the sensitivity from the interactions between X_i and X_j. The total-effects index, S_{Ti}, measures the main effect of X_i, plus the effects of all its interactions with parameters other than X_i (second-order index values). Note: S_i and S_{ij} are limited to the value range ($0 \leq S_i \leq 1$), while total-effects indices can sum to a value greater than 1.

The default analysis generally produces two sensitivity indices, the main or primary effects index (S_1) and the total effects index (S_T) for each input parameter. The S_1 index is a measure of the input parameter's influence by itself on the value of the output. The S_T index is a measure of the effect that the input parameter has, both by itself and in combination with all other input parameters, on the value of the output. The value of S_T is most comparable to the measure associated with $\mu*$ of the Morris method.

The version of APSIM-NextGen used for this research has an interface for conducting Sobol analyses. This feature calls an R subsystem which used the *SobolSalt* function in the *sensitivity: Global Sensitivity Analysis of Model Outputs* (https://CRAN.R-project.org/package=sensitivity) (Iooss et al. 2020) package library to generate the randomised sets of simulations and to analyse the simulation outputs. The APSIM modelling system generated and ran the crop model simulations. For comparison against the other SA approaches, the S_T values for each input parameter were plotted, along with their 95% confidence intervals. For brevity, three of the seven outputs, ones which most clearly showed the range of responses and allowed clear comparisons between methods, were chosen for presentation in the results. The calculation of the Sobol statistics involves the evaluation of multiple samples from population of values generated by the Monte Carlo sampling plan. These samples generate slightly different values for the Sobol statistics S_1 and S_T. As the sample size increases, the variations in the predicted values of these statistics decreases. The Sobol method calculates the Root Mean Squared Error (RMSE) of each of these statistics providing an indicator of the stability of the statistic. As a measure of convergence towards stable predicted values, the RMSE of the S_T values was plotted against the number of simulations executed. The computational requirements of the Sobol method are $N = n * (p + 2)$, where N is the total number of simulations

required, n is the number of sets of simulations that the user requests to establish the value of S_T, and p is the number of input parameters being evaluated. For comparison against the other SA methods, these sets (n) of simulations are simply reported as N, the total count of simulations that are required to be run.

3.2.5 MARS method

The Multivariate Adaptive Regression Splines (MARS) approach to SA is considered as a form of advanced data analysis tool with feature extraction skill utilising ML algorithms. This is an explanatory method developed by Friedman (1991a), further described in Friedman and Roosen (1995). The technique uses recursive partitioning of response functions by introducing new basis functions to progressively improve a model of high dimensional data interactions to predict an output value. Unlike many ML models based on Artificial Neural Networks (ANN) which are described as 'black boxes' for the lack of transparency as to how they achieve an output, MARS generates a mathematical equation that produces a continuous model with continuous derivatives. This equation, which aims to explain the interactions between explanatory and target variables, is made up of basis functions which form the splines, and parameters that describe knot locations and hinge details and can be reported during model development. Evaluation of this equation allows for the generation of surface response plots showing input parameter interactions as well as calculating the importance of input parameters in the generation of an output value, or SA as it is referred to in this text. The method also meets the criteria for a ML data analysis tool as the resulting model is automatically determined by the data used to generate the model and does not require additional programming to address the specific problem that the data relates to. The data determines the model. Consideration of the performance of other ML algorithms which represent the 'black-box' approaches will be considered in subsequent chapters of this research. For this study, the Jekabsons implementation of the MARS method, referred to as Adaptive Regression Splines (ARES) (Jekabsons 2016), was used under MATLAB®. Reports produced by APSIM-NextGen listing the chickpea model's input parameters and their values, as well as the seven outputs and their values for each simulation run were used to generate datasets. These datasets were

then read into MATLAB, pre-processed to remove data of simulations that did not produce a crop, and randomised to ensure a consistently mixed dataset. The datasets were split into 80/20 proportions as training and testing subsets, then passed into the ARES routines. Prior testing using similar crop model simulation datasets had shown that the ARES model building routine, *aresbuild*, performed best, based on a balance between processing time and final model accuracy, with parameters for the maximum number of basis functions, *maxFuncs*, set to 101 and the maximum level of input parameter interactions to analyse, *maxInteractions*, set to 4. All remaining parameters were set, by omission, to use default values. The default settings use a piecewise-cubic regression function for the model generation (Jekabsons 2016).

The ARES build process generates a model, also known as an emulator, which takes as inputs the values of the input parameters selected for reporting by the APSIM-NextGen crop model and predicts the output value used as the target for the build of the emulator. To assess the predictive ability of the emulator, test data not used in the building of the emulator is run through the emulator and the values of the target output parameter evaluated against the true values generated by the APSIM modelling system. The accuracy of the emulator improves with an increase in the size of the training dataset. As the size of the training dataset increases, the variability of the predicted values decrease, as measured by the RMSE of the predicted values. To facilitate a comparison with the Sobol method, the RMSE value of the predicted values of the test dataset has been used to evaluate the convergence towards a stable emulator that has low predictive error. The convergence towards a stable prediction for the ARES model is an analysis of the stability of an output value produced by the model. This means that there is only one value series to plot per model output, rather than the series of seven input values plotted for Morris and Sobol methods.

The *aresimp* function (Jekabsons 2016) was run by the MARS emulator models to report the importance indices for each input parameter. The most influential input was assigned an index value of 100, with each other input given an index value proportional to this input's significance. Input parameters that did not contribute to the calculation of the output were assigned an index value of zero. For the purpose of comparison against the other SA methods, the MARS input importance indices were

converted to a zero to one value range and presented graphically. The derivation of these indices of importance is based on the complex mathematical equation which describes the relationship between each input and the output value.

Sensitivity indices of input parameter importance produced by the ARES emulator are mathematically derived from the complex regression equation which describes the emulator. The variability of predictions is associated with the output value, not with the prediction of the sensitivity indices as is the case with the other SA methods. The predictive ability of the emulator has been assessed using two statistical values, the coefficient of determination (R^2) and the coefficient of efficiency (COE_{LM}, also known as Legates-McCabe index). The mathematical formulae are shown in Eq. (3.10) for R^2 and Eq. (3.11) for COE_{LM} (Legates & McCabe Jr 1999).

$$R^2 = \left(\frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}} \right)^2 \tag{3.10}$$

$$LM = 1 - \left[\frac{\sum_{i=1}^{n} | y_i - x_i |}{\sum_{i=1}^{n} | x_i - \overline{x} |} \right] \tag{3.11}$$

Where *n* is the number of data points, *x* values are the values generated by the APSIM modelling system ('observed') and *y* values are the predicted values generated by the ARES emulator.

The R^2 statistic is a widely used goodness-of-fit measure for testing how well a model fits the observed data but it does come with limitations that can produce biased assessments of a model's ability to simulate observed data (Willmott 1981; Willmott et al. 1985; Legates & Davis 1997). The COE_{LM} index, developed by Legates and McCabe Jr (1999) is a more robust measure of the goodness-of-fit between observed and predicted values as it addresses the short-comings of the R^2 statistic. In evaluating the accuracy of MARS model to emulate the results of the

APSIM model, both the R^2 and the COE_{LM} indices were assessed, with the values closer to unity for both statistics assessed as being more favourable.

The indices of input importance for the ARES model are calculated as part of the model generation and training process by calling function *aresimp*. Jekabsons (2016) notes that the variable importance estimates are calculated according to the work of Friedman (1991a) Section 4.4:

> "*The relative importance of a variable is defined as the square root of the generalised cross validation (GCV) of the model with all basis functions involving that variable removed, minus square root of the GCV score of the corresponding full model, scaled so that the relative importance of the most important variable (using this definition) has a value of 100.*"

They are calculated on a 0 to 100 scale, with the input having the most significant influence on the output value assigned a value of 100. Other inputs are assigned values proportionate to the most significant input based on their influence on the output value. Inputs that do not contribute to the output value are assigned an index of importance value of zero.

3.2.6 Comparisons methodology

MARS emulators were developed for each of the output targets. Morris and Sobol SA were then performed on the inputs for each of the output targets and their results and computational effort compared to the sensitivity results, as measured by variable importance indices, and the computational effort of the MARS emulators. The SA methods were compared on three features: the rate of convergence towards stable input index of importance values; the consistency of these index values with expected results and with the values produced by the other SA methods; and the amount of information or insight about the input parameter's influence on the crop model's performance. The measure of the rate of convergence towards stable index values reflects the computational efficiency of the method. For the Morris method, while there might be statistical approaches to yield accurate convergence measures, for example, those used by Sarrazin et al. (2016), the purpose of this research is to

compare the usefulness of these SA approaches for use by developers and users of biophysical models. From this perspective, the ubiquitous approach of 'assessment by visual inspection' of the convergence plots was the most appropriate. For the Sobol method, the plotting of calculated RMSE values for the influence indices allows a simple comparison of the convergence of the underlying index values against the convergence of the MARS method. For the MARS method, the additional statistics of the R^2 and COE_{LM} indices provided additional indicators of the stability of the emulator and therefore of the input importance indices. Using the underlying concepts of analysing the convergence of predicted outcomes towards a stable result and using an index for the assessment of the significance of an inputs contribution to an output's value, a comparison can be made between these diverse methods for their usefulness in the SA of biophysical crop models.

To enable a graphical comparison of the importance indices generated by each of the three methods, the Morris significance indices based on the value of $\mu*$, and the total effects indices, S_T, of the Sobol method, were normalised to the 0 to 100 scale and graphical plots generated. A similar approach was used by Sarrazin et al. (2016), who normalised Morris indices to a 0 to 1 range for comparison against Sobol Total Effects Indices. For assessment of the validity of the values of the importance indices, the Morris and the Sobol methods offer little in the way of crosschecking their accuracy. With the MARS method the user has the statistical analysis of the accuracy of the performance of the emulator, in the form of the R^2 and COE_{LM} indices, to assess whether the results are trustworthy. The R2 statistic, while not the most robust measure for measuring goodness-of-fit between two data sets, has been included in the analysis as it is the most widely used and recognised statistic of this type and so forms a useful baseline for the comparison with other goodness-of-fit statistics. The Legates-McCabe index is less well known but has been shown to be much more discerning in its analysis of goodness-of-fit. For these reasons, both statistics have been used and are presented in the results of this experiment. Assessment of the level of insight that the SA method provides into the input parameter's influence on the crop model's performance was subjective and done on the basis that more information is almost always desirable, especially where there is no additional computational load required.

3.2.7 Expanded data sets

The detailed analysis of comparing the index values of individual input parameters against each other across each of the SA methods is a necessary and suitable approach to establish confidence in how the different methods compare against each other. This approach was not, however, suitable for comparing the SA methods across larger data sets. For this purpose, the sensitivity index values of each method were chosen, as described in previous sections of this research, to normalise these values using a zero to one hundred range, and then use simple xy-scatter plots to visually observe the agreement between the methods. The input parameters having the most significant effect on the output grouped at or near the 100 mark on the sensitivity scale, while values assessed has having no impact, or very little input, grouped at the low end of the scale close to zero. Even where the rankings of factors were consistent between methods, the actual index values vary between methods, so greater variation around mid-range index values was anticipated to be seen in the scatter plots. This approach for comparing methods is not presented as a statistically robust method. It was simply a method to allow rapid visual inspection of large data sets as to whether the different SA methods are roughly in agreement.

To expand the crop simulation data set, simulations were developed for three different sowing times at each of three geographically diverse cropping locations (Table 3-2). These configurations were selected to reveal expected changes in the chickpea phenology sub-model's sensitivity to different input factors. The growth of crops is driven, or limited, by different environmental factors under different growing conditions. The SA was used to reveal which factors were driving the crop growth under which scenario.

Table 3-2. Locations and sowing dates for crop simulations covering three diverse production locations and three sowing dates for each location.

Name	Latitude/Longitude	Sowing Dates
Gorgan, Iran	36°51′N / 54°16′E	15 Nov, 15 Jan, 15 Mar

	J	F	M	A	M	J	J	A	S	O	N	D
Mean MaxTemp (°C)	12.6	14.2	16.7	21.1	26.4	30.8	32.7	34.1	30.8	26.1	18.9	13.5
Mean MinTemp (°C)	3.0	3.9	6.6	10.9	15.4	20.3	23.4	23.8	20.6	14.7	9.0	4.4
Mean Rain (mm)	42.9	37.1	60.4	85.2	47.4	20.6	23.6	13.3	21.8	30.1	66.6	45.2
Mean SolarRadn (MJ/m^2/d)	9.3	12.7	15.6	18.3	21.9	24.7	23.9	23.0	19.1	14.7	10.1	8.0

Kununurra, Australia	15°38′S / 128°44′E	9 May, 1 Jun, 15 Jun

	J	F	M	A	M	J	J	A	S	O	N	D
Mean MaxTemp (°C)	36.3	35.8	35.9	35.7	32.9	30.7	30.5	33.0	36.2	38.4	38.8	38.0
Mean MinTemp (°C)	25.3	25.0	24.4	22.2	19.2	16.4	15.1	17.0	20.6	24.1	25.5	25.9
Mean Rain (mm)	209.2	216.0	142.7	30.1	7.0	3.0	1.4	0.1	5.2	23.5	59.6	142.6
Mean SolarRadn (MJ/m^2/d)	19.0	18.4	19.8	20.6	19.6	19.1	20.3	22.6	24.4	25.0	24.0	21.1

Warwick, Australia	28°12′S / 152°06′E	1 May, 1 Jun, 15 Jul

	J	F	M	A	M	J	J	A	S	O	N	D
Mean MaxTemp (°C)	29.7	28.6	27.0	24.2	20.6	17.6	16.8	18.5	22.5	25.2	26.9	29.0
Mean MinTemp (°C)	17.0	16.9	14.8	12.0	9.1	5.5	3.4	4.7	7.4	10.1	13.6	15.4
Mean Rain (mm)	59.6	62.5	38.2	42.4	51.0	20.4	33.3	18.4	17.2	45.0	45.3	72.4
Mean SolarRadn (MJ/m2/d)	22.9	20.7	19.2	15.9	12.3	11.7	12.5	16.4	20.1	21.7	22.9	23.6

3.3 Results

The primary focus of the type of SA that was considered in this chapter is the identification of the input parameters which have the greatest, or least, influence on the value of a given model output and the efficiency, in terms of the number of simulations run, required for this identification. All three approaches provide values, in a variety of forms, which achieve this objective. Each approach also provided a basis to order the importance of input parameters, with varying degrees of accuracy.

3.3.1 Efficiency of convergence

3.3.1.1 Morris Method Convergence

The panels of Figure 3-1a show how the influence index, the predicted value of $\mu*$ converge towards stable values for the Morris method of SA for three model outputs. By visual inspection, the values of $\mu*$ stabilised by around 300 simulation evaluations for all of the model outputs. The days to emergence (Figure 3-1a Subpanel i) had three principal influencers, base temperature (BaseTemp), optimum temperature (OptTemp) and peak thermal effect at optimum temperature (OptTempDD). The other four inputs; maximum temperature (MaxTemp), critical photoperiod (CriticalPhotoperiod), optimum photoperiod (OptPhotoperiod) and twilight angle (Twilight); only had minor influence on the output value. In this particular case, the values of $\mu*$ for the input OptTempDD indicated that 350 simulations would yield a more stable prediction than 300, but this did vary between outputs. The order of the inputs' influences was established with fewer simulation runs, between 150 and 200 simulations, except where the values of the influences are close in value, in which case the orderings continue to be unstable up to 500 simulations, the maximum number of simulations run for this analysis. This was most clear in the results for days to flowering (Figure 3-1a Subpanel ii) where four input parameters, BaseTemp, OptTemp, Twilight and OptPhotoperiod, are closely grouped in the mid-range of influence index values and their orderings continue to fluctuate beyond 400 simulations.

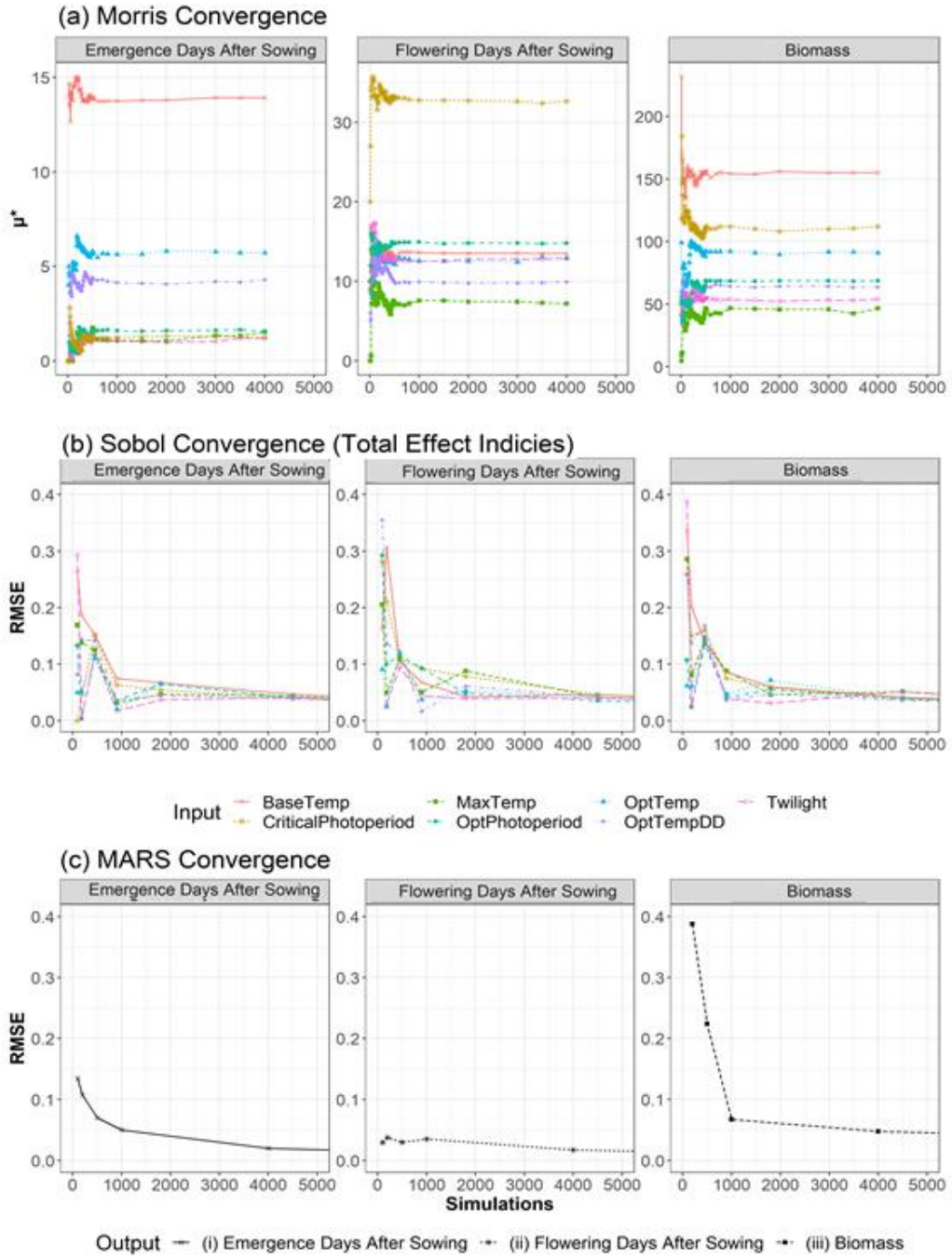


Figure 3-1. Convergence of predicted values.
(a) Morris method's μ∗ values for seven input parameters across three model outputs, against a count of model simulations. μ∗ is the elementary effect of the input on the output value in the units of the output parameter. (b) Convergence of Sobol Total Effects Indices, as measured by the Root Mean Squared Error (RMSE) of the estimated index values, against a count of model simulations. The values for seven input parameters are shown for each of three model outputs. (c) Convergence of the multivariate adaptive regression splines (MARS) models outputs towards prediction values with low variance for three model outputs. Outputs: (i) Emergence Days After Sowing, is the days after sowing to emergence; (ii) Flowering Days After Sowing, is days after sowing to flowering; (iii) Biomass, is above ground dry weight (kg/ha).

3.3.1.2 Sobol Analysis Convergence

In accordance with present results, around 9,000 simulations are required to see convergence and stability of the predicted influence indices for the Sobol analysis (Figure 3-1b). However, a greater number of simulations continue to reduce the RMSE values of the predicted values, but at a diminishing rate. The Sobol analysis reported, by default, the bias and standard error of the predicted index values. Up to and including 4500 simulations, there was a spread in the range of margin of error for the different input parameters. By 9000 simulations the margins of error become very similar, with further simulations only slightly reducing all margins of error for all parameter estimates.

3.3.1.3 MARS Simulation Convergence

The convergence towards stable values for three outputs for the MARS emulator are shown in Figure 3-1c. The statistic used to measure the convergence is the RMSE of the predicted value of the output and was derived from the MSE value generated and reported by the ARES routines during the development and testing of the MARS model. Visual assessment of the convergence plots for different model outputs revealed that the stability of the predictions for a given number of simulations varied depending on the output. For days to emergence and days to flowering (Figure 3-1ci,ii), RMSE was reduced to 0.02 or below by 4000 simulations, while RMSE for Biomass (Figure 3-1ciii) was 0.05 at 4000 simulations but remained above 0.02 up to 14,000 simulations.

3.3.1.4 Comparison of Convergence Results

Comparing the convergence graphs of the Morris method (Figure 3-1a) and the Sobol method (Figure 3-1b), the computational efficiency of the Morris method was evident as the stability of $\mu*$ was established for each input parameter for each output target by 300 to 350 simulations (RMSE of $\mu*$ values < 0.05). Sobol required around 9000 simulations to establish consistently low RMSE values (RMSE of ST values < 0.05) for the corresponding inputs across the same output targets. The MARS method's convergence (Figure 3-1c) showed a different pattern. The RMSE values of the predictions of days from sowing to emergence (Figure 3-1ci) and days from sowing to flowering (Figure 3-1cii) settled at values below RMSE of 0.025 for

simulation counts of 4800, with very little gained by running more simulations. The RMSE for the prediction of biomass (Figure 3-1ciii) was reduced to approximately 0.05 by the running of 4800 simulations and was reduced further to about 0.03 by the running of 14,000 simulations. A low variance threshold of an RMSE of less than or equal to 0.05 was established for the acceptance of the sample size being adequately large to produce stable predictions of the measured statistic. A summary is shown in Table 3-3.

Table 3-3. Summary of three SA methods for the degree of computational effort. Indicated by the number of simulations required, to produce measures of sensitivity with low variance (RMSE < 0.05).

Output	**Morris**	**Sobol**	**MARS**
EmergenceDAS	300	7000	3000
FloweringDAS	300	7000	1000
MaturityDAS	350	9000	4000
Biomass	400	9000	5000
GrainWt	400	9000	5000
PodWt	400	9000	5000
NodeCnt	350	9000	5000

3.3.2 Measures of Parameter Importance

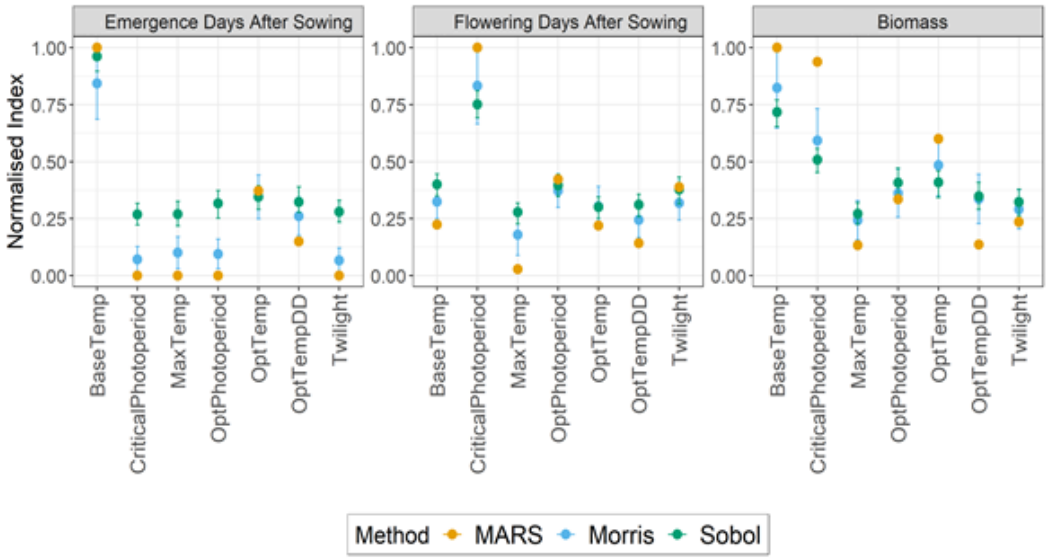


Figure 3-2. Parameter importance indices for seven input parameters for each of the three SA methods.
Indices of 0 are least important and values of 1 are most importantThree model outputs are shown; EmergenceDAS is the days after sowing to emergence; FloweringDAS is days after sowing to flowering; Biomass is above ground dry weight (kg/ha). Input importance indices for the multivariate adaptive regression splines (MARS) models were calculated from sample sets containing 9216 data points. The indices are defined as the square root of the generalised cross validation (GCV) of the model with all basis functions involving that variable removed, minus square root of the GCV score of the corresponding full model. Morris method values are input parameter influence indices (μ∗) with 95% confidence interval error bars. Sobol values are Total Effects Indices calculated from 9000 model simulation runs, reported as a measure of input importance with 95% confidence intervals.

3.3.2.1 Morris Parameter Importance

Parameter importance calculated by the Morris method are reported as influence index (μ∗) values with 95% confidence interval error bars derived from the method's σ values for each input parameter. Normalised (0 – 1) values of μ∗ were calculated for graphing. These results were derived from 520 simulation evaluations of the crop model. This represents 65 data sets as the computational requirements for the Morris method is: Simulation count = n * (p + 1), where n is the number of data paths (data sets) through the input parameter space, and p is the number of input parameters being evaluated. For these results, this equated to 520 = 65 * (7 + 1) simulations. The value of 65 data sets, a user definable input to the Morris calculation, was established after reviewing multiple runs involving from 20 to 100 data sets. A value between 60 and 70 data sets was shown to be optimal. The Morris values in Figure 3-2 show

how the most influential input parameters vary between model outputs. For days to emergence (Figure 3-2 Emergence Days after Sowing), the most influential input parameters are BaseTemp, OptTemp and OptTempDD, while the other four inputs, MaxTemp, CriticalPhotoperiod, OptPhotoperiod, and Twilight, had minimal effect on the output value. From the other panels (Figure 3-2 Flowering Days after Sowing, Biomass), critical photoperiod (CriticalPhotoperiod) was the standout influence on days to flowering, while CriticalPhotoperiod and BaseTemp appear to be driving the phenology's contribution to biomass production for this data set.

The confidence intervals reveal that, with the exception of BaseTemp for emergence and CriticalPhotoperiod for flowering, the rank order of the input parameter influences was not absolutely certain. Where influence values are close, the confidence intervals overlap significantly. This was consistent with the observation in the analysis of the convergence of the Morris method, that the order of input parameter influences remains uncertain, even when the number of simulation exections was increased. The identification by the Morris method of the most important influencers and the insignificant influencers was still correct.

3.3.2.2 Sobol parameter importance

Sobol Total Effects Indices were calculated using 9,000 simulations. The values, with 95% confidence intervals, are displayed in Figure 3-2 for the seven input parameters for each of three model outputs. The BaseTemp was the most influential input parameter for determining the value of the model output days to emergence (Figure 3- 2). The influence indices for the output days to flowering (Figure 3-2)) show the input CriticalPhotoperiod was the most influential input parameter. The input parameters BaseTemp and OptPhotoperiod were identified as the next ranked indices, with Twilight having a value that was close to the values of these two, but with a wider margin of error, so its ranking was more uncertain. For the output Biomass, the input parameter BaseTemp has a clear ranking of one, with CriticalPhotoperiod being ranked number two, while MaxTemp had the lowest ranked influence factor. The influence indices of the other four inputs parameters had values which were too close together to assert an order of ranking.

3.3.2.3 MARS method parameter importance

The importance indices for each input parameter were reported by the MARS emulators (Figure 3-2). Results for the days to emergence (Figure 3-2) showed BaseTemp as the most significant input factor with an index of importance of 100, OptTemp with an index of importance of 37 and OptTempDD with an index of importance of 15. The other four input parameters, MaxTemp, CriticalPhotoperiod, OptPhotoperiod and Twilight, were identified as unused by the MARS model and given importance indices of zero. It is worth noting that for the MARS analysis, the values reported reflect the analysis of the emulator, not an analysis of the functioning of the APSIM crop model itself. This is a limitation of using an emulator as an approximation of a full model. Results for the days to flowering show that CriticalPhotoperiod was the most important input parameter, MaxTemp as unused and the remainder of the inputs falling between index of importance values of 13 and 45. The input MaxTemp was classified as having an input importance of just 2.8 on the 0 to 100 scale by MARS. Input importance for Biomass (Figure 3-2)) showed CriticalPhotoperiod and BaseTemp as the most important input factors, with all inputs making a contribution to the calculation of the biomass output.

3.3.2.4 Comparison of parameter importance results

When comparing the parameter importance indices, the Morris method had notably larger confidence intervals (error bars) than the Sobol method for each of its input parameters. The shorter confidence intervals provided greater distinction between values when assessing the index value's order and greater accuracy when assessing the index's probable true value. The Morris estimates, however, required only 6% of the number of simulations (520 simulations instead of 9216 simulations) that the Sobol method required. The MARS importance indices were not reported with confidence intervals. Parameter importance rankings, based on the median index values, were consistent between all three methods with the exception of one pair of parameters values in each of the MARS method's flowering and biomass rankings. For the Morris and the Sobol methods, OptTempDD and Twilight inputs ranked fifth and sixth, respectively. For the MARS method this order was reversed to Twilight being ranked fifth and OptTempDD being ranked sixth. If the confidence intervals of the Morris and Sobol methods were taken into consideration, then it was not possible

to assert that this discrepancy in order had any significance as either order fell within the margins of error. Additionally, these are both showing low levels of sensitivity compared to more important parameters. The ranking of the more important parameters is consistent between methods.

In comparing the values of the importance indices relative to other indices calculated by the same method, more discrepancies between the methods became apparent. For the days to emergence, the Morris method (Figure 3-2) had the same pattern for the relative values of the indices to that of the MARS method (Figure 3-2), with the exception that the MARS values for CriticalPhotoperiod, MaxTemp, OptPhotoperiod and Twilight were zero where Morris values were close to zero. The Sobol method had only BaseTemp as significantly higher than all the other indices, with the other six index values all close to 0.3. None of the Sobol Total Effects indices for the days to emergence had a value of zero within their error margin, which was notably different from the results of the MARS method. The results for the days to flowering (Figure 3-2) also showed consistency between the Morris method and the MARS method. The results for the Sobol method (Figure 3-2) showed the index value for CriticalPhotoperiod as being 0.75, which was about one and a half times the values of the other input parameters. The Morris and MARS methods indicated that the importance of CriticalPhotoperiod was three to four times the value of the other input parameters. A similar situation was observed for the biomass output (Figure 3-2), where the range of values for the Sobol total effects index was much narrower than the range of values for either the Morris index or the MARS index of importance.

3.3.3 Comparison of additional results

The MARS method delivered several potentially useful additional outputs. Firstly, the MARS model building process developed a detailed and precise mathematical equation which described the contributions and interrelationships of all input parameters required to produce the predicted output value: a listing of the equation for the ARES emulator for Days to Emergence is included in Appendix I. Secondly, as an option when running the MARS model, additional statistics can be generated to

provide simple validation of the performance of the model. Statistics generated during the research phase of this project included: correlation coefficients (r), RMSE, residual root mean squared error (RRMSE), mean absolute error (MAE), and relative mean absolute error (RMAE). A third optional output of the MARS approach is the ability to create surface plots of the interactions between a set of any two input parameters, as shown in Figure 3-3. The panel on the left side of Figure 3-3 shows the interactions between twilight and critical photoperiod for their combined impact on biomass over the input parameters' normalised input ranges. Twilight is shown to have an increase in its effect as its value increases while critical photoperiod is low but has a constant effect as critical photoperiod nears its upper limit. Critical photoperiod is shown to be the predominant driver of the combined effect of these two input parameters, with increasing critical photoperiod having increased effect. The right-hand panel of Figure 3-3 shows a similar but opposite effect for the interaction between base temperature and optimum temperature effect. Base temperature is shown to be the primary driver of the interactions, with values low in its range having a significant effect while higher values have reduced effects, regardless of the value of the optimum temperature effect.

To produce these surface plots of the interactions between two input parameters and their combined contribution to the value of an output variable, a database of a great many input parameter value settings and the value of the sensitivity index, as it is affected by varying each of the other input parameters of interest, is required to be built. The Morris and Sobol sensitivity indices are calculated as the mean effect on the output value for the range of values of the input parameters used for the analysis run. To calculate the sensitivity indices for each point of interest across all input ranges and input interactions would require a very large number of individual SA runs. While, in theory, this could be done for Morris and Sobol sensitivity indices, the issues of running such large numbers of simulations makes the approach impractical.

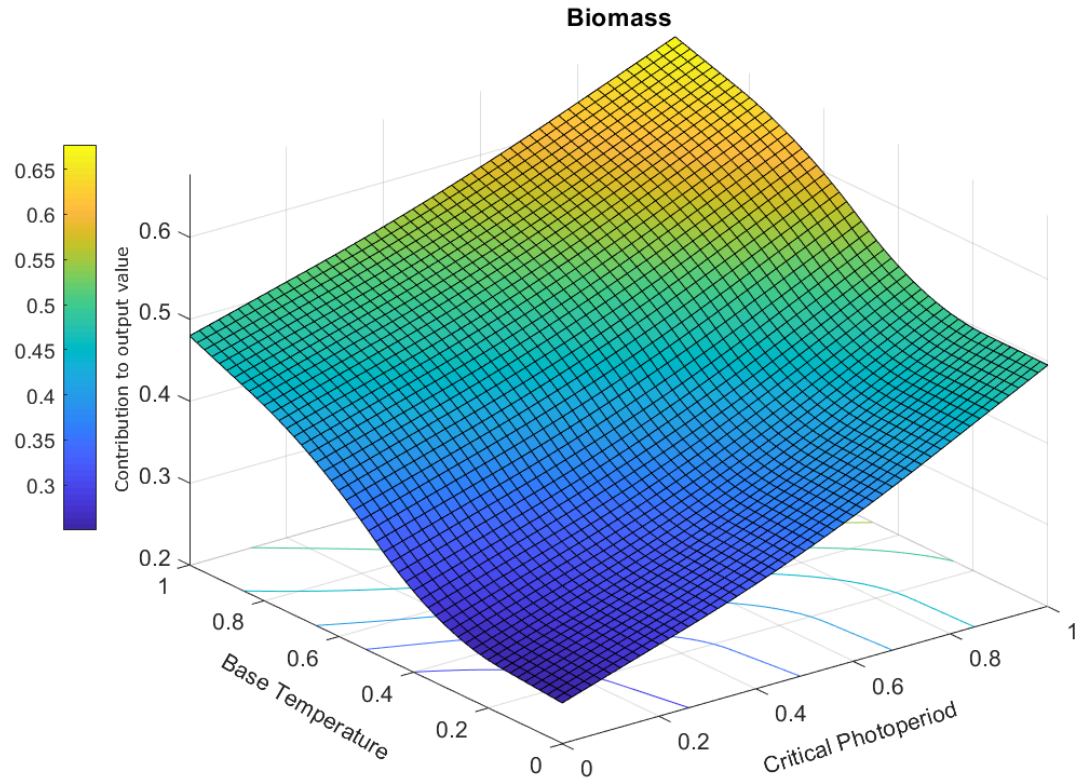


Figure 3-3. Surface plot produced from data generated by the multivariate adaptive regression splines (MARS) method.
This plot shows the interactions between two input parameters, BaseTemperature and CriticalPhotoperiod, and their combined contribution to the biomass output value. The input variables are shown with 0 to 1 value scales which equate to the minimum and maximum values of these variables used to develop the MARS model. The contribution to output value, the z-axis, indicates the relative importance the combined values have in contributing to the biomass value as a proportion of each input's maximum contribution for this output across all combinations of inputs.

Table 3-4. MARS model input importance indices.
Seven input parameters for each of seven models of output targets. The models were developed based on 15360 APSIM simulations, with the simulation data set split 80/20 between training and testing of the MARS models (12,288 training / 3,072 testing simulation data points). The accuracy of the models was assessed by calculating the R2 coefficient of determination and the COELM index for the agreement between the APSIM output values and the values generated by the MARS model.

Input Parameter	Output Targets						
	EmergenceDAS	FlowerDAS	MaturityDAS	Biomass	GrainWt	PodWt	NodeCnt
BaseTemp	100	30	33	100	22	25	35
OptTemp	37	24	16	58	24	24	26
MaxTemp	0	2	24	12	16	16	0
OptTempDD	14	13	28	15	7	7	20
Twilight	0	34	32	26	40	41	23
OptPhotoperiod	0	39	36	39	39	41	33
CriticalPhotoperiod	0	100	100	99	100	100	100
r^2	0.999	0.998	0.996	0.983	0.968	0.966	0.981
COE_{LM}	0.976	0.956	0.940	0.871	0.810	0.805	0.868

Values for seven MARS models developed to predict outputs are shown in Table 3-4. Included as measures of confidence are the R^2 and COE_{LM} indices. These values assess the accuracy of the MARS models to predict the true output value for the sets of test data. All models demonstrated very high predictive ability. The slightly lower COE_{LM} index values (in the range of 0.80 to 0.85) calculated for the grain weight and pod weight models reflect outputs that were more complex to calculate, or that were being influenced by input parameters that were not included in the analysis, and so have output values that could not be calculated with the same level of confidence as outputs that have less complex influences from inputs parameters.

By expanding the number of APSIM crop simulations to include a range of sowing dates and a range of crop production regions, the sensitivity of the phenology routine of the crop model to different environmental situations was able to be assessed. In place of the detailed analysis of the sensitivity indices of each SA method, the indices of each method were normalised to a common scale and compared using simple scatter plots. The results from the expanded set of simulations showed a reassuring level of agreement between the three SA methods. The agreement

between the Morris method and the MARS method (Figure 3-4b) was stronger than the agreement between the Sobol Total Order Index (Sobol St) method and either the Morris method (Figure 3-4a) or the MARS method (Figure 3-4c). The combined R^2 values for each of these comparisons; Morris to Sobol (St), Morris to MARS, and Sobol (St) to MARS; were 0.81, 0.89, and 0.82, respectively. The majority of zero value indices for the MARS method, evident in both Figure 3-4b and Figure 3-4c, represent the indices for CriticalPhotoperiod, OptPhotoperiod and Twilight for Emergence Day after Sowing for all sowing times at all locations. This is a reflection that these input features were not used in any of the MARS emulators for Emergence Days after Sowing.

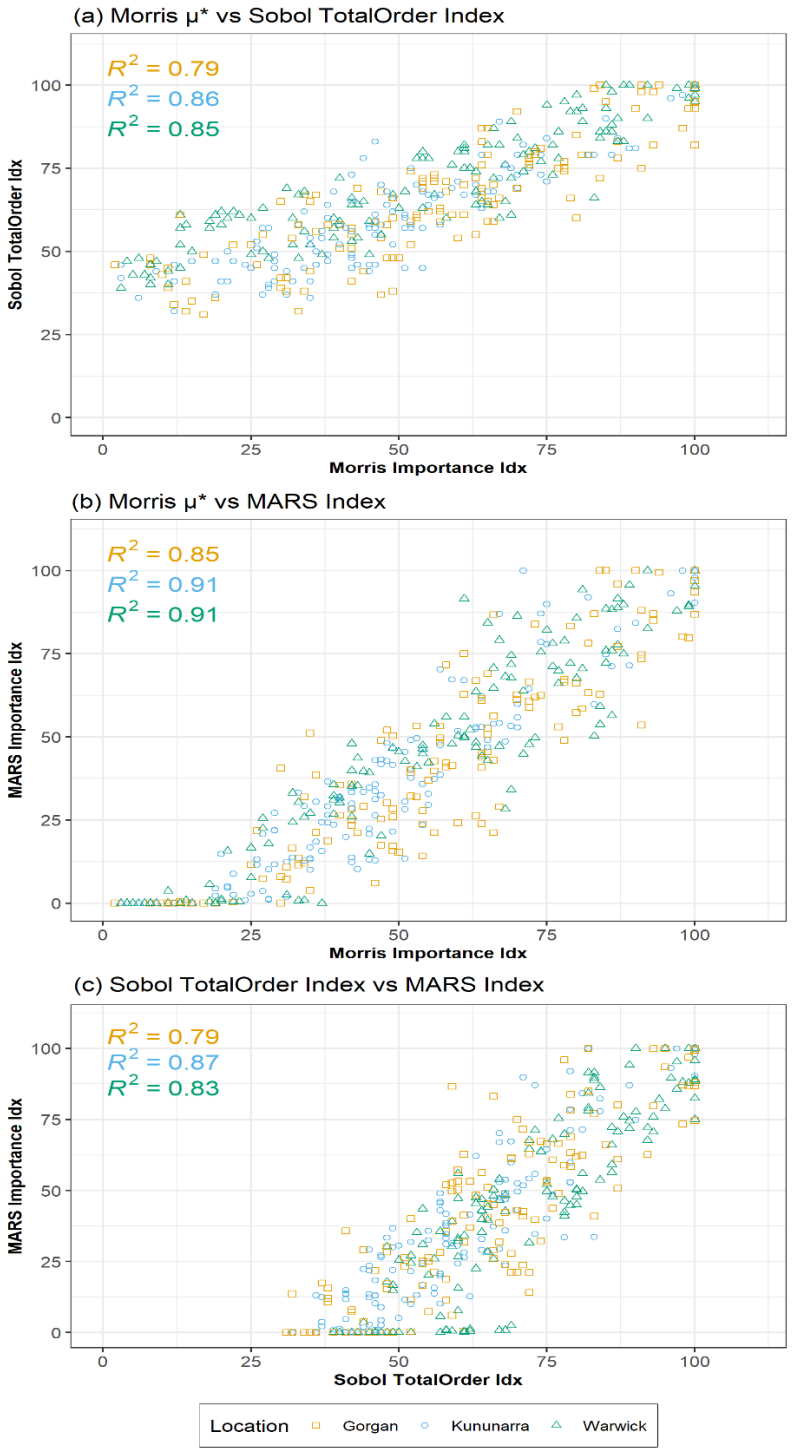


Figure 3-4. Scatter plots of the normalised index of importance values. All input factors on all outputs for each of (a) Morris versus Sobol Total Order (St); (b) Morris versus MARS, and (c) Sobol (St) versus MARS for the expanded data set of three crop production locations. The R^2 values of the linear regressions are shown. The index of importance values are a measure of the relative importance of each input parameter for its effect on a given model output parameter. Indices of importance range in value in this analysis from 0 for no contribution to the output, to 100 for the most significant contribution. Values for indices of seven input parameters and their importance for each of seven output parameters have been included in these data sets for analysis.

3.4 Discussion

This research has focused on three aspects of SA techniques; efficiency, accuracy, and level of insight into parameter interactions; to assess the applicability of the MARS method for the SA of process driven models, using a model of chickpea phenology as an example. Each method of SA uses a different approach to analyse the problem, produces different measures of interaction between inputs and outputs, and their accuracy is assessed using different statistical measures. This lack of a common approach to the analysis and subsequent measures of statistical performance of each method meant that observable end results with justifiable measures of performance have been used as the basis of comparisons.

3.4.1 Efficiency

Efficiency has been measured by the number of APSIM simulations required for the SA method to be able to consistently determine the significance that each input has on an output value. With the MARS method, the number of simulations required for accurate predictions of the output values and, by extension, for accurate prediction of input parameter importance indices, varied with the output target. Both the Morris and Sobol statistical methods take very different approaches to estimating the importance of input parameters compared to the data driven algorithm of the MARS method, and this variation in determining input importance between different output targets was not observed for these statistical methods. Outputs, such as days to emergence and days to flowering, are determined in the crop model by less complex processes than outputs such as biomass and grain weight. This is partially due to the time period involved, with emergence and flowering occurring earlier in the simulation process than the outputs which are reported at the time of harvest. The complexity in calculating the impacts of all inputs on the output value is reflected in the number of simulations required to determine these relationships. Consequently, the days to emergence and days to flowering required fewer simulations, about 4000, to establish very accurate predictions, while predictions for biomass were continuing to improve with greater than 10,000 simulations. Using the MARS method, 500 simulations produced R^2 and COE_{LM} index values of 0.95 and 0.84 for days to

emergence and values of 0.98 and 0.86 for days to flowering. The input importance indices with this number of simulations were consistent with the Morris method at 350 simulations, noting that the confidence intervals for the Morris method are quite wide. This indicates that, in situations where the calculation of the output value is not complex, the MARS method can have computational efficiency of the same order of magnitude as the Morris method. For more complex output values, computational requirements were in the order of ten times that required for the Morris method, being 4000 simulations as compared to 350 simulations. The MARS method was still twice as efficient as the Sobol method at 9000 simulations in this comparative study. Consistent with other research (Morris 1991; Campolongo et al. 2007; Pianosi et al. 2016; Sarrazin et al. 2016; Jaxa-Rozen & Kwakkel 2018) the Morris method was found to be very computationally efficient, returning stable predictions of the importance of all input parameters after 350 simulation runs. This is within the guidelines for expected values reported by Pianosi et al. (2016). The Sobol method required approximately 9000 simulations to be run to determine the first order and total order effects indices with low margins of error. These findings show that the MARS approach is closer to the Morris method in terms of efficiency. Both these methods may be considered to be more efficient than the Sobol method. The focus of the Sobol method is to produce a comprehensive sensitivity analysis with the ability to analyse multiple orders of input parameter interaction.

3.4.2 Accuracy

The second aspect of SA that has been considered was the accuracy of the analysis of input parameter importance. Each analysis method has a different way of determining the significance that variations in the input parameter is having on the output. Morris uses EET, Sobol apportions the variation in the output value to individual inputs and MARS analyses the increase in the value of the generalized cross validation (GCV) when removing an input variable. It is expected that the differing approaches will yield some variations in the values reported for input parameter importance. Where inconsistencies are observed between the outputs of the different methods, expert knowledge of the underlying relationships is required to assess which result has the higher probability of being correct. A result of

particular interest in the analysis of the chickpea phenology is that of the input importance indices for the days to emergence. In the field, soil temperature is the primary driver for emergence with minimal influence from day-length or solar radiation, other than from the heating effect on the soil. The chickpea model uses minimum and maximum temperatures from the climate file to calculate the accumulation of thermal time and this is used to determine the rate of emergence of the crop. The Morris method ranks the importance of the factors, most significant to least significant, as: BaseTemp, OptTemp, OptTempDD, MaxTemp, OptPhotoperiod, Twilight, and CriticalPhotoperiod; with the three light factors being rated close to zero. This was an expected result. The results of the MARS method agreed with the results of the Morris method. For the MARS method, the four input factors; CriticalPhotoperiod, OptPhotoperiod, Twilight and MaxTemp; are reported as not used in the calculation of the days to emergence. That is, for the MARS emulator, the days to emergence was completely insensitive to the values of these factors. The results of the Sobol analysis were less clear. While the ranking of the factors could not be considered as different, within the margins of error, from the rankings of the other two methods, the values of the total effects' indices were consistently higher than expected. All input parameters, apart from BaseTemp, were analysed as having total effects index values, within margins of error, of 0.3. All input parameters apart from BaseTemp having consistent effect was not in agreement with this expectation, nor with the results of the other two methods. The reason for these Sobol values being higher than the Morris and MARS values was not immediately evident but is a function of the interactions between input parameters, the underlying concept of proportion of variance and the effect that normalising these values has on their reported value. What was clear, however, was the more consistent agreement between the results of the Morris and the MARS methods than either of these methods with the results of the Sobol method. This observation can be attributed to the fact that the Sobol analysis approach is focused on providing a comprehensive and accurate analysis of input parameter sensitivities, while the Morris and MARS approaches are more suitable for efficient estimations of approximate sensitivities.

3.4.3 Confidence in reported values

As part of the review of the accuracy of the reported values, consideration was also given to the level of confidence that could be placed in the reported values, and if there were any additional indicators that might highlight erroneous analysis. Morris and Sobol report only the values calculated from the data analysed. Careful review of results and expert knowledge is required to detect if errors might have occurred. For example, early data sets for this research included combinations of input parameters which resulted in a failure to produce a crop yield in a small number of instances for one location. While valid for the crop simulation, these few data combinations corrupted the Sobol analysis. The error was not found until the results were shown to be inconsistent with those of the Morris and MARS analyses and a detailed review was undertaken to identify the problem. For the MARS method, the level of confidence that can be placed in the order of the input parameters and the values of their indices of importance corresponds directly to the level of confidence that can be placed in the ability of the emulator to predict the output values. This approach forms the basis of many of the input selection analyses reported in literature (Kursa & Rudnicki 2010; Salcedo-Sanz et al. 2014; Chowdhury et al. 2015; Li et al. 2016; Prasad et al. 2017; Chekole 2019; Gebauer et al. 2019). In this research both the R^2 and COE_{LM} index values have been used as measures of the accuracy of the emulator in predicting the output values. Two factors affect the accuracy of the emulators. Firstly, the number of simulations used to develop the emulator, with less complex outputs, such as the days to emergence, requiring less simulations than outputs that involve more input factors and more complex computation, such as biomass. The predictive abilities of all emulators improved with more simulation data (Ma et al. 2014; Şahin et al. 2014; Taormina & Chau 2015; Karandish & Šimůnek 2016). Secondly, the accuracy of the emulators appeared to be limited if the input parameters did not provide all the data required to calculate the output values, even when large numbers of simulations were evaluated. Input parameters other than the phenologically sensitive ones being assessed, such as rainfall in the case of biomass, would be expected to contribute to the output value. The ability of the emulator to accurately predict the output produced by the biophysical crop model was compromised, and this was reflected in the lower values of R^2 and the COE_{LM} index which do not improve significantly when more simulations were included in the

training data set. During the assessment process, it was found that the R^2 and COE_{LM} indices were simple and intuitive measures with which to assess the level of confidence that could be placed in the predictions of the SA outcomes for the MARS method.

3.4.4 Additional benefits

While all three methods return index of importance values (Morris 1991; Sobol' 2001; Campolongo et al. 2007; Jekabsons 2016), what is not evident from the reported results is how this index of importance value might vary across the range of input values and how one input might be interacting with another input parameter. Once the MARS emulator was built during the analysis process, surface plots were created (e.g., Figure 3-3) showing the interactions of a pair of selected input parameters and their combined contribution to the output over the ranges of values tested for the inputs on a normalised (0 – 1) scale (Jekabsons 2016). This graphical output can be useful in providing a simple visual understanding how parameters are interacting, for assessing if a response is linear or not, to identify if a local minima or maxima has been targeted or assist in refining input parameter settings by targeting a feature in the response surface that is of particular interest. The Morris method provides a sigma value which indicates the level of non-linearity of the input parameter or its level of interaction with other input parameters, while the value of $\mu*$ itself is a mean of all the elementary effects of the specific input parameter. The Sobol method can evaluate first, second and further orders of effects (levels of parameter interactions) as well as the total order effects, but there is no simple way of visualising how these interactions vary across the input parameter's selected range of values. Another feature that the MARS method provides is that of the emulator that is generated. For some applications, the creation of this emulator may well be the focus a development exercise (Kouadio et al. 2018; Lawes et al. 2019). This research has demonstrated that a range of modelling practitioners may well derive significant benefits for SA out of the process of developing the emulator. Whether or not the emulator itself has a particular use would depend on the task being undertaken.

One disadvantage of the MARS method is that model interpretation can be difficult if two input parameters are closely correlated. The MARS algorithm will randomly select one of the input features and develop the emulator based on its influence. The correlated feature may be evaluated as not contributing any additional predictive power, and so will be assessed as having a zero or low index of importance (Boehmke & Greenwell 2019). The Morris and Sobol methods will assess each input's effects on the output value, regardless of correlation with any other input feature. Another disadvantage of the MARS method is that it does require extra steps to generate the emulator before the SA results are known. This involves both some data manipulation of the APSIM simulation outputs, as well as computational processing time for the generation of the emulator. For this research, the development of the MARS emulators was undertaken using MATLAB® (ver. R2017b). There are alternative code libraries available for other language platforms including R and Python (Rudy 2017; Milborrow 2019). As an indication of the additional computational time required for the MARS method, where this study required approximately two hours to run 15,360 APSIM-NextGen simulations, an additional 45 minutes was required for the MATLAB routines to generate the seven emulators for the seven selected output targets. This additional time covered the generation of the individual emulators, which took between three to twenty minutes for an individual emulator depending on the complexity of the calculation process, the testing of the emulators, generation of log files and summary spreadsheets and the generation of several surface plots per emulator. While it is a significant amount of time, the potential time and computational savings offered over the Sobol method are not insignificant. When the additional insights offered by the MARS method are taken into consideration, the benefit to cost ratio of using this method becomes attractive. Some advantages and disadvantages of the different methods are listed in Table 3-5.

Table 3-5. Advantages and disadvantages of using the Morris, Sobol and MARS methods.
Findings from the study allow comparisons with using the MARS methods to undertake sensitivity analysis of process driven models.

Method	Advantages	Disadvantages
Morris	Computationally very efficient. Elementary Effects Test (EET) a simple concept to understand and form accurate conclusions from.	No detailed analysis of input parameter interactions.
Sobol	First, second and total order interactions of input parameters possible.	Computationally expensive to very expensive. Proportion of variation of response analysis approach requires careful consideration of results to form accurate conclusions.
MARS	Computationally more efficient than Sobol method (40% to 80% gains compared to Sobol first order analysis). Mathematical equation produced detailing the contribution and interactions of each input parameter. Data produced by emulator allows 2D and 3D graphics to be plotted to allow visualisation of parameter interactions and output response. Analysis of the goodness of fit value for the emulator's predicted values against the observed values gives a good measure of the accuracy and trustworthiness of the reported SA results.	Additional computing steps and computing power required to generate ML emulator. Results can be misleading if closely correlated input parameters are included. One parameter may be selected while the other is omitted from inclusion in the development of the emulator.

3.5 Conclusion

The results of the research presented in this chapter have answered in the affirmative the first of the research questions of this thesis, that is: that there are features of MLEs, specifically input variable importance, which are comparable for the purposes of SA, though not identical, as the indices generated for the input parameters of the process-driven model itself by the Morris and the Sobol methods. Morris and MARS

are most directly comparable as they involve simplified methodologies for the calculation of sensitivity indices, while the Sobol approach is a more comprehensive analysis method and generates sensitivity indices for (potentially) multiple orders of input parameter interactions. From this point of view, the Sobol approach may be considered to produce a more accurate sensitivity analysis than either of the Morris or MARS approaches. Development, parameterisation and application of process driven models require that appropriate adjustments are made to the input parameters that drive the model's response functions. The determination of which input parameters are most influential in determining particular outputs is determined using SA. More advanced SA techniques can reveal how those input parameters interact over a selected range of input values. This research has shown that the data centric MARS method produces sensitivity importance indices with similar computational efficiency of the Morris method for simply calculated output values and can be about twice as efficient as the Sobol method for complex output values. The confidence that can be placed in the order and the values of the importance indices can be crosschecked in the case of the MARS method with the statistical accuracy of the emulator. Additional insights into how the input parameters affect the model's output values are offered by MARS method but are not offered by the Morris or Sobol approaches.

CHAPTER 4: PAPER 2 - Comparison of machine learning methods emulating process driven crop models

Preamble

A range of ML algorithms was considered for inclusion in the previous research chapter, Chapter 3, of this thesis. These included random forests (RF), artificial neural networks (ANN), extreme learning machines (ELM), and multivariate adaptive regression splines (MARS). The MARS algorithm was selected due to a user's ability to analyse what was happening in calculating the indices of parameter importance. This research was exploratory in nature, so it was important to be able to investigate what was happening if the results did not match expectations. Each of the other ML approaches operated as 'black boxes', where the internal calculations of the model are not available for analysis. The MARS models allowed the objective of this phase of the research to be met by showing that there is a high correlation between the parameter importance indices produced by the MARS method and the indices produced by the Morris method, and to a lesser extent, to the indices of the Sobol method.

The next phase of the research, the focus of Chapter 4 of this thesis, involved expanding the number of parameters of interest for inclusion in the SA. This was done by evaluating chickpea time-of-sowing simulated trials across multiple locations in Australia. In preparing the experimental design and undertaking preliminary testing for this phase of the research, it became evident that the MARS method eliminated from its models, inputs that it assessed as having no, or very small effects, on the output values. The dropping out of input parameters rendered thorough SA impossible. An alternative approach to conducting SA using ML models was sought. It was identified that the speed of model development and highly efficient execution times for the MLEs could be utilised to alleviate the computational burden of undertaking traditional approaches for SA. The focus of the second and third research chapters (Chapters 4 and 5) of this thesis then pivoted to assessing the potential to use MLEs to run the very high number of simulations required for SA. The MARS method was included to maintain continuity with the

first research chapter. The RF and ANN approaches were included in addition to MARS method as they were the most used ML algorithms in the context of agricultural crop modelling. Subsequently, only the RF and ANN models were assessed as being suitable to undertake the Morris and Sobol analyses presented in Chapter 5 of this thesis. The pivot in thinking between Chapter 3 and Chapter 4 is not evident in the text of the chapters and could lead to confusion on behalf of the reader. What has been learnt from the need to redirect the research focus is that there is no single ML approach that is suitable for all situations, and that without quite thorough testing, there is no way of being certain that a particular approach will fulfil the objectives of a given task. In addition to this pivot in thinking, the software environment used to develop and test the MLEs was changed from MatLab (the machine learning thesis supervisor's preferred platform) to an R environment, as this was what APSIM used as its interface for statistical analysis. This change is reflected in the source code of the algorithms implemented, and the reported speed of model development and model execution in the case of the MARS/ARES models.

4.1 Introduction

The agricultural and environmental science disciplines have long utilised the power of computer modelling for scientific enquiry and knowledge advancement (Jones et al. 2016). Mechanistic models have been developed for many biological and environmental processes, and these models have subsequently been integrated together to form whole of system simulation computing environments which are complex and computationally expensive to configure, validate and run (Keating et al. 2003; Holzworth et al. 2014). New developments in computer modelling are often driven by the need for cost reduction and improved efficiencies, as these two concepts are integral in the functioning of most modern economies and exist as non-negotiable goals for most projects. As computing costs have progressively reduced over the past few decades, the size and complexity of experiments and analysis based on computer modelling has grown. These simulation experiments can require the running of many thousands, or even millions of model runs, and produce extensive amounts of data (e.g. Phelan et al. (2018) and Casadebaig et al. (2016)). A reduction

in the computational costs of producing large amounts of data is one area that is a target of improved efficiency efforts.

Machine learning (ML) approaches for predictive modelling are having a significant impact on many areas of society, including areas of scientific research, not the least of which are agricultural and environmental sciences. Computational efficiency in producing predicted outcomes is one benefit of ML algorithms (Balakrishnan & Muthukumarasamy 2016; Karandish & Šimůnek 2016; Shastry et al. 2016; Singh et al. 2017; Ryan et al. 2018; Feng et al. 2019; Niazian & Niedbała 2020). Much research involving ML technologies revolves around the approaches being able to take diverse data sources, such as remote imaging and multiple sensor inputs, and predict outcomes such as vegetation type, soil water content, biomass and crop health (Shakoor et al. 2017; Prasad et al. 2018; Lawes et al. 2019; Feng et al. 2020; Obsie et al. 2020; Zhang et al. 2020; Fajardo & Whelan 2021; Guo et al. 2021; Paudel et al. 2021), while the potential computational efficiency gains have received much less attention. Systems modelling, be it for weather, environmental or agricultural systems, are undertaken using complex, process driven models. The agricultural production systems simulator (APSIM-NextGen) (Holzworth et al. 2018) is one such modelling system in the agricultural and environmental sciences domain. While process driven modelling systems like APSIM-NextGen provide extensive modelling and research opportunities due to their complexity and flexible configuration, they are computationally expensive. This limits experimental designs where resources are insufficient to run large numbers of simulations (e.g. Casadebaig et al. (2016)). Sensitivity analysis (SA) often requires large numbers of simulations to evaluate the interactions between changes in input factor values and the effects these have on target output values. While the expectations and requirements to validate models using SA continues to grow (Razavi et al. 2021), the ability to undertake thorough SA of complex systems models is compromised by the limitations imposed by computing power.

There are studies which consider the use of emulators to improve the efficiency of performing SA on complex environmental models. For example, Stanfill et al. (2015) and (Ryan et al. 2018) both used the statistical approach of generalised additive models to improve computational efficiency of SA applications. Wallach and

Thorburn (2017) and Sexton et al. (2017) discuss the relatively new approach, at least in crop modelling research, of utilising machine learning based emulators (MLEs) to improve computational efficiency in uncertainty analysis. Apart from these studies, little research, has been done on the potential of using ML approaches to improve the computational efficiency of SA of complex process-driven biophysical models. The research undertaken in Chapter 3 of this thesis looked at comparing SA measures generated using traditional statistical methods applied to the process-driven model directly, and the measures of variable importance generated for an MLE. The MLE approach showed little or no gain over the efficient Morris method for the purpose of screening parameters. It did, however, demonstrate significant potential computational gains over the Sobol variance decomposition method. More research is required to assess if a wider range of biophysical modelling scenarios can similarly benefit from using ML to undertake SA. Underlying this question is the issue of whether any particular ML approach is better able to be trained to predict the outputs of complex systems models. This issue has not been adequately addressed in literature.

The objective of this research was to demonstrate that, by using input parameters used to configure and run APSIM-NextGen chickpea crop simulations, MLEs could be developed which are able to predict selected APSIM model outputs. If this is demonstrated, then the use of these MLEs would allow the replacement of the APSIM system model with a small and efficient predictive model that is effective for the range of input parameter variations used in the training data set. These MLEs could then be used to undertake SA of the underlying modelled relationships. A further objective was to test and see, when the input parameters used to develop the MLEs were diverse enough and contain enough variation in values observed, if the MLEs developed were robust enough to be able to accurately predict crop outputs for all locations within the regions covered by the training data set. To fulfil these objectives, the APSIM-NextGen chickpea model was configured to simulate crop production over a 120-year period at seven locations throughout the chickpea production regions in Australia. Six model outputs were reported and further used to train emulators based on three ML algorithms: 1) artificial neural network (ANN), 2) multivariate adaptive regression splines (MARS) and 3) a random forest (RF)), using 24 input factors from the APSIM simulations. The MLEs were assessed for

predictive accuracy, input variable importance and computational effort. The assessments of model performances were conducted for the locations for which the MLEs were trained, as well as two additional locations not included in the training data set to test emulator robustness.

4.2 Methods

Three MLEs representing different ML algorithmic approaches were developed from data generated from APSIM simulations of chickpea growth, development, and yield for seven locations in the Australian chickpea production regions. The MLEs were trained on 80% of the generated data and then tested using the remaining 20% of data. Goodness-of-fit of emulator generated data against the original APSIM data for six model outputs were analysed and are presented in the results section. The output targets were as follows: 1) days from sowing to emergence (EmergenceDAS), 2) days from sowing to flowering (FloweringDAS), 3) days from sowing to first fruiting pod (PoddingDAS), 4) days from sowing to crop maturity (MaturityDAS), 5) above ground crop biomass at harvest (kg/ha) (Biomass), and 6) weight of harvested grain (kg/ha) (GrainWt); cover some of the more significant chickpea model outputs for monitoring and assessing crop growth from emergence to harvest. Additionally, two test locations within the chickpea production area, but not included in the original seven locations, were used to generate the ML data that was then compared against the APSIM generated outputs for further benchmarking purposes.

4.2.1 Computing environment

All simulations and data analyses were performed on an Intel Core-*i*7 7600U CPU 2.9 GHz based computer with 16 GB RAM running Microsoft Windows 10 operating system. The APSIM version used was APSIM-NextGen (version 2020.02.05.4679) (Holzworth et al. 2018). The APSIM-NextGen prototype chickpea model was used as the crop model. Built-in features of the APSIM-NextGen User Interface were used to configure and execute factorial simulation experiments which generated the data used for building the MLEs.

4.2.2 Machine learning based emulators

The MLEs were developed and run in an installation of R (version 4.0.3 (2020-10-10)) (R Core Team 2020) in Microsoft Windows. The R environment was also used for data preparation and manipulation, reporting and graphics generation, with the packages ggplot2 (version 3.3.3) (Wickham 2016) and other packages from the tidyverse library (version 1.3.0) (Wickham et al. 2019) primarily used for these functions. The three MLEs, which are detailed below, were: nnet representing an ANN, Earth representing a MARS implementation and a Random Forest representing a decision tree implementation.

4.2.2.1 Artificial Neural Network

An ANN is a computing paradigm which consists of a massively interconnected network of nodes acting in parallel which simulate the actions of biological neurons. Each network connection is characterised by a weighting factor. Each neuron calculates the sum of its weighted inputs and produces an activation level output value via a generally nonlinear activation function. Models based on ANNs are developed by adjusting the number of neurons, number of layers of neurons (topology), neuron characteristics of activation functions and bias, and the sensitivities to training responses (Lippmann 1987). In this experiment, the standard R library, *nnet* (version 7.3-15, 2021-01-21) based on the work of Venables and Ripley (2002) has been used to implement a feed-forward neural network with 20 nodes in its hidden layer and utilising 100 iterations for self-configuration. These settings were established by trial and error as optimal for predictive accuracy. The number of nodes was tested over the range of 10 nodes to 40 nodes, using increments of 2 nodes. The iterations for self-configuration were tested over a range of 50 to 200 in increments of 10. Default settings were utilised for all other model parameters. The ANN algorithm has been included in this study because of its general applicability in environmental and biological studies and its wide use as a baseline for comparative ML studies.

4.2.2.2 Multivariate Adaptive Regression Splines

The MARS method for modelling is a flexible regression modelling approach which has its roots in the recursive partitioning approach used in some forms of regression analysis. Continuous models with continuous derivatives are generated by repeatedly splitting product regression splines and introducing new basis functions for additional splines. This continues until the addition of more splines fails to improve the fitting of the response curves to the sampled data (Friedman 1991a). For this study, the *earth* package (version 5.3.0) (Milborrow 2020) in R was used to implement the MARS algorithm. The MARS algorithm has been included in the ML approaches for this study to follow on from the research conducted in Chapter 3 of this thesis. It provides an interesting comparison for computational performance and predictive accuracy with the other two pure ML based approaches.

4.2.2.3 Random Forest

Random forests are a computing paradigm based on an ensemble of decision trees. A random selection of features is used to split each node, with the accuracy of prediction used to weight the strength of each tree. The generalisation error for forests reduces as the number of trees increase and correlation between strong individual trees increases. Random forests have been shown to be quite robust with respect to outlier data points and noise within datasets (Breiman 2001; Sexton & Laake 2009). The implementation of the RF algorithm used was the *randomForest* package (version 4.6 – 14 2018-03-22) (Liaw & Wiener 2018) in the R environment. Default values were used for all model settings. The default settings include that the number of features to be included in each decision tree is (p/3), where p is the number of input parameters. The default settings also specify that the algorithm calculates, via its internal code, the number of decision trees that are formed to optimise its predictive accuracy during its learning phase. The RF algorithm has been included in this study because of its wide applicability and use in agricultural and environmental modelling.

4.2.3 Simulation configuration

Simulations of chickpea crops were configured in APSIM-NextGen for seven locations throughout the chickpea growing regions in Australia (Figure 4-1) for 120 years (1900 to 2019). Reports were configured in APSIM to record all relevant input settings, summarise weather details, report the days after sowing of key crop development phases and report final above ground biomass and grain yield. The report produced one row of data at each harvest event, the report row constituting one 'observed data' set. The input settings and summarised weather details were used as the inputs to train the MLEs, with the crop development times, biomass and yield details used as the output targets for training and testing. In addition to the seven locations used to train the MLEs, two extra test locations, not included in the training and testing data sets, were used to test the robustness of the MLEs for locations outside the development data set.

4.2.3.1 APSIM simulation configuration

A typical soil type for the area was selected for each location. The details of these are shown in Table 4-1. All simulations had plant available soil water reset to 70% capacity on 1st March in each simulation year. Sowing dates were simulated for each 5-day interval from 30 March until 5 August. Row spacing was consistent at 0.5 m, sowing depth was 50 mm, and plant population was 30 plants/m^2 for northern sites (above 32^o S), and 40 plants/m^2 for southern sites (below 32^{o}S). Two chickpea genotypes, Desi and Kabuli, were sown at each location, with three varieties for each genotype; Seamer, HatTrick and CICA1521 for Desi; Monarch, Almaz and Kalkee for Kabuli. The genotypes differed from each other in four phenological parameters, each defined in terms of thermal time; ShootLag, VegTarget, LateVegTarget and FloweringTarget.

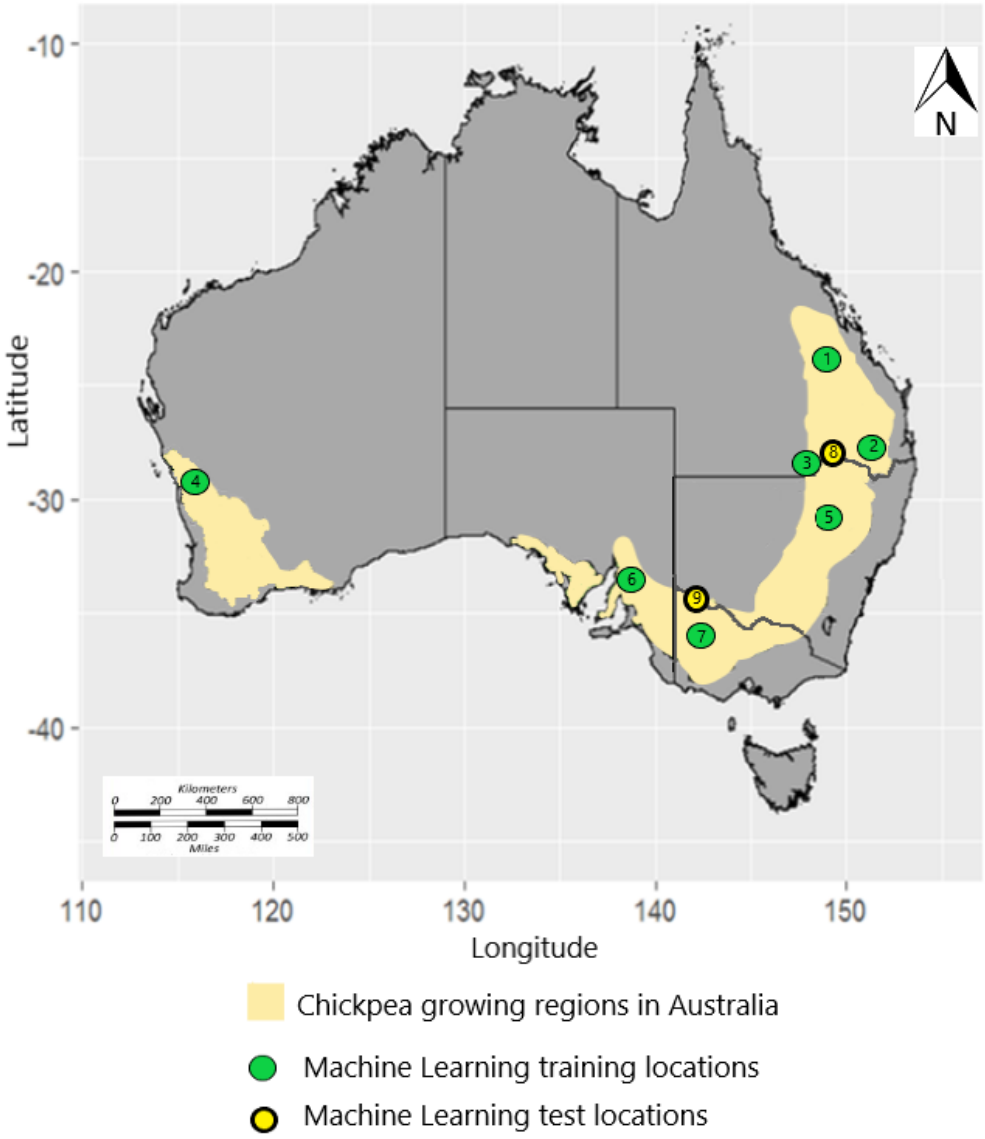


Figure 4-1. Chickpea growing regions in Australia and the seven locations used to develop machine learning MLEs.
These seven locations (green dots), ordered by latitude are: 1) Emerald, Qld, 2) Bongeen, Qld, 3) Mungindi, NSW, 4) Mingenew, WA, 5) Gunnedah, NSW, 6) Clare, SA, and 7) Horsham, Vic. Additionally, two independent test locations (yellow dots) were also included: 8) Goondiwindi, Qld, and 9) Mildura, Vic.

Table 4-1. Soil descriptions by location used for chickpea crop simulations. The soil type descriptions and reference codes refer to the APSoil database of soils from which the properties of the modelled soils were sourced.

Location	APSoil description and code	Profile depth (mm)	Plant available water capacity (mm)
1. Emerald	Grey Vertosol (No 106)	1500	282
2. Bongeen	Black Vertosol (No 001)	1800	335
3. Mungindi	Grey Vertosol (No 906)	1800	339
4. Mingenew	Clay (No 71)	1800	320
5. Gunnedah	Black Vertosol (No 1174)	1800	285
6. Clare	Clay Loam on Clay Loam over Clay (No 290)	1500	284
7. Horsham	Grey Cracking Clay (No 1008)	1300	341
8. Goondiwindi	Grey Vertosol (No 219)	1800	262
9.Mildura	Sandy Loam over Sandy Clay Loam (No 332)	1400	142

4.2.3.2 Machine learning emulator inputs

The development and evaluation of MLEs were assessed for six output targets of interest for chickpea production: EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt. The final development involved data for seven production locations around Australia, with additional testing of the MLEs undertaken using two additional locations which were not included in the development and testing data set. Input factors (Table 4-2) used to train the MLEs were sourced from the reports generated by APSIM-NextGen. Weather details were summarised for three blocks of time from the day of sowing: 0 to 30 days, 31 to 60 days and 61 to 90 days. Temperatures, both maximum and minimum, were averaged for each time block, while rain and solar radiation were summed to give totals for each time block. Soil water was represented in two ways. Firstly, a single value of how much plant extractable soil water (mm) was present at sowing (SowingESW) was included. Secondly, the soil's water holding capacity, measured as the plant available water capacity (mm) (PAWCmm) and the sowing water content as a fractional value of this (FracPAWCmm), were included in the input parameters.

These two measures are highly correlated within a soil type, but variable between soil types.

Table 4-2. Machine learning input factors used for the development of each of the three ML emulator types.
The same inputs were used to develop MLEs for each of the six output targets.

Input Factor Name	**Description**
AvgMaxT0_30	Average maximum temperature for 0 to 30 days after sowing
AvgMaxT31_60	Average maximum temperature for 31 to 60 days after sowing
AvgMaxT61_90	Average maximum temperature for 61 to 90 days after sowing
AvgMinT0_30	Average minimum temperature for 0 to 30 days after sowing
AvgMinT31_60	Average minimum temperature for 31 to 60 days after sowing
AvgMinT61_90	Average minimum temperature for 61 to 90 days after sowing
Cv	Chickpea cultivar (coded as 1 to 6 for the different genotype/cultivar combinations used)
FloweringTarget	Phenological parameter. Differs between genotypes.
FracPAWCmm	Amount of soil water present at sowing. As a fraction of PAWC.
Lat	Latitude of the sowing location.
LateVegTarget	Phenological parameter. Differs between genotypes.
PAWCmm	Soil's plant available water capacity to 1.5m depth (mm)
Population	Sown plant population in plants /m^2
Radn0_30	Sum of solar radiation for 0 to 30 days after sowing
Radn31_60	Sum of solar radiation for 31 to 60 days after sowing
Radn61_90	Sum of solar radiation for 61 to 90 days after sowing
Rain0_30	Sum of rainfall for 0 to 30 days after sowing
Rain31_60	Sum of rainfall for 31 to 60 days after sowing
Rain61_90	Sum of rainfall for 61 to 90 days after sowing
ShootLag	Phenological parameter. Differs between genotypes.
SowDepth	Sowing depth of crop
SowingDOY	Sowing date as Day Of Year
SowingESW	Extractable soil water at sowing
VegTarget	Phenological parameter in thermal time. Differs between genotypes.

4.2.3.3 Machine learning emulator targets

Six APSIM-NextGen chickpea model outputs were recorded in the APSIM reports, along with their corresponding input factor values, to create 'observed data' sets. Each of the three ML approaches was assessed on how well an emulator could predict the output values generated by the APSIM-NextGen simulation, as well as assessing the time taken, indicating computational effort required, to develop each

ML emulator. This was undertaken on a comparative basis to assess differences between the various approaches.

4.2.4 Statistical measures for 'goodness-of-fit'

The 'goodness-of-fit' between the APSIM generated target values and those generated by the MLEs was assessed using the following statistical measures: (4.1) mean bias (MB), (4.2) mean absolute error (MAE), (4.3) root mean squared error (RMSE), (4.4) coefficient of determination (R^2), and (4.5) coefficient of efficiency (COE_{LM}, also known as Legates-McCabe index) (Legates & McCabe Jr 1999). These metrics were used to compare the ML predicted versus APSIM-generated values datasets to determine the degree of match between the tested datasets.

Mean bias (MB) measured in days or kg/ha, depending on the output

$$MB = \frac{\sum_{i=1}^{n}(y_i - x_i)}{n} \tag{4.1}$$

Mean absolute error (MAE) measured in days or kg/ha, depending on the output

$$\text{MAE} = \frac{\sum_{i=1}^{n}|(y_i - x_i)|}{n} \tag{4.2}$$

Root mean squared error (RMSE) measured in days or kg/ha, depending on the output

$$RMSE = \sqrt{\left(\frac{\sum_{i=1}^{n}((y_i - x_i)^2)}{n}\right)} \tag{4.3}$$

Coefficient of determination (R^2)

$$R^2 = \left(\frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}} \right)^2 \tag{4.4}$$

Coefficient of efficiency (COE_{LM} : Legates McCabe index)

$$COE_{LM} = 1 - \left[\frac{\sum_{i=1}^{n} | y_i - x_i |}{\sum_{i=1}^{n} | x_i - \overline{x} |} \right] \quad (4.5)$$

In equations 4.1 through 4.5: '*n*' is the number of pairs of (APSIM-generated (*x*), predicted (*y*)) values, where 'APSIM-generated' is the APSIM generated value; and 'predicted' is the ML emulator generated value for the model output. '*i'* is the output generated from the i^{th} set of input parameters. The six target outputs generated were: EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt.

4.2.5 Variable importance

The contribution that each input factor (Table 4-2) has towards the value of the output target (EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass or GrainWt) is calculated by the ML algorithm. The values reported (Figure 4-2) have been standardised so that the most significant input is assigned an importance index value of 100, non-contributing inputs are given a value of zero (0) and all other inputs are rated with index values proportionate to the most influential input. The implementations of the ML algorithms for ANN, MARS and RF all had internal routines that calculated and reported variable importance indices. Each of these routines was configured to report index values rated on the reduction in the residual sum of squares (RSS) value of generated predictions versus the actual target values when the input parameter being assessed was included in the model. That is, the input that resulted in the greatest reduction in the RSS when it was added to the algorithm was assigned an importance index of 100 (Friedman 1991a; Milborrow 2019).

4.3 Results

4.3.1 Performance based on training data set

Results from the training data set, where the MLEs were trained on a random subset of 80% of the data and then tested on the unused 20% of data, showed that each of the three ML approaches, ANN, MARS and RF algorithms, can produce MLEs with significant predictive accuracy for each of the six crop output targets (Table 4-3). There were no observed occurrences of any model encountering overfitting issues, which would have been evidenced by the accuracy of the predictions of the validation data set being significantly lower than the accuracy for the training data sets. All reported values are those for the validation data sets for each MLE. The accuracy of prediction, the importance of input variables used to achieve these predictions, and the computational effort required to develop the MLEs, did vary between the approaches. Across all outputs, the RF emulators showed the best and most consistent accuracy at prediction. This, however, come at significant computational investment.

4.3.1.1 Graphical and statistical analysis of ML approaches

A visual inspection of the ML predicted versus APSIM generated data plots (Figure 4-2) confirmed the accuracy of the predictions for the six target outputs (EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt). The corresponding values from the statistical analyses of the data of these graphs is presented in Table 4-3. Of note is the superiority of the RF emulators' predictions for each output target. All three MLEs produced exceptional results for predicting the start of flowering (FloweringDAS). Regional variations are evident for each ML emulator with northern locations flowering after a shorter duration than locations with more southern latitudes (cooler climates). Interestingly, predictions of podding date were much less precise for each of the MLEs, with noticeably wider variations occurring at Mingenew. This indicates that some crop growth factors used within APSIM which affected early pod development were not included in the input parameter details, and that this was experienced to a greater extent at Mingenew than what it was in the other locations. While producing the most accurate predictions of podding date for most locations, most noticeably for Horsham, the RF emulator's predictions for Bongeen were slightly less accuracy than other MLEs. There was no clear indication as to why this was the case. The above ground crop biomass and the crop yield, reported as GrainWt, were the least predictable outputs for each ML

algorithm. Each of the MLEs, on average, under-predicted the values for both biomass and crop yield, as indicated by the negative mean bias values (Table 4-3). The RF emulators had about one quarter the amount of variance of the other two MLEs, as shown by the mean absolute error (MAE) values (Table 4-3). The ANN and MARS emulators each produced predictions with a wider distribution around the APSIM predicted values than the predictions of the RF emulators. The data points, however, are still most densely clustered along the one-to-one line, as Figure 4-3 shows. Again, RF emulators did a noticeably better job of predicting each of these outputs than emulators based on the other ML algorithms.

Further analysis of the least accurate ten percent of predictions for each MLE for the outputs biomass and crop yield, showed highly variable results between the three MLEs. For the ANN emulators, the least accurate predictions generally resulted in significant under-predictions of biomass and crop yield. These results were strongly associated with late maturing crops, with a mean MaturityDAS value of 172 days compared to an average for the rest of the simulations of 148 days. A likely cause of such errors is that environmental factors that caused a decrease in the above ground crop biomass and yield in the APSIM simulations occurred late in the crop lifecycle. With ML weather inputs only recording meteorological data up to 90 days after sowing, weather events or dry conditions late in the crop cycle would not have been considered by the ANN emulators. For the MARS emulators, the least accurate predictions also tended to result in under-prediction of biomass and crop yield, but these were not biased towards late maturing crops. Instead, these simulations tended to have drier soil conditions at sowing (low SowingESW) and lower solar radiation levels later in the crop's life. The RF emulators showed a very different pattern again, with the least accurate ten percent of predicted biomass and crop yield values generally being associated with over-prediction of values. For the RF emulators, the poor predictions were more strongly associated with elevated soil water at sowing (high SowingESW), higher than average rainfall beyond 60 days and lower solar radiation during the same period. Poor prediction of biomass was also associated with earlier sowing dates and small LateVegTarget parameter values. The 'black-box' nature of ML models makes detailed and accurate investigation of underlying model issues impossible.

In the case of the RF emulators, it is worth noting that these outlier values only represent between 20 and 40 data points out of a set of 26,185 data points, indicating that the visual impact of these points is overstating their importance. This is confirmed by the hexbin plot of the distribution density of the data points (Figure 4-3). One interesting aspect to note that differs between the MLEs is the generation of erroneous negative values for GrainWt. RF did not suffer from this feature, while the MARS emulators showed this feature for both the crop yield and above ground biomass predictions. One of the noted strengths of the RF algorithm is bootstrap aggregation, also known as 'bagging', which results in an ensemble of RF models. This approach has the benefits of reducing bias and variance in the resulting prediction model and producing a more representative outcome for variable data (Sexton & Laake 2009; Biau & Scornet 2016). A disadvantage of this ensemble approach used by the RF algorithm is the increased computational effort required to both develop the MLE and to run simulations compared to the ANN algorithm's approach. This pattern of fast emulators being the least accurate in both bias and error statistics calculated, as well as the accuracy of predicted target values, is observed in the data presented in Table 4-3. This is most likely a reflection of the fact that accurate predictions are more consistently produced when greater numbers of values are processed and averaged. There appears to be a generalised inverse relationship between emulator speed and accuracy of prediction.

Table 4-3. The predictive ability of the MLEs against outputs generated by the APSIM-NextGen chickpea crop model.
Statistical measures for goodness-of-fit performance analysis for seven locations used to train the machine learning emulators (MLEs). The statistics shown are, MB: mean bias reported in days or kg/ha, depending upon the output variable; MAE: mean absolute error reported in days or kg/ha, depending upon the output variable; RMSE: root mean squared error reported in days or kg/ha, depending upon the output variable; R^2: coefficient of determination; COE_{LM}: coefficient of efficiency (Legates McCabe index). The three machine learning MLEs are Artificial Neural Networks (ANN), Multivariate Adaptive Regression Spline (MARS) and Random Forest (RF). The analysis is for the predictive ability of the MLEs against outputs generated by the APSIM-NextGen chickpea crop model.

Emulator/Target	**MB**	**MAE**	**RMSE**	**R^2**	**COE_{LM}**
ANN					
EmergenceDAS (days)	0.00	0.68	0.88	0.95	0.79
FloweringDAS (days)	-0.01	1.15	1.55	0.99	0.93
PoddingDAS (days)	-0.12	5.13	7.88	0.95	0.82
MaturityDAS (days)	-0.02	3.25	4.67	0.98	0.88
Biomass (kg/ha)	-3.64	76.92	102.60	0.92	0.75
GrainWt (kg/ha)	-0.59	35.89	48.44	0.91	0.74
MARS					
EmergenceDAS (days)	0.00	0.69	0.89	0.95	0.79
FloweringDAS (days)	0.00	1.94	2.54	0.99	0.88
PoddingDAS (days)	0.02	5.93	8.88	0.93	0.79
MaturityDAS (days)	0.02	3.97	5.60	0.97	0.85
Biomass (kg/ha)	-0.94	87.62	115.29	0.90	0.72
GrainWt (kg/ha)	-0.40	40.79	54.27	0.88	0.70
RF					
EmergenceDAS (days)	0.00	0.22	0.31	0.99	0.93
FloweringDAS (days)	0.01	1.00	1.35	1.00	0.94
PoddingDAS (days)	0.05	2.51	4.37	0.98	0.91
MaturityDAS (days)	0.01	1.79	2.81	0.99	0.93
Biomass (kg/ha)	-0.26	16.18	27.70	0.99	0.95
GrainWt (kg/ha)	-0.20	12.26	20.37	0.98	0.91

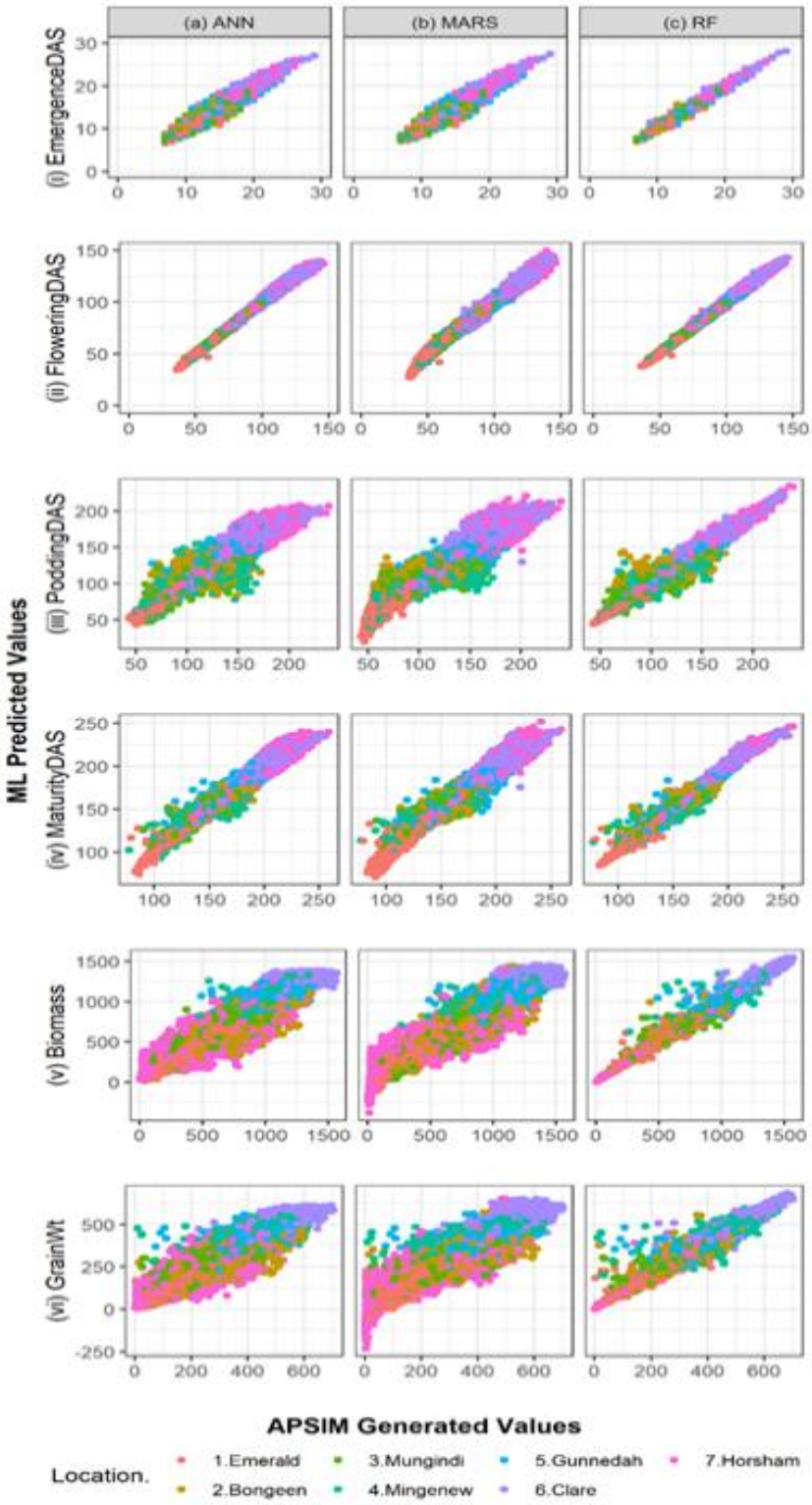


Figure 4-2. APSIM generated actual versus machine learning emulator (MLE) predicted values.
Values are for seven locations used to train the MLEs. The output values are for six APSIM-NestGen chickpea outputs; EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, above ground Biomass and GrainWt.

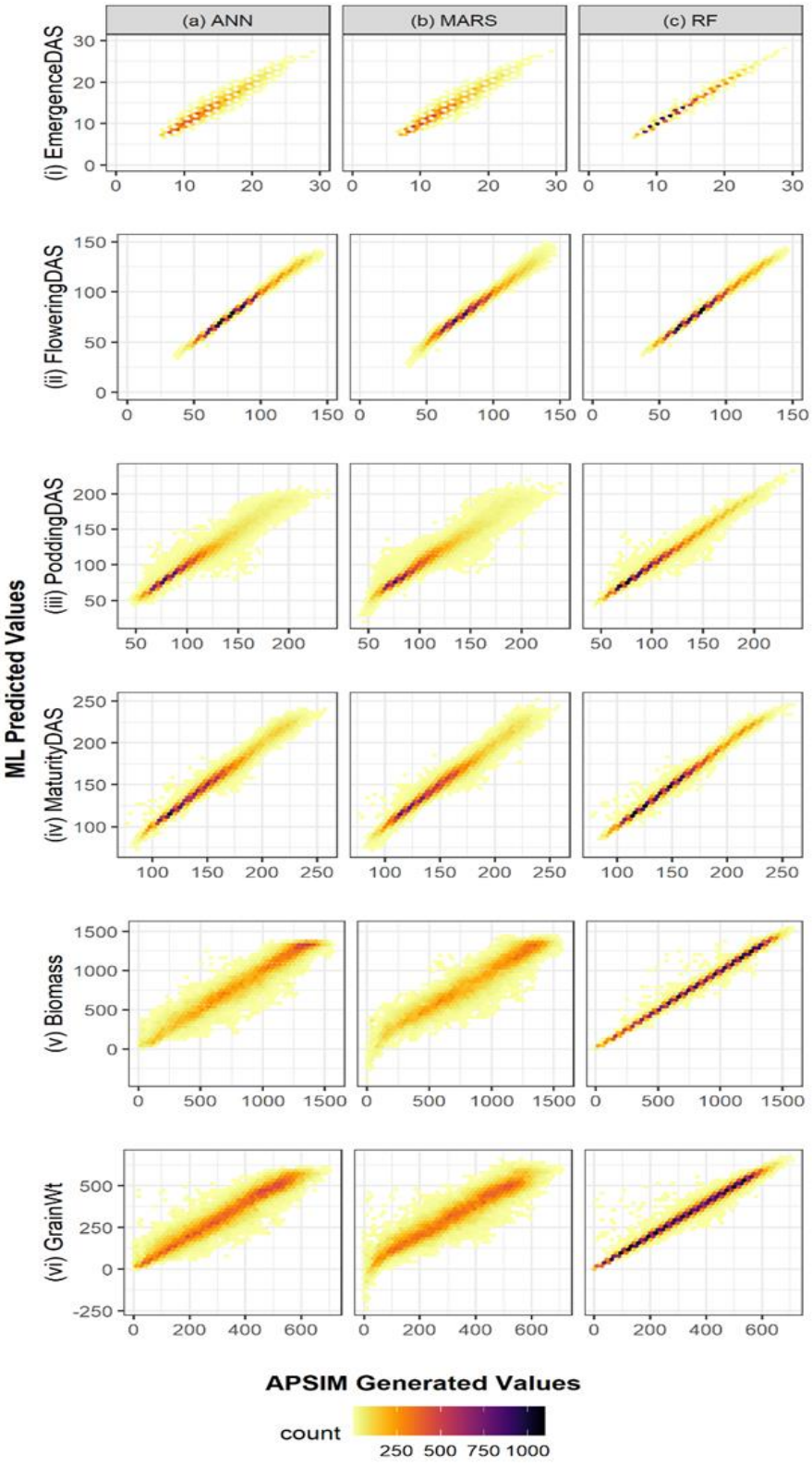
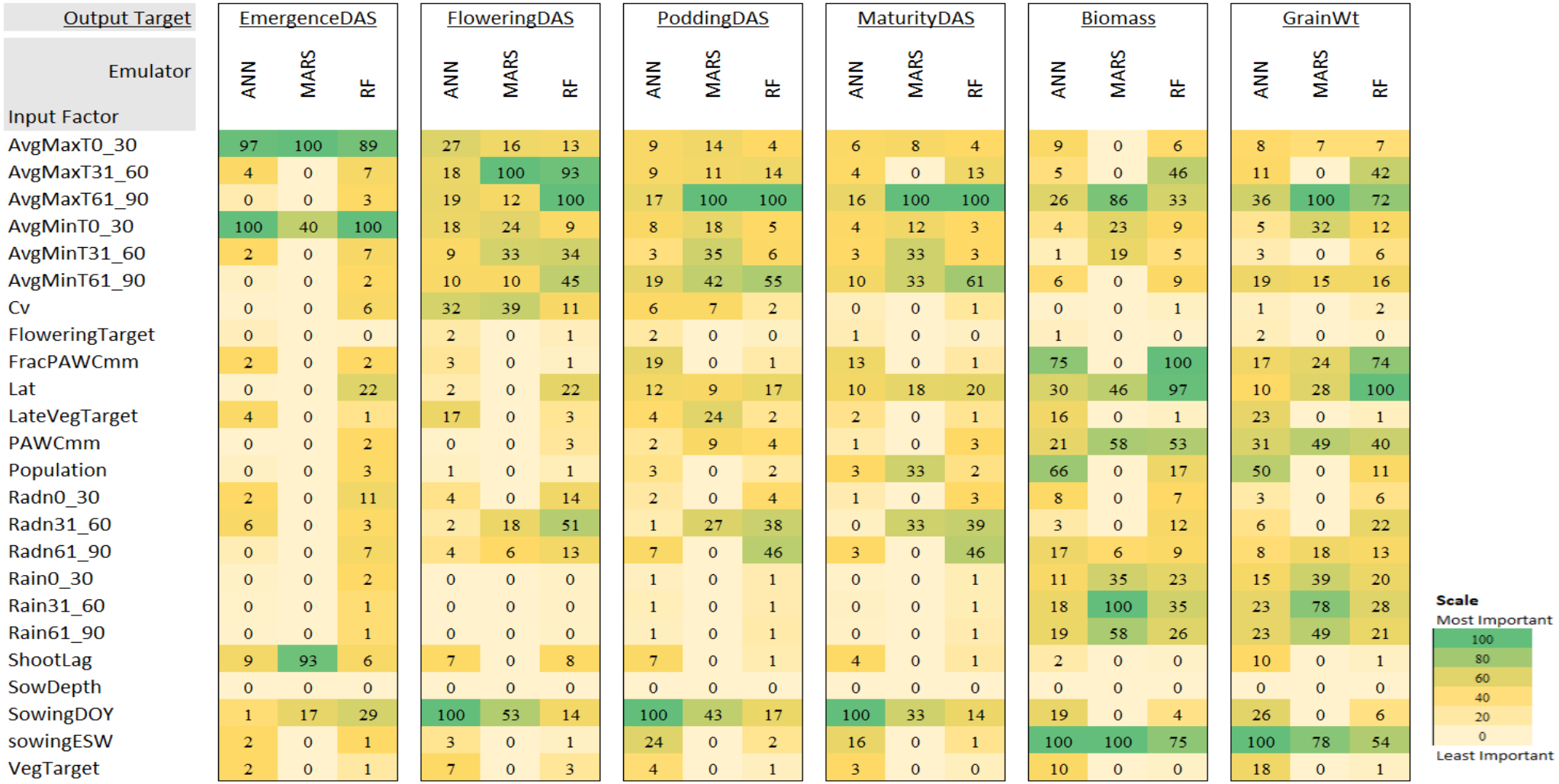


Figure 4-3. HexBin plot of the distribution density of data points for the emulator development validation data sets.
Each panel shows the summary of 26,185 data points.

4.3.1.2 Variable importance

By comparing the influence that the input factors have on the outputs across each of the MLEs, patterns and variations can be observed in what is driving each emulator. Figure 4-4 highlights the patterns of the index values. For EmergenceDAS, all three MLEs were strongly influenced by the maximum and minimum temperatures during the first 30 days after sowing. This is expected as emergence is primarily a temperature driven response in the chickpea model, and it occurs in the first 30 days of the crop simulation. Responses should not be driven by data after this time period. The MARS algorithm was the only algorithm tested to detect the ShootLag, which differed between genotypes, as a driving input parameter, rating it at an index value of 93. Interestingly the R^2 and COE_{LM} values for the EmergenceDAS predictions for the ANN and MARS emulators were consistent values of 0.95 and 0.79, respectively. Other output targets showed greater diversity in the input variables identified as most important. For the ANN emulators, the input SowingDOY was very significant for predicting the output target FloweringDAS, while the MARS and RF emulators rated average maximum temperatures between 31 and 60 days after sowing as highly influential. PoddingDAS and MaturityDAS showed something of a consist pattern between MLEs, with SowingDOY being most important for the ANN emulator, while average maximum temperature for 61 to 90 DAS was the most significant for the MARS and RF emulators. The RF emulator was the only one to have an additional value over 50, that of AvgMinT61_90.

The patterns of rating significance for both biomass and crop yield were similar in each of the MLEs. Above ground biomass and crop yield were both strongly influenced by SowingESW by all ML algorithms, although RF emulators used the closely correlated FracPAWCmm input instead. Only the RF emulator rated the latitude (Lat) variable as a significantly important input, which it did for both above ground biomass and crop yield. Both the MARS and the RF emulators used the AvgMaxT61_90 for crop yield prediction, while no temperature, rainfall or radiation inputs were rated above an importance of 36 by the ANN emulator for crop yield.



Output Target	EmergenceDAS			FloweringDAS			PoddingDAS			MaturityDAS			Biomass			GrainWt		
Emulator	ANN	MARS	RF	ANN	MARS	RF	ANN	MARS	RF	ANN	MARS	RF	ANN	MARS	RF	ANN	MARS	RF
Input Factor																		
AvgMaxT0_30	97	100	89	27	16	13	9	14	4	6	8	4	9	0	6	8	7	7
AvgMaxT31_60	4	0	7	18	100	93	9	11	14	4	0	13	5	0	46	11	0	42
AvgMaxT61_90	0	0	3	19	12	100	17	100	100	16	100	100	26	86	33	36	100	72
AvgMinT0_30	100	40	100	18	24	9	8	18	5	4	12	3	4	23	9	5	32	12
AvgMinT31_60	2	0	7	9	33	34	3	35	6	3	33	3	1	19	5	3	0	6
AvgMinT61_90	0	0	2	10	10	45	19	42	55	10	33	61	6	0	9	19	15	16
Cv	0	0	6	32	39	11	6	7	2	0	0	1	0	0	1	1	0	2
FloweringTarget	0	0	0	2	0	1	2	0	0	1	0	0	1	0	0	2	0	0
FracPAWCmm	2	0	2	3	0	1	19	0	1	13	0	1	75	0	100	17	24	74
Lat	0	0	22	2	0	22	12	9	17	10	18	20	30	46	97	10	28	100
LateVegTarget	4	0	1	17	0	3	4	24	2	2	0	1	16	0	1	23	0	1
PAWCmm	0	0	2	0	0	3	2	9	4	1	0	3	21	58	53	31	49	40
Population	0	0	3	1	0	1	3	0	2	3	33	2	66	0	17	50	0	11
Radn0_30	2	0	11	4	0	14	2	0	4	1	0	3	8	0	7	3	0	6
Radn31_60	6	0	3	2	18	51	1	27	38	0	33	39	3	0	12	6	0	22
Radn61_90	0	0	7	4	6	13	7	0	46	3	0	46	17	6	9	8	18	13
Rain0_30	0	0	2	0	0	0	1	0	1	0	0	1	11	35	23	15	39	20
Rain31_60	0	0	1	0	0	0	1	0	1	0	0	1	18	100	35	23	78	28
Rain61_90	0	0	1	0	0	0	1	0	1	0	0	1	19	58	26	23	49	21
ShootLag	9	93	6	7	0	8	7	0	1	4	0	1	2	0	0	10	0	1
SowDepth	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SowingDOY	1	17	29	100	53	14	100	43	17	100	33	14	19	0	4	26	0	6
sowingESW	2	0	1	3	0	1	24	0	2	16	0	1	100	100	75	100	78	54
VegTarget	2	0	1	7	0	3	4	0	1	3	0	0	10	0	0	18	0	1

Figure 4-4. Heat maps of input variable importance for three MLEs.
Results are for six output parameters. Importance indices are rated from zero (0) for no effect on the output value, to 100 being the input with the most significant effect on the output values. Index values are relative to the most significant input rated at 100.

4.3.1.3 Computational requirements

The time taken to train the MLEs is an indicator of the computational costs associated with developing each emulator system Table 4-4 shows that there was a great spread in the computational requirements needed to develop each type of emulator. Times ranged from 12.1 seconds for the MARS algorithm to develop a predictive emulator for the output EmergenceDAS, to a high of 17,644.8 seconds (4hrs 54mins) for the RF algorithm to produce a predictive emulator for the same output. On average, MARS emulators were developed with least computational effort, ANN emulators were almost three times more costly, and RF emulators were approximately 500 times more costly based on the performance of the code libraries and computing environment used for this study.

Table 4-4. Time (in seconds) taken to train each MLE.
Training data sets used 26,185 data points for each target output. The times are representative only and were obtained from developing the MLEs in an R environment on an Intel core-*i7* laptop computer.

Output	**ANN**	**MARS**	**RF**
EmergenceDAS	77.8	12.1	17644.8
FloweringDAS	77.9	37.4	10930.8
PoddingDAS	86.5	34.8	16149.9
MaturityDAS	86.3	31.5	17191.2
Biomass	76.5	35.0	13544.1
GrainWt	77.5	34.9	14530.5
Average:	80.4	30.9	14998.6
(all times in seconds)			

4.3.2 Performance at test locations

The MLEs developed using data from seven locations within the Australian chickpea production regions, were tested using data from two additional locations, also within the same production regions. Scatter plots of the APSIM generated values plotted against the values generated by the predictive MLEs are shown in Figure 4-5, with the statistical analyses of the 'goodness-of-fit' of the data values provided in Table 4-5. For predictions of EmergenceDAS and FloweringDAS, the three ML algorithms,

ANN, MARS and RF, all performed well with consistent R^2 values of 0.91 for EmergenceDAS and of 0.98 for FloweringDAS. The corresponding values of COE_{LM} ranged between 0.72 and 0.73 for EmergenceDAS and between 0.86 and 0.88 for FloweringDAS. Values for each test location were equally well predicted. For the three ML emulators, ANN, MARS and RF, the prediction of MaturityDAS was the next most accurate output with R^2 values of 0.95 and 0.96 and COE_{LM} values ranging from 0.72 to 0.82. This was followed by the predictions for PoddingDAS with R^2 values ranging from 0.87 to 0.90 and COE_{LM} values ranging from 0.60 to 0.72.

All three ML approaches failed to accurately predict above ground biomass and crop yield. The MLEs were incapable of making accurate predictions for the test locations based on the data from the training locations. Given that biomass and crop yield were both strongly influenced by soil water holding capacity and soil water content at sowing (Figure 4-5), it is most likely that insufficient soil types and soil water conditions were included in the original data set to allow the test locations to be accurately modelled. The test locations effectively fell outside the parameter value ranges and effects observed at the training locations and so predicted values were nonsense.

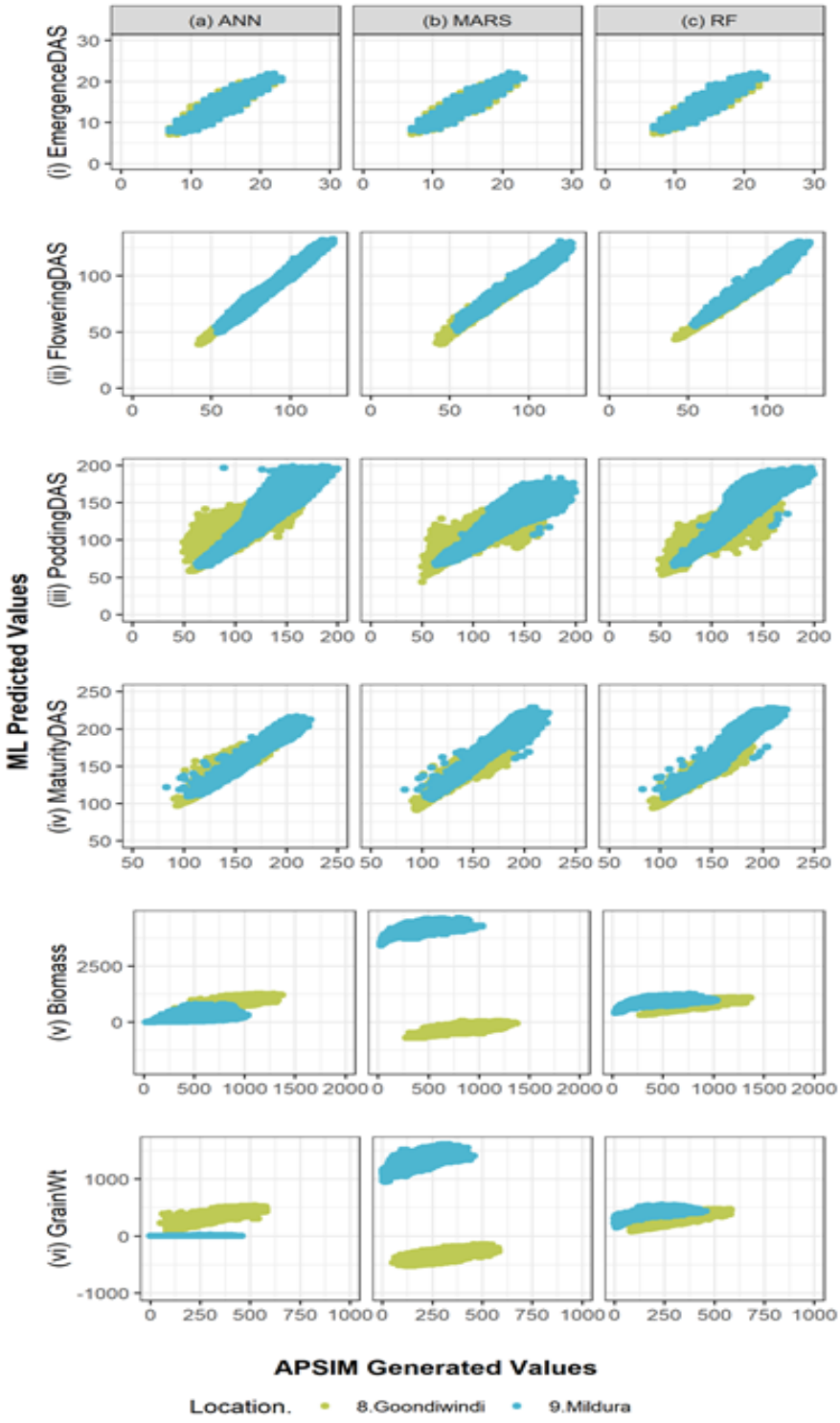


Figure 4-5. APSIM generated actual versus ML predicted values for two test locations.
Test locations were not used to train the MLEs. Target values are for six APSIM-NestGen chickpea outputs; EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, above ground Biomass and GrainWt.

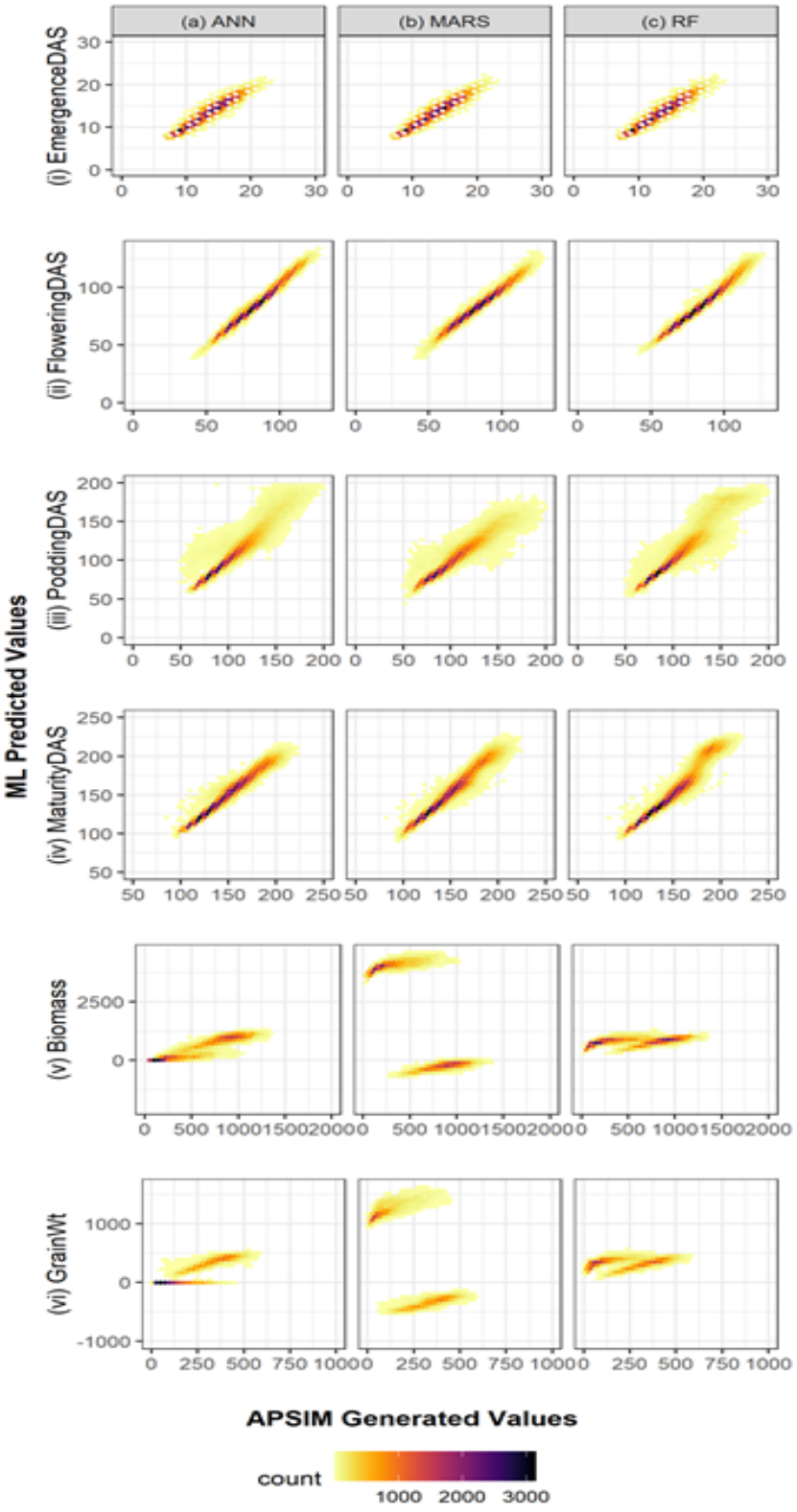


Figure 4-6. HexBin plot of the distribution density of data points for the test location data sets.
Each panel shows the summary of 37,395 data points.

Table 4-5. The predictive ability of the MLEs for two test locations against outputs generated by the APSIM-NextGen chickpea crop model.
Statistical measures for goodness-of-fit performance analysis for two test locations. The statistics shown are, mean bias (MB) reported in days or kg/ha, depending upon the output variable, mean absolute error (MAE) reported in days or kg/ha, depending upon the output variable, root mean squared error (RMSE) reported in days or kg/ha, depending upon the output variable, coefficient of determination (R^2), coefficient of efficiency (Legates McCabe index) (COE_{LM}). The three machine learning emulators (MLEs) are Artificial Neural Net (ANN), Multivariate Adaptive Regression Spline (MARS) and Random Forest (RF).

Emulator/Target	MB	MAE	RMSE	R^2	COE_{LM}
ANN					
EmergenceDAS (days)	0.07	0.74	0.95	0.91	0.72
FloweringDAS (days)	0.61	1.58	2.10	0.98	0.88
PoddingDAS (days)	3.32	8.97	13.18	0.87	0.60
MaturityDAS (days)	2.04	3.84	5.42	0.96	0.82
Biomass (kg/ha)	-136.26	193.16	260.08	0.76	0.35
GrainWt (kg/ha)	-21.50	74.85	95.84	0.79	0.39
MARS					
EmergenceDAS (days)	-0.08	0.75	0.95	0.91	0.72
FloweringDAS (days)	-0.04	1.77	2.25	0.98	0.86
PoddingDAS (days)	-1.96	6.26	9.17	0.89	0.72
MaturityDAS (days)	3.68	5.41	7.32	0.95	0.75
Biomass (kg/ha)	1315.60	2437.08	2772.21	0.62	-7.25
GrainWt (kg/ha)	227.21	901.09	931.87	0.48	-6.28
RF					
EmergenceDAS (days)	0.02	0.73	0.93	0.91	0.73
FloweringDAS (days)	0.70	1.74	2.47	0.98	0.87
PoddingDAS (days)	4.40	7.23	11.43	0.90	0.68
MaturityDAS (days)	3.88	6.12	9.18	0.95	0.72
Biomass (kg/ha)	224.23	290.65	374.34	0.20	0.02
GrainWt (kg/ha)	105.06	134.92	173.40	0.10	-0.09

4.4 Discussion

4.4.1 Performance with training data set

The results of this study have shown that MLEs can be developed that are able to replace the APSIM system model for the set of input parameter value ranges used for testing. They show that all three ML approaches tested are capable of being used to

generate predictive regression MLEs for the crop model outputs tested. The FloweringDAS prediction was the most accurate output for each of the MLEs, indicating that the input factors included did cover all the important driving variables for this output. It is revealing that the importance of the input variables (Figure 4-5, panel 2) was not consistent between the different algorithms. For FloweringDAS, the ANN emulator was heavily reliant upon the time of sowing, with no other input coming close to having as significant an impact. The MARS emulator relied almost entirely on mid-season maximum temperatures, with its next most important input, time of sowing, rated as only half as important. The RF emulator was most strongly influenced by mid and late-season maximum temperature. This shows clearly that great care must be taken if interpreting the input importance values for MLEs as being an accurate predictor of the importance of input factors for an underlying model. Different algorithms can, and do, predict the correct answer in the majority of instances, using significantly different importance weightings of input values. Boehmke and Greenwell (2019) have previously warned that algorithms, like that used in the MARS approach, can give misleading results for variable importance where there are closely correlated input factors. This is due to the algorithms approach of selecting input factors based on their contribution to an output value and discarding additional inputs if they do not improve the prediction by some given marginal amount. This can result in only one of a closely correlated set of inputs being used to predict output values, with the other inputs, although equally as influential on the output, rated as not used or of low importance. Breiman (2001) and Dumancas and Bello (2015) indicate that the RF algorithm is well suited to cope with multicollinearity of inputs, and so is not subject to this limitation to the same degree as the MARS algorithm. For neural networks, which is represented by the ANN algorithm, the robustness and accuracy of their predictions have been found to be adversely affected by collinearity between input factors (Dumancas & Bello 2015; Samarasinghe 2016). These authors advise that feature selection needs to be undertaken in order to remove non-influential inputs and inputs that exhibit collinearity from the data set, before reliable neural net models can be built. For the purpose of comparing the ML algorithms based on a consistent approach, this step was not undertaken in this study.

The greatest differences between the accuracy of predictions of the MLEs was for the outputs of above ground biomass and grain weight (yield). These two outputs are the

ones in the output set most influenced by a wide range of crop, environmental and management factors, and represent the sum of everything the crop has experienced. They are key outputs for most crop models (Stöckle et al. 1994; Jones et al. 2003; Keating et al. 2003; Stöckle et al. 2003). For these two outputs, the RF emulator was clearly a superior predictor than the emulators produced by the other three ML algorithms. The reasons for this difference in accuracy are not easily determined. Contributing factors are likely to include the inherent suitability of the underlying ML algorithm for the data being analysed, and the extent to which the data set has been optimised for the ML approach. One factor that was identified during analysis of this data was that the summary climate details were only recorded until 90 days after sowing, while many of the crops with the poorest predictions of biomass and yield reached maturity (as shown by harvest date) well beyond this cut-off. It is probable that adverse weather conditions during the final stages of crop growth and crop maturation resulted in unpredictable crop vigour and yield loss. Extended periods of weather details in the input parameters may have aided in more accurate predictions of biomass and yield. While feature selection and dimensionality reduction steps are warranted for the neural net based algorithms (Samarasinghe 2016), the purpose of this study was to compare the performance of the core approaches. The investigation of optimal feature selection algorithms would constitute a research study in its own right. It is worth noting that, under the constructs of this study, where the outputs of the simulation model are being predicted rather than real world observations, all potential input factors for the MLEs are known, albeit a very large number of them. This makes the possibility of identifying a complete set of driving input factors a feasible objective.

Based on the accuracy of predicted values, the RF algorithm is the best of the three algorithms tested. The accuracy of predicted output values produced by the RF emulators for the locations on which it was trained are good, with the lowest accuracy being for both PoddingDAS and GrainWt at $R^2 = 0.98$ and $COE_{LM} = 0.91$. With this level of accuracy, the RF emulators could be used to predict with a high degree of confidence, any of the six model outputs for any of the seven training locations for input values within the range of values observed in the training set. The design of this experiment meant that one set of input factors was tested for their ability to be used to predict each of the six outputs. With careful review and iterative

testing, it should be possible to improve the predictive accuracy for any chosen output.

The computational costs involved in developing, or training, the MLEs (Table 4-4) varied widely between the different algorithms. A key feature of this study that differs from many studies into the development of ML models or MLEs, is the expected use and life timeframe of the emulator. To be useful as a tool to run SA on a systems model, a ML emulator needs to be able to be rapidly developed, used, and discarded rather than having an iterative development and retention lifecycle. This is because each SA will be based on a different scenario and designed to test different input parameters or different ranges for input parameter values each time they are run. As the MLEs are generated for specific sets of inputs and can only be used to predict outputs for input settings within the value ranges with which they were developed, reuse of MLEs is limited. Even where MLEs can be reused, great care would be required to ensure the value ranges of all input parameters were within the development limits of the MLE, and the mix of those inputs was of a pattern that was not dissimilar to patterns used to develop the MLEs. While broadly applicable MLEs might be possible to produce, a narrowly applicable MLE developed for a specific application is a safer option if unpredictable outcomes are to be avoided. Comparisons of development times of ML models are not readily available in the literature. In this study the MARS algorithm was, on average, almost 500 times faster to train than the RF algorithm, with the ANN algorithm being approximately 200 times faster than the RF algorithm. It must be noted that this represents just one snapshot of specific implementations of three algorithms out of potentially dozens of alternative algorithms. The code used to develop the algorithm solution, the computing environment utilised to run the code and the computing hardware that the ML was run on, all have the potential to significantly affect the outcomes of such a comparison. Advances in, or reimplementation of, any of these factors, or the selection of alternative algorithms or environments, will have effects on the outcomes. For this study, the outcome is clear; the RF algorithm was the most accurate of the ML approaches, but it came at a significant computational cost. The superior results from the RF emulator are in contrast to Kouadio et al. (2018) who found an extreme learning machine, which is an advanced form of ANN algorithm, superior at forecasting coffee yield. Obsie et al. (2020) reported an extreme gradient

boosting model produced better results than a RF model for blueberry yield prediction, although both the gradient boosting model and the RF model performed better than a multi linear regression approach. Other researchers (Jeong et al. 2016; Dayal et al. 2019; Feng et al. 2019; Lawes et al. 2019; Feng et al. 2020) have chosen RF models as their preferred ML approach in studies predicting crop growth.

4.4.2 Performance with test locations

A second part of this study independently assessed the robustness of the MLE solutions, and what potential they might have for replacing the APSIM-NextGen modelling system for the specific prediction task for which they were developed. The three ML algorithms were not as accurate in predicting the chronological development of the crop, that is the EmergenceDAS, FloweringDAS, PoddingDAS and MaturityDAS, as for predicting the training data set, but predictions were not unrealistic for the ANN, MARS and RF emulators, as shown in Figure 4-5 and associated statistical values in Table 4-5. This fulfills the second aspect of the objectives for this chapter, that the MLEs, if developed with sufficiently diverse data sets, are robust enough to predict outputs for any location in the production region, regardless of whether that location was used in the training data set or not. The FloweringDAS predictions, with R^2 values 0.98 and COE_{LM} values ranging from 0.86 to 0.88 for each of the algorithms, were the most accurate of the predictions for the test locations. The other statistical measures generated to test the accuracy of the emulators, MB, MAE and RMSE, all followed the same relative patterns of which was the most to least accurate emulator, with RF being the most accurate, ANN being next, and MARS being the least accurate. With this level of accuracy, the use of any of these three MLEs to predict flowering date as days after sowing, for any location within the Australian chickpea production regions, would be justifiable.

By using test locations, most of the input factors used to train the MLEs were able to be controlled and ensure that they fell within the ranges used to develop the MLEs. Factors that were not controlled and had the potential to fall outside the development set boundaries were related to the soil, specifically the water holding capacity of the soil and starting soil moisture levels. The predictions for above ground biomass and grain weight (Figure 4-5 and Table 4-5) are shown to be erroneous for all three ML

algorithms. As noted previously, these outputs reflect the sum of all the factors that influence crop growth. Consequently, their predicted values are most likely to reveal any weakness in the robustness of the MLEs. Even though the management and genomic factors were consistent with the training data, the test locations introduced different soils to the simulations. For example, the predictions of biomass and yield for Mildura were the least accurate and most varied between the different MLEs. Mildura soil was the only sandy loam in the data set and had the lowest water holding capacity of any of the soils. This soil was the most in contrast to the other soils, and the emulators performed most poorly with it. This is consistent with the findings of Shahhosseini et al. (2021) who identified soil water parameters as key drivers of ML models used to predict corn yields. Some of the patterns that define the relationships between input factors and output values observed at the test locations in my study were not present in the training data, so none of the ML algorithms could predict them for the new locations. This clearly stands as a warning about the potential use of MLEs in replacing process driven models for generating predicted outputs. All patterns of input factors affecting output values must be included in the training data to develop an ML emulator that is capable of robustly predicting outputs. Other recent research integrating process-driven models with ML has focused on the effects of climate change on crop yields (Feng et al. 2019; Leng & Hall 2020). Both studies have reported significant benefits in integrating the two modelling approaches but have not highlighted the dangers and limitations of supplying incomplete data sets to the ML models during development. In my study, the training data included all required patterns for predicting FloweringDAS but lacked details which determine above ground biomass and grain weight. As a result, the FloweringDAS predictors are more robust than the above ground biomass and yield predictors.

4.5 Conclusion

The results from this chapter have answered the second of the research questions presented in Chapter 2 of this thesis, that being: Which ML algorithms produce the most accurate emulators, and at what computational cost? And what are the advantages and disadvantages of each algorithm for producing an emulator? This

study has shown that emulators of crop models, built on ML algorithms, can be developed to predict a range of simulated crop outputs. The accuracy of predictions varies based the algorithm used and the output being predicted, with the RF emulator being the most consistently accurate emulator used in this study. Computational costs, measured as the time taken to train the MLEs, also varied by algorithm. The MARS emulators were the fastest emulators to be trained in this study, with the RF emulators having the longest training times. These findings will have implications for the choice of algorithm if this approach of utilising MLEs were to be used to improve the time efficiency of running very large numbers of model simulations. Additionally, the robustness of the emulator needs to be tested for each output variable. There is no set of input factors that will be suitable for all outputs in all situations. It is, however, reasonable to assume that it is possible to develop accurate predictive MLEs for any output as all input factors for the process driven simulation model are known, so it should be possible to generate training data sets with all input factors required for the prediction of the target output. A potential disadvantage of the MARS algorithm is that it discards input parameters if they are found to be unimportant during its development. This could limit its usefulness as a tool for SA as parameters of low importance within one scope may become more important if the scope is altered by fixing some of the more influential parameters. This is exactly the situation encountered during the design phase of the experiment for Chapter 5 of this thesis. As a result, the MARS emulator was assessed as not being suitable for undertaking SA in the next stage of this thesis, and only the ANN and RF emulators were utilised in Chapter 5.

CHAPTER 5: PAPER 3 - Substituting process driven biophysical models with machine learning based emulators for undertaking sensitivity analysis

Preamble

In the previous chapter, Chapter 4, of this research thesis, emulators of the APSIM-NextGen chickpea model were developed using machine learning algorithms for multivariate adaptive regression splines (MARS), artificial neural networks (ANN) and random forests (RF). These emulators were shown to be highly effective in their predictive capacity for seven model output parameters, EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainYield. The accuracy of the predictions did vary between the different ML algorithms, with the RF being shown to be the most consistently accurate across each of the target outputs.

In Chapter 5 of this research, the machine learning based emulators (MLEs) will be used to undertake Morris and Sobol analyses. With the different approach to researching the possibility of using MLEs in the SA of process-driven models that was introduced between Chapter 3 and Chapter 4 of this thesis, the results presented in this chapter, Chapter 5, are not comparable with the results of Chapter 3. In Chapter 3, the parameter importance indices of the MARS model were compared with the SA indices produced by Morris and Sobol analyses undertaken on the input-output sensitivities of the process-driven model. As the MARS algorithm is designed to eliminate input parameters which contribute little towards the output values, it has been dropped from this experiment because the results of a Morris or Sobol analysis without a consistent set of selected input parameters would be meaningless. In Chapter 5, ANN and RF emulators of the process-driven model and previously generated from the outputs of APSIM-NextGen runs, will be used to run the simulations required for Morris and Sobol analyses. These analyses will assess the MLEs input-output sensitivities. The aim of this experiment is to test if the computational efficiency of the MLEs can be harnessed by undertaking SA using the MLEs, with the results reflecting the underlying relationships expressed by the process-driven models.

5.1 Introduction

Sensitivity analysis (SA) is a recommended part of most research involving computer modelling (Plischke et al. 2013; Pianosi et al. 2016; Janssen et al. 2017). Global SA, where the sensitivity of outputs is assessed by varying inputs over the entire input space, enables analysts to gain insights related to input-output mappings, including key drivers of output values and of model uncertainty (Saltelli et al. 2000). Knowledge of these key drivers can focus future model development efforts, enhance appropriate model application, and guide the allocation of resources for more effective and efficient data collection (Plischke et al. 2013). Undertaking global SA on complex models with large numbers of model input factors, however, will present challenging levels of computational burden.

Significant research has been undertaken to address the problem of high computational burden for undertaking SA, as well as for efficiently producing predicted outcomes for large numbers of model evaluations when simulation experiments are undertaken. A range of approaches have been detailed in literature. These include, but are not limited to, the use of high performance computers to undertake model runs (Thorp et al. 2020), state dependent parameter metamodeling (Ratto et al. 2007), global evolutionary algorithms (Shin et al. 2015), polynomial chaos expansion (Shin et al. 2015; Liu & Choe 2021), multiple linear regression algorithms (Friedman 1991b; Friedman & Roosen 1995) and ML algorithms such as support vector machines (Raghavendra. N & Deka 2014). Many of these approaches rely on reducing the dimensionality of the complex model by producing a meta-model, or emulator, which has lower complexity and faster execution time than the original model. Chapter 3 of this thesis developed a method to show that the variable importance indices of a MARS based machine learning emulator (MLE) are comparable with, but not fully substitutable for, the Morris and the Sobol statistics generated from the underlying process-driven model.

Emulators based on ML approaches of artificial neural networks and random forest algorithms have been shown to be effective for generating emulators capable of

predicting outputs of biophysical crop models, as developed and demonstrated in Chapter 4 of this research thesis. While these MLEs have been shown to predict a range of APSIM-NextGen chickpea model outputs with a very high degree of accuracy, it has not been investigated as to whether they can be used in the SA of the process-driven APSIM models by generating the very large number of outputs required for SA. No literature is available which addresses the question of whether MLEs are useful for the SA of biophysical models.

In the current chapter of this study, MLEs were used to conduct a Sobol first-order and total-effects SA, and a Morris method SA. The MLEs were developed to emulate six outputs from the APSIM-NextGen chickpea model using two different classes of ML algorithm, artificial neural networks (ANN) and random forest (RF). Examples of ANN models have been used to predict outputs, such as yield, from biological and environmental systems (Shastry et al. 2016; Ghimire et al. 2018; Sanikhani et al. 2018; Nettleton et al. 2019; Shahhosseini et al. 2021). They are typical of the 'black box' style of MLE, where models are trained on input data and automatically self-calibrate to classify or predict output values, the internals of the MLE generally not being able to be observed by a user of the system. RF models use ensembles of decision trees for classification tasks or prediction of outcomes. They are one of the most widely used forms of ML frameworks for both classification and regression (Cravero & Sepúlveda 2021). There are many examples in agricultural literature of RF models being the ML model of choice and have been used for soil models (Gebauer et al. 2019; Hussein et al. 2020), yield forecasting (Kouadio et al. 2018; Feng et al. 2019; Feng et al. 2020; Obsie et al. 2020; Guo et al. 2021) and analysis of remote sensing (Belgiu & Drăguţ 2016; Dahms et al. 2016). Two analyses were undertaken to assess the computational efficiency of the MLEs: one SA involved all 22 inputs factors, while the second analysis focussed on just six phenology critical inputs while holding other input factors constant. The analyses of 22 input factors required 2.4 million model evaluations for each of six model outputs for each of the two classes of MLE. Along with execution times, comparisons were made between the outcomes of the analyses for the two classes of emulator. The objective of this research study is to demonstrate the effectiveness of undertaking SA using the Morris and Sobol methods by using an MLE to run simulations rather than the process-driven model itself. This approach has the potential to greatly reduce the

computational burden of running the very large numbers of simulations required for SA involving more than twenty input parameters. This research aims to provide guidance to researchers and developers of biophysical models as to the suitability and practicality of using MLEs to undertake SA of an underlying process driven model.

5.2 Methods

Sensitivity analysis was undertaken on MLEs developed in Chapter 4 of this thesis using two algorithmic approaches, ANN and RF. The MLEs were developed as emulators of the APSIM-NextGen chickpea model for the prediction of six model outputs: 1) days from sowing to emergence (EmergenceDAS), 2) days from sowing to flowering (FloweringDAS), 3) days from sowing to first fruiting pod (PoddingDAS), 4) days from sowing to crop maturity (MaturityDAS), 5) above ground crop biomass at harvest (kg/ha) (Biomass), and 6) weight of harvested grain (kg/ha) (GrainWt). The SA involved an elementary effects analysis using the Morris (Morris 1991) method, further enhanced by Campolongo et al. (2007) and a variance decomposition analysis using the Sobol (Sobol' 2001) method.

5.2.1 Computing environment

All simulations and data analyses were performed on an Intel Core-i9 11900H CPU 2.5 GHz based computer with 32 GB RAM running Microsoft Windows 10 operating system. The MLEs, developed in the Chapter 4 of this research thesis, were run in R (version 4.1.0 (2021-05-18) (R Core Team 2021) in Microsoft Windows under RStudio 1.4.1717 (RStudio Team 2021).

5.2.2 Simulation configuration

Two MLEs have been used to undertake the simulations required to undertake both the Morris and Sobol SA. In these analyses, the APSIM-NextGen chickpea model has not been used, other than to generate the data that was used to develop the MLEs. The MLEs were developed and trained as part of the research undertaken in Chapter

4 of this thesis, on data sets generated by running APSIM-NextGen chickpea model to simulate crops for seven locations throughout the chickpea growing regions in Australia for 120 years (1900 to 2019). The locations ranged from a northern-most latitude of -23.569 (location 1, Figure 5-1) to a southern-most latitude of -36.670 (location 3, Figure 5-1), with the approximate centre point being Gunnedah, NSW (location 2, Figure 5-1) at a latitude of -30.954. To perform either a Morris or a Sobol SA, input parameters which are not of interest for the analysis must be fixed to set values, while the values of input parameters being analysed are varied over known controlled ranges, as specified below in Table 5-1. These value ranges were within the ranges used for the development of the MLEs in Chapter 4. Input factors were fixed to set values by selecting individual simulations from the original APSIM-NextGen generated data set. These simulations were for sowings at specific locations and formed a set of 'base' simulations upon which the SA was subsequently undertaken. The simulations selected were for sowings on day-of-year 100, 150 and 200 at locations Emerald (location 1, Figure 5-1), Gunnedah (location 2, Figure 5-1) and Horsham (location 3, Figure 5-1). The selected base-simulations each demonstrated high agreement with the APSIM-NextGen generated values for each of the six output targets. Where all 22 input parameters for the MLEs were varied, the base-simulation chosen was sowing two (day-of-year 150) at Gunnedah, as this was the mid-point time and central most location around which to vary all the parameters. Values of the 22 input factors of the MLEs were varied within the ranges used to train the emulators. The name, description and minimum and maximum of the value ranges used for SA are listed in Table 5-1.

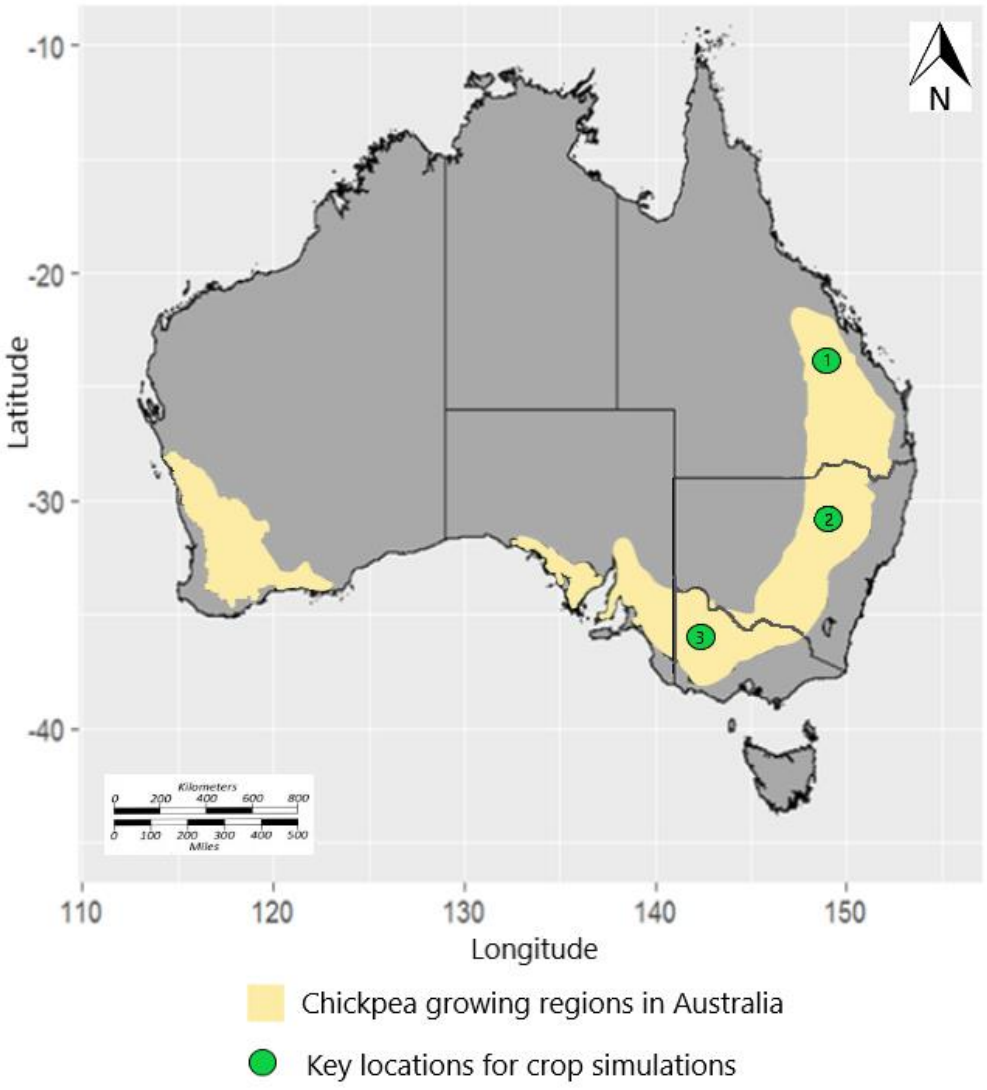


Figure 5-1. Chickpea growing regions in Australia showing the northernmost, southernmost, and central locations for crop simulations.
The three locations indicated (green dots) are: 1) Emerald, Qld, the northern most latitude simulated, 2) Gunnedah, NSW, the central location used as the base scenario and 3) Horsham, Vic., the southernmost latitude simulated.

Table 5-1. Machine learning input factors used for the sensitivity analysis of the ML emulators.
The same inputs and range of input values were used for the sensitivity analysis of each of the six output targets.

Input Factor Name	Description	Minimum Value	Q1	Q2	Q3	Maximum Value	Mean
AvgMaxT0_30	Average maximum temperature for 0 to 30 days after sowing	10	16	22	27	33	22
AvgMaxT31_60	Average maximum temperature for 31 to 60 days after sowing	12	17	23	28	33	23
AvgMaxT61_90	Average maximum temperature for 61 to 90 days after sowing	11	17	24	30	36	24
AvgMinT0_30	Average minimum temperature for 0 to 30 days after sowing	0	4	8	12	16	8
AvgMinT31_60	Average minimum temperature for 31 to 60 days after sowing	0	4	7	11	14	7
AvgMinT61_90	Average minimum temperature for 61 to 90 days after sowing	0	5	9	14	18	9
FloweringTarget	Phenological parameter. Differs between genotypes	100	125	150	175	200	150
FracPAWCmm	Amount of soil water present at sowing. As a fraction of PAWC	0.2	0	1	1	1.0	1
Lat	Latitude of the sowing location	-36.0	-33	-30	-27	-24.0	-30
LateVegTarget	Phenological parameter. Differs between genotypes	0	63	125	188	250	125
PAWCmm	Soil's plant available water capacity to 1.5m depth (mm)	260	280	300	320	340	300
Population	Sown plant population in plants /m^2	30	33	35	38	40	35
Radn0_30	Sum of solar radiation for 0 to 30 days after sowing	150	270	390	510	630	390
Radn31_60	Sum of solar radiation for 31 to 60 days after sowing	150	298	445	593	740	445
Radn61_90	Sum of solar radiation for 61 to 90 days after sowing	170	328	485	643	800	485
Rain0_30	Sum of rainfall for 0 to 30 days after sowing	0	74	148	221	295	148
Rain31_60	Sum of rainfall for 31 to 60 days after sowing	0	95	190	285	380	190
Rain61_90	Sum of rainfall for 61 to 90 days after sowing	0	95	190	285	380	190
ShootLag	Phenological parameter. Differs between genotypes	120	125	130	135	140	130
SowingDOY	Sowing date as Day Of Year	90	118	145	173	200	145
SowingESW	Extractable soil water at sowing	60	125	190	255	320	190
VegTarget	Phenological parameter in thermal time Differs between genotypes.	400	450	500	550	600	500

5.2.3 Machine learning emulators

Two of the classes of MLEs developed in Chapter 4: ANN models and RF models, were assessed as being suitable to be appraised for their usefulness in SA of the MLEs. Each class of MLE had emulators for six outputs from the APSIM-NextGen chickpea model. An installation of R (version 4.1.0 (2021-05-18) (R Core Team 2021) in Microsoft Windows was used to run the emulators. The R environment was also used for data preparation and manipulation, reporting and graphics generation, with the packages *ggplot2* (version 3.3.3) (Wickham 2016) and other packages from the *tidyverse* library (version 1.3.0) (Wickham et al. 2019) primarily used for these functions. The standard R library, *nnet* (version 7.3-15, 2021-01-21) based on the work of Venables and Ripley (2002) has been used to implement feed-forward neural networks and develop the ANN emulators. The R package *randomForest* (version 4.6 – 14 2018-03-22) (Liaw & Wiener 2018) was used to implement the RF algorithm and develop the emulators.

5.2.4 Sensitivity analysis methods used

(The equations presented here in subsections 5.2.4.1 and 5.2.4.2 are repeats of the equations presented previously in sections 3.2.3 and 3.2.4 of this thesis. They are repeated here for ease of reading and simplicity for referral).

5.2.4.1 Morris method indices

Morris (1991) developed a method for measuring the effects that a change in an input variable has on an output of a mathematical model. Process-driven models are a type of mathematical model. The standardised effect of a positive or negative Δ change of an input variable can be calculated using Eq. (5.1). This is also known as the Elementary Effect (EE) (Morris, 1991).

$$EE_i(X) = [y(x_1, x_2, \dots, x_{i-1}, x_i + \Delta, x_{i+1}, \dots, x_k) - y(x)]/\Delta \qquad (5.1)$$

where Δ is magnitude of step, which is a multiple of 1/(p - 1); p is the number of 'levels', or values, over which the variables can be sampled.

Morris detailed the calculation of two measures, namely the mean (μ) and standard deviation (σ) of the set of EEs for each input variable. This was further refined by Campolongo et al. (2007) who introduce the modified index, μ^*, which allows for more robust analysis as positive variations are not cancelled by negative variations. They are calculated using:

$$\mu_i = \frac{\sum_{n=1}^{r} EE_n}{r} \tag{5.2}$$

$$\mu^*{}_i = \frac{\sum_{n=1}^{r} |EE_n|}{r} \tag{5.3}$$

$$\sigma_i = \sqrt{\frac{1}{r} \sum_{n=1}^{r} (EE_n - \mu_i)^2} \tag{5.4}$$

The strength of the relationship between the *i*-th input variable and the output response due to all first- and higher-order effects that are associated with that variable is assessed by the sensitivity index μ_i (Campolongo & Braddock 1999). Campolongo et al. (2007) develop the use of μ^*, the mean of the distribution of absolute values of the EE_i, as given in Eq. (5.3). When $\mu^*{}_i$ is high in contrast to other variables, this input variable has a larger influence on the output value. Conversely, a variable with a low $\mu^*{}_i$ value has low sensitivity associated to it as the same Δ change causes a relatively small change in output (King & Perera 2013). The variance (spread) of the finite distribution of the EE_i values, denoted by σ_i, is calculated by Eq. (5.4). The greater the value of σ_i, the greater the indication of possible interactions with other variables and/or that the variable has a non-linear effect on the output (Campolongo & Braddock 1999). In this research the *morris* function from the R *sensitivity* package (version 1.27.0) (Iooss et al. 2020) was used to implement the Morris screening method.

5.2.4.2 Sobol method indices

Sobol's method (Sobol' 1993) is a global sensitivity analysis approach based on variance decomposition. In this approach the total variance of a function, V(Y) Eq.

(5.5), is decomposed into component variances from individual parameters and their interactions:

$$V(Y) = \sum_i v_i + \sum_{i<j} v_{ij} + \sum_{i<j<k} v_{ijk} + \cdots + \sum_{1\ldots p} v_{1\ldots p} \qquad (5.5)$$

where v_i is the amount of variance due to the i-th parameter X_i, and v_{ij} is the amount of variance due to the interaction between parameter X_i and X_j. The sensitivity of single parameter or parameter interaction, i.e., Sobol's sensitivity indices of different orders, is then calculated based on their percentage contribution to the total variance V:

First-order index $$S_i = \frac{v_i}{V} \qquad (5.6)$$

Second-order index $$S_{ij} = \frac{v_{ij}}{V} \qquad (5.7)$$

Total-effects index $$S_{Ti} = S_i + \sum_{j \neq i} S_{ij} + \cdots \qquad (5.8)$$

where S_i measures the sensitivity from the main effect of X_i, S_{ij} measures the sensitivity from the interactions between X_i and X_j. The total-effects index, S_{Ti}, measures the main effect of X_i, plus the effects of all its interactions with parameters other than X_i (second-order index values). Note: S_i and S_{ij} are limited to the value range ($0 \leq S_i \leq 1$), while S_{Ti} can sum to a value greater than 1.

Sobol has a computational expense, measured as model evaluations, of $(p + 2) * n$, where p is the number model input factors and n is the number of sample sets the users is requesting to be run. Indices are calculated with 95% confidence intervals (CIs) with the minimum and maximum index values defining the CI range for each index value. When n is small, the CI ranges are relatively large. For indices with a true value close to zero, the S_i can be reported as negative, which is a meaningless result, while the CI range includes the value zero. By increasing the number of model evaluations – a larger n, the CI ranges are reduced. The size of n needs to be sufficiently large to produce meaningful S_i values and narrow CIs. A convergence test was undertaken to establish the sample size (n) required. As each input produced

a separate index and CI, the mean of the ranges of all CIs of input factors was used to establish the level of sampling required to give accurate and meaningful results across all output targets. Selected sample sizes (*n*) between 10 to 100,000 were used to establish the convergence towards zero of the CI ranges. Data was combined from simulations representing one sowing date (SowingDOY = 150) at each of three locations (Emerald, Gunnedah and Horsham). Subsequent Sobol analysis runs were all conducted using a sample size (n) of 100,000. In this research the *sobolSalt* function from the R *sensitivity* package (version 1.27.0) (Iooss et al. 2020) was used to implement the Sobol analysis method.

5.2.5 Analysis undertaken

There were two primary foci of the analysis in this research. Firstly, to assess the computational effort and time requirements of undertaking large Sobol analyses using MLEs, and secondly, to appraise the SA results by comparing the indices of the Morris and the Sobol methods produced for the two MLEs developed using different ML algorithmic approaches. In this experiment, MLEs developed to predict the outputs of the process-driven APSIM-NextGen chickpea model were used to undertake the Morris and Sobol analyses to assess the input/output relationships being expressed in the modeled scenarios.

To determine the rate of convergence towards stable predictions of the Sobol first-order and total-effects index values, an initial Sobol analysis was undertaken involving six input factors for the MLEs: FloweringTarget, FracPAWC, LateVegTarget, Population, ShootLag and VegTarget. Analyses were run using sample sizes of 10, 100, 1000, 10 000, 50 000 and 100 000. Subsequent to this initial testing phase, all Sobol analyses were conducted using a sample size of 100,000. To address the first research focus, all 22 input factors used to drive the MLEs were included in Sobol analyses. The same input data set was used for both the ANN and the RF emulators across all six of the output targets. Based on the computational cost of Sobol being (p + 2) * n, using a sample size of 100,000 equated to (22 + 2) * 100,000 = 2.4 million model evaluations per model. There were six MLEs, one for each target output to be predicted, by two model classes, ANN and RF. In total, this represented 2.4 million * 6 * 2 = 28.8 million model evaluations. The Morris

analyses were undertaken using samples of 10,000 and required an additional 2.76 million model evaluations in total.

In addition to the analysis using 22 input factors, a second series of analyses was conducted for a reduced set of six input factors, the remaining input factors being held constant for the Sobol analyses. This series of simulations removed the variations in the environmental and climatic factors that dominated the first set of analyses, thereby allowing the influences of variations in varietal factors to be observed. By observing the influences of these factors, that normally have subtle effects on the output values, the suitability of MLEs to perform SA can be more thoroughly understood, and differences between the analyses by the types of MLE can be highlighted. The six input factors being varied for this series of experiments were, FloweringTarget, FracPAWC, LateVegTarget, Population, ShootLag and VegTarget. To ensure robustness of testing, simulations were selected from the original MLE development data test sets which showed very strong correlations to the original APSIM-NextGen chickpea results for all six output targets. Three simulations were selected, one for each of the sowingDOY 100, 150 and 200, for each of the three test locations, Emerald, Gunnedah and Horsham. Sobol analyses were again undertaken with sample set sizes of 100,000 and Morris analyses with sample sets of 10,000.

5.3 Results

The results of the initial task of establishing the sample size required to obtain reliable index values are shown in Figure 5-2. This analysis is for six target outputs for both the ANN and the RF MLEs. Values shown are the mean widths of the CIs for the six input factors being analysed. Both MLEs showed very similar patterns and rates of convergence across all six output targets: EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt. In all cases the first-order indices showed greater variability than the total-effects indices for the same sample size and the same output target. When the sample size was 50,000 or greater, the CIs for both first-order and total-effects variabilility were nearing zero and the underlying index values had attained stable estimates.

Shown in Figure 5-3 are plots of index values and their CIs for the six input factors for three selected outputs: FloweringDAS, MaturityDAS and GrainWt, for sampling sizes of 100, 1000 and 10,000. Of note here is the number of RF indices reported as negative values for the sample size of 100. This is a clear indication that the implementation of the Sobol method used in this study requires a considerably large sample size in order to calculate meaningful statistics.

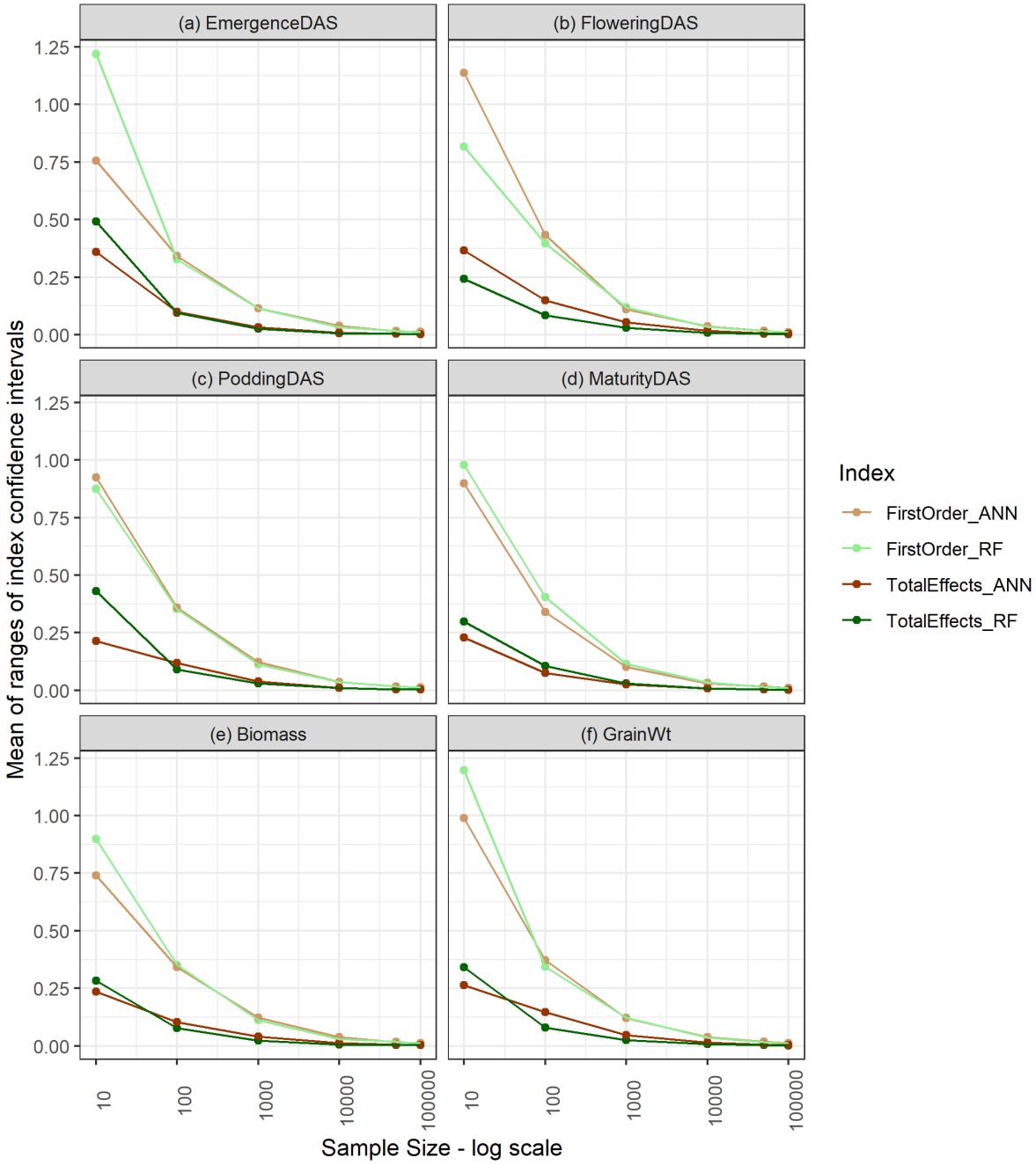


Figure 5-2. Convergence of Sobol index mean band widths.
The mean width of the confidence intervals of six input factors is plotted for the Sobol first-order and total-effects indices for sampling sizes from 10 to 100000. Twelve machine learning emulators (MLE) are assessed; two classes, ANN and RF, for each of six outputs. Each output requires a different MLE. The convergence towards zero indicates more stable estimations of the value of the underlying statistic.

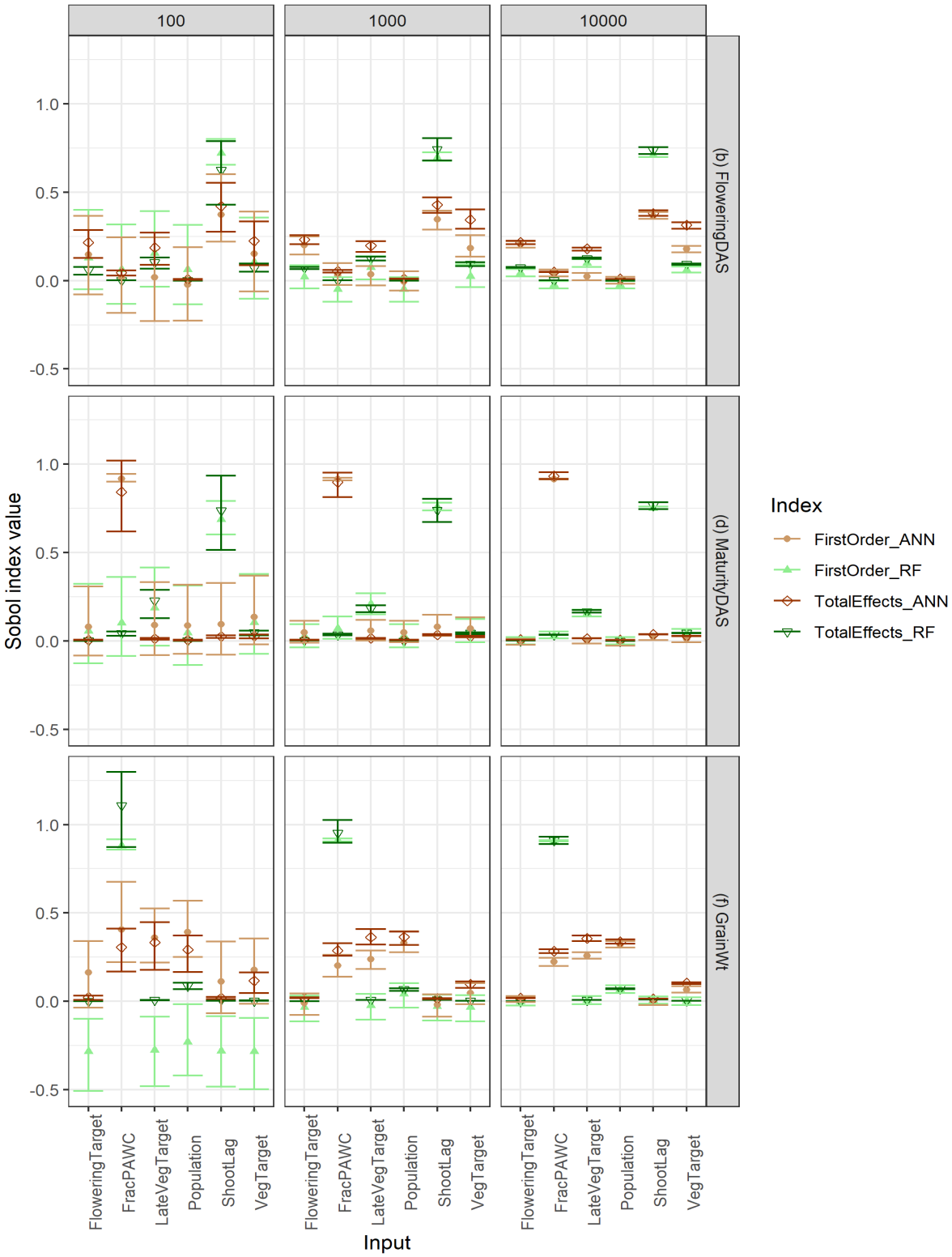


Figure 5-3. Confidence Intervals (CIs) plots.
Plots are for six input factors for each of two machine learning emulators for three selected outputs. The three horizontal panels show the narrowing of the CIs for increasing sample sizes. The plots are for location: Gunnedah, sowing #2.

5.3.1 Computational efficiency of MLEs

The computational efficiency of the MLEs became evident during this testing. For the analyses using a sample size of 100,000, which equates to 800,000 model evaluations with the six input factors being assessed, the average time taken to run each model was 3.3 seconds for the ANN MLEs and 24.7 seconds for the RF MLEs. This included all steps of the Sobol methodology from generating the Monte Carlo sampling plan, generating the input data set, running the input data through the MLE, combining the generated output with the inputs and calling the Sobol routine to generate the Sobol indices. As this represented a negligible time constraint, all further Sobol analyses run in this research experiment utilised the large sample size of 100,000.

5.3.2 Sobol analysis of all MLE input factors

The Sobol analysis of all input factors for the MLEs involved 22 parameters, and required the evaluation of 2.4 million emulator runs for each of two classes of MLE, ANN and RF, and six target outputs, giving a total of 28.8 million model evaluations. Results are shown in Figure 5-4. The input factors having the greatest effects varied between output targets, and also varied, though to a lesser extent, between the MLE types. Broadly, the most influential factors were the maximum and minimum temperatures, with SowingDOY also having an influence on FloweringDAS, PoddingDAS and MaturityDAS, but only for the ANN emulators. The RF emulators showed responsiveness to fewer factors than the ANN emulators. In all cases, the RF emulators showed first-order index values equal to, or very nearly equal to, the total-effects index value. This is indicated in Figure 5-4 where the green dots (FirstOrder-RF values) align with the open green circles (TotalEffects-RF values). The RF emulators are shown to be not affected by second order interactions between input factors. This is an unexpected result. This result indicates that the RF emulators are disregarding second-order interactions in the input factors. This may be due to the RF algorithm removing some of the complexities of the process-driven model as the RF emulator is only predicting one output target at a time. The exact cause of the observation that second-order effects are not being influential is easily anlaysed. The ANN emulators, likewise, show close alignment of first-order and total-effects

indices – brown dots and brown open circles in Figure 5-4, though not to the same extent as the RF emulators. In all cases, the first-order index was greater than, or equal to, the total-effects index, so the results are not overtly wrong.

Figure 5-4. First-order and total-effects Sobol indices of 22 input factors.
First-order and total-effects indices were calculated for the 22 input factors that drive the machine learning emulators (MLEs). Two classes of MLE were included in the analysis, an artificial neural network (ANN) and a random forest (RF). Emulators for six target outputs: EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt, were analysed.

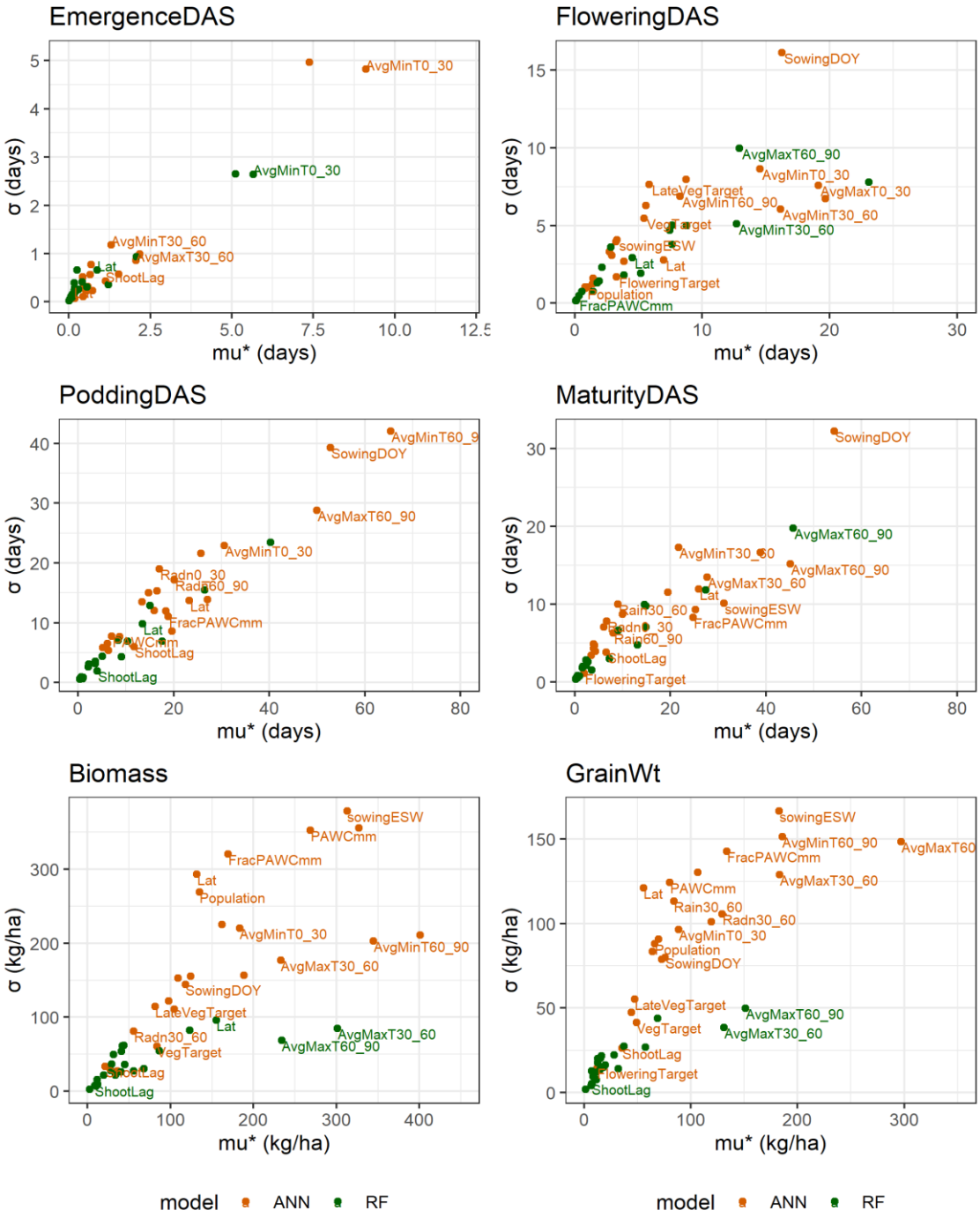


Figure 5-5. Morris analysis for 22 input factors.
The factors include all input parameters used to drive the machine learning emulators (MLEs). Two classes of MLE were included in the analysis, an artificial neural network (ANN) and a random forest (RF). Emulators for six target outputs: EmergenceDAS, FloweringDAS, PoddingDAS, MaturityDAS, Biomass and GrainWt, were analysed.

5.3.3 Morris analysis of all input factors

The results of the Morris analysis of the same input factors used for the Sobol analysis are shown in Figure 5-5. For the Morris analysis, the μ* value is a measure of the mean absolute elementary effect that the input factor has on the output value, while the σ value is a measure of the variance of the elementary effects. It is clearly seen in Figure 5-6 that the ANN emulators had larger elementary effects and larger variances than the RF emulators. This is further shown in Figure 5-6 where the μ* and σ values for the two classes of MLE are plotted in XY-scatterplots. The majority of points for both μ* and σ fall below the one-to-one line indicating lower values for the RF emulators for both statistics. The reasons for this observation are not easily discerned as the emulators are 'black-boxes' as to how values are actually derrived. One possible factor that could be contributing to the RF emulator reporting lower effects lies in the part of the method used to assess the influence of input values. The RF algorithm uses a processes called bagging to combine and summarise groups of output results from clusters of decision trees. It is the bagging effect that helps the RF models to be more tolerant of outlier values and noise within the data, without thowing out the accuracy of its predicted values. This means that it appears to be less reactive to input value extremes, which for the Morris analysis may be reflected in lower μ* and σ values being calculated. More detailed analysis of the values contributing to the panels in Figure 5-6 were undertaken. The plot of the EmergenceDAS values is shown in Figure 5-7. Of note here is that environmental factors, such as the temperature values, again dominate the elementary effects, with phenological factors, such as FloweringTarget and VegTarget being concentrated close to zero in the bottom left quadrant. The lower the values, the less influence they have on the output value. Only Biomass and GrainWt showed greater sensitivity to non-climate factors, such as FracPAWC, PAWCmm, Lat and Population, with the ANN MLEs showing significantly higher sensitivity than the RF MLEs.

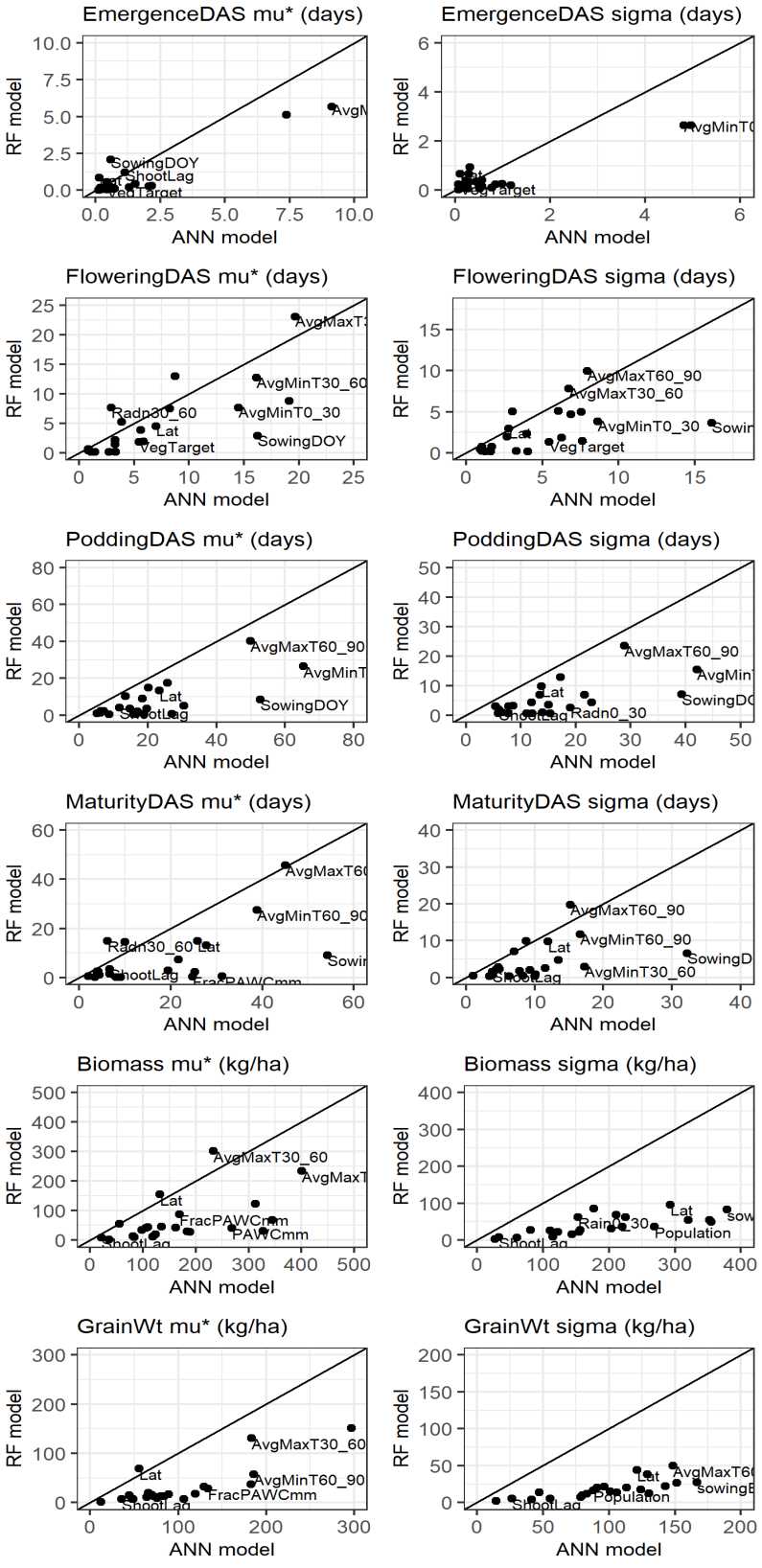


Figure 5-6. XY-Scatterplots of the mu* and the sigma values for the RF and ANN machine learning emulators.
Values below the one-to-one line indicate smaller elementary effects and small variances in the RF emulators than is observed in the ANN emulators.

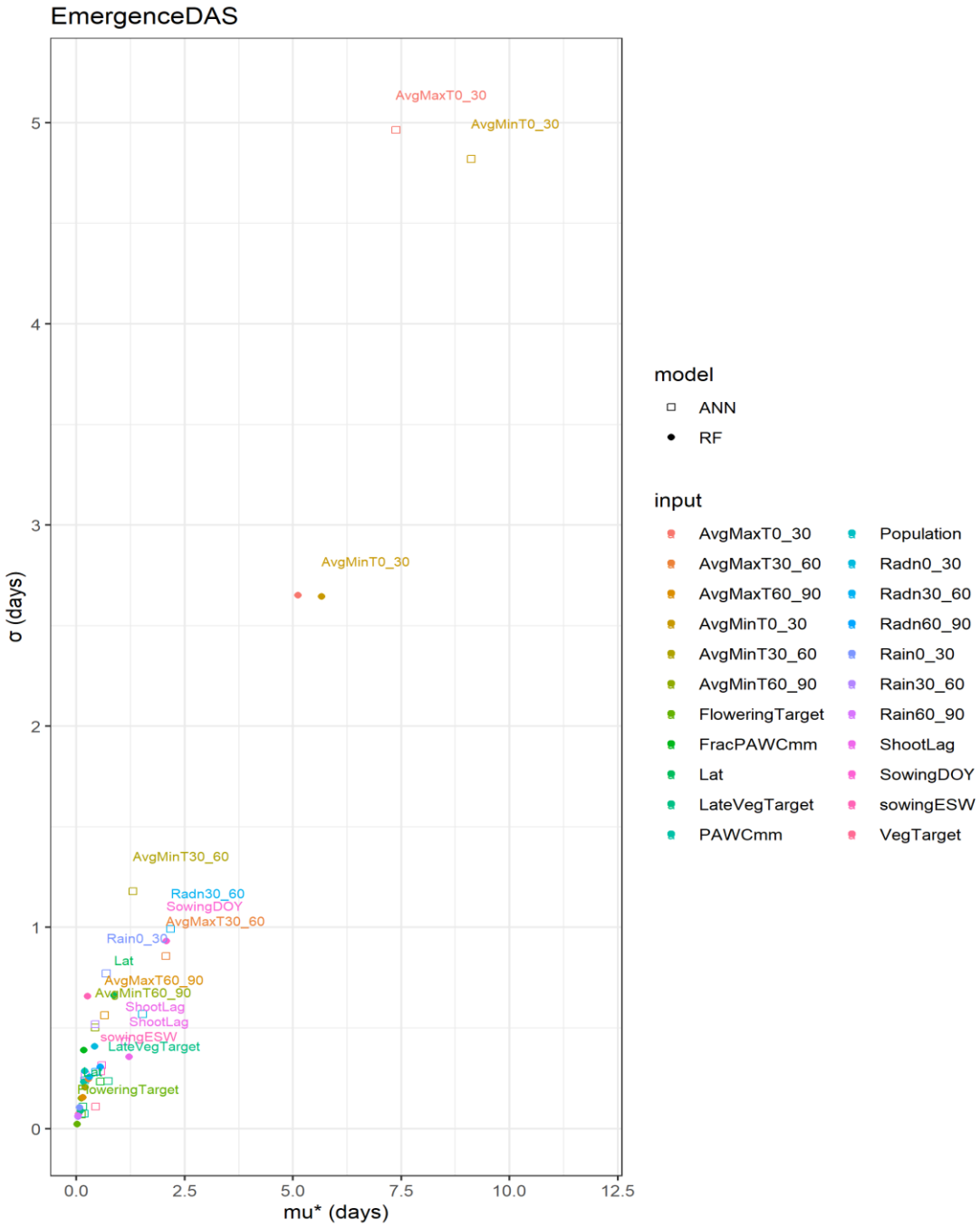


Figure 5-7. Morris statistics for the output target EmergenceDAS from two MLEs, artificial neural network (ANN) and random forest (RF).
Values are shown for the 22 input factors that drive the machine learning emulators.

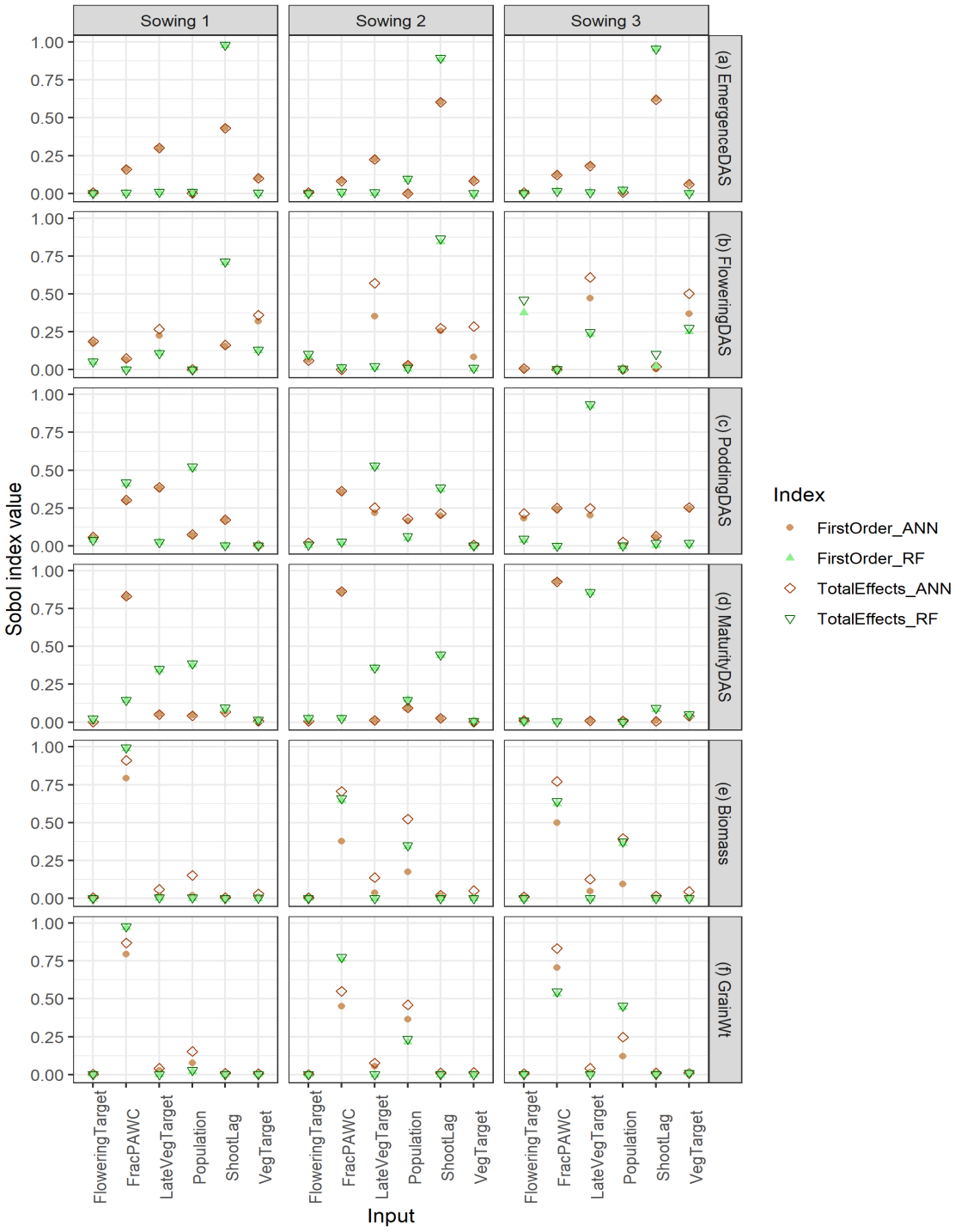


Figure 5-8. First-order and total-effects Sobol indices of six phenology input factors. Sobol indices were calculated for the six input factors that are phenology parameters or are significant drivers of phenological processes. Environmental factors, such as temperatures and rainfall, have been set to fixed values. Two classes of MLE were included in the analysis, an artificial neural network (ANN) and a random forest (RF).

5.3.4 Sobol and Morris analysis of phenologically focused factors

The Sobol and Morris analyses of the 22 input factors that drive the MLEs showed that environmental factors, such as temperature, were the dominant driving factors for each of the MLEs for the six output targets reviewed. In order to assess the effects of the genetic factors included in these input factors, these environmental factors were held constant and the factors more directly influencing phenology were assessed. This was done by assessing three separate sowing dates for one location, and varying only six input factors: FloweringTarget, FracPAWC, LateVegTarget, Population, ShootLag and VegTarget. The plots of the Sobol indices for the Gunnedah sowings are shown in Figure 5-8. Each of these factors shows increased importance for at least one of the outputs for at least one of the MLEs. The different sowing times show varying sensitivity to different factors. For example, FloweringDAS is shown to be sensitive to ShootLag in the first two sowings for RF, but not very sensitive in the third sowing. The ANN emulator shows a similar pattern, but for a much lower portion of the variance. For Biomass, Population is not a contributing factor for the first sowing for RF, but is a contributing factor for the second and third sowings, while ANN shows a small sensitivity for the first sowing. ANN also indicates significant second order effects linked to Population of sowings two and three, while RF shows no second order effects.

5.4 Discussion

Sensitivity analysis of complex modelling systems requires large numbers of model simulations to be run. Efficient screening approaches, such as the Morris elementary effects method may take as few as several hundred simulations to be run to provide meaningful results, while variance decomposition methods, such as Sobol, often require many thousands, or even tens of thousands of simulations to be run. Although improvements in computing power continue to make running large numbers of simulations less of an issue, there remains great advantages in reducing the overall cost of running large simulation sets. It is impossible for a range of reasons to directly compare the computational burden of running SA on a process-driven model with the burden of running MLEs trained to predict the outputs of the

process-driven model, but on a simple time of execution basis, the APSIM-NextGen chickpea model required in excess of six hours to run 80,000 simulations, while ANN based MLEs took 3.3 seconds to generate 800,000 simulation results on the computer used in this study. Ignoring the many dissimilarities between these two approaches, this represents a speed improvement of approximately 50,000 times to produce the model output values required for a Sobol analysis. The RF MLEs reduced simulation times by a factor of approximately 6,000 compared to the process-driven model. Some of the factors that are not taken into consideration in this time comparison are: the process-driven model produces all required outputs in a single execution run, while MLEs produce a single predicted output value per model execution; a separate MLE needs to be built/trained for each desired output target; process-driven models evaluate multiple years of weather data within a single run, while MLEs evaluate only a single set of climatic settings within a single run; and input setting for process-driven models are much more robust, while MLEs require careful input selection to avoid invalid simulation results. These observations are supported by Stanfill et al. (2015) who highlights the high level of time and care required by researchers to ensure SA undertaken with emulators does not return qualitatively incorrect results – an issue that is noted as a common problem with non-ML emulators (Stanfill et al. 2015). None of these issues are inconsequential in estimating the time and effort required to generate predicted results of simulation models using MLEs.

The Sobol analysis of 22 inputs of the MLEs (Figure 4-5) shows that both the ANN and RF emulators are generally sensitive to the same inputs as each other across all six target outputs, though the actual proportions of the variations in output values assigned to particular inputs varied between them. Of note in Figure 4-5 is that for all RF indices, the first-order index of a value coincides almost precisely with the total-effects index, as shown by the green dot (first-order index) being superimposed on the green triangle (total-effects index). The indices of the ANN emulators (brown dots and open brown diamond shapes) also show the same tendency, although there are some examples where the total-effects index is greater than the first-order index, e.g. for SowingDOY in the prediction of FloweringDAS. This coincidence of values indicates that the RF emulators and, to a large extent, the ANN emulators, are not sensitive to second or higher order input factor interactions, at least in this analysis.

A conclusion that there are no second or higher order interactions between input factors is certainly incorrect. For example, the sowing location's latitude (Lat) will be interacting with the SowingDOY for the crop's growth. Molnar et al. (2020) in their research about pitfalls in understanding ML model outputs, caution about problems with interpreting ML model outputs and linking them with inputs via assumed causality.

Another observation from the Sobol analysis of 22 input factors (Figure 4-5) is that ANN emulators are sensitive to more inputs than are RF emulators. Based on the results presented in Chapter 4, the predictive abilities of both the ANN and RF are quite similar. This indicates that, although the output values are quite similar, the MLEs are using different information to achieve their predicted outputs. While the results of the SA on the MLEs is not overtly wrong, the results are not entirely consistent with those of the original process driven model. The Morris analysis undertaken (Figures 4-5 and 4-6) clearly demonstrate the differences between the two MLE classes. The elementary effects (μ^* values) associated with each of the outputs of the ANN emulators were significantly higher - more affected by variations in the input values - than were the outputs of the RF emulators, and the variance of the output values (σ) was also much higher for the ANN emulators than the RF emulators. The clustering of the RF values near the (0,0) origin of the Morris (μ^* versus σ) plots can be seen in the sample EmergenceDAS plot in Figure 4-7. Overall, the Morris method highlights the differences between the ANN and RF emulators more clearly, while the Sobol method, which is reporting the proportions of the variances observed, tends to show greater consistency between the emulators.

Fixing the values of the environmental input factors by selecting one location and three selected sowing dates from the training data set, allowed a second Sobol analysis to focus on the six inputs most closely involved in the phenology of the crop. These input factors: FloweringTarget, FracPAWC, LateVegTarget, Population, ShootLag and VegTarget, showed virtually no sensitivity when included in the set of 22 input factors, but demonstrate significant contributions to the variability of all six output target values when their effects are assessed as a small group of six genetic and management input factors (Figure 5-8). Even between the different sowings, factors are shown to contribute varing amounts to the output's variance. For example, ShootLag's contribution to FloweringDAS is significantly more

pronounced in sowings one and two (Figure 5-8) for both models, than for either of the models in sowing three. The dominance of environmental factors over genetic factors in broadscale simulation experiments is logical and observed in all biophysical simulation systems (Chen et al. 2016). The effects that genetic variations, as represented by phenolgical process settings, have on overall crop growth will commonly be orders of magnititude smaller than the effects that environmental factors; temperature, water and light, have on the the crop's growth potential. So these results are in keeping with general knowledge. They do, however, highlight that the different classes of MLEs respond differently to the same input values while producing very similar outputs.

The use of MLEs does offer some interesting possibilities for the design of simulation experiments. Exceptional computational efficiency means that the size of simulation experiments can be extended into the realms of what has, until now, only been possible with the use of large computing clusters. Data, the lack of which is possibly the greatest limiting factor to the use of ML models, is largely in the hands of the researcher, as the data sets are generated by the running of the process-driven model. Carefully selected parameters and experimental design can produce data sets that generate robust and highly accurate emulators for specific outputs. Additional inputs can be specified for the development of the MLEs, inputs that are not included in the inputs of the process-driven model. For example, the summary climatic values used in this research were generated as outputs from APSIM. There are no equivalent summary values available as inputs to the APSIM-NextGen chickpea model. Experiments could be designed to use these inputs to the MLEs to easily appraise the influence of climatic variations on crops at different growth stages. Novel data sources could also be used, such as remote imagery, provided validated output results are available, or can be generated, with which to train the MLEs. The SA of such simulation experiments would be expected to generate new knowledge.

5.5 Conclusion

Results from the research undertaken in this chapter of my thesis have answered the third research question for this thesis in the negative. That is, that undertaking SA

using MLEs cannot be considered equivalent to undertaking SA on the underlying process-driven model that the MLEs were built to emulate, regardless how accurately the MLEs predict the outputs of the base model. The SA will reflect the input/output relationships of the MLE rather than those of the process-driven model. Machine learning models developed to emulate the outputs of process-driven biophysical models, referred to in this research as machine learning emulators (MLEs), have been shown to be highly computationally efficient, both in term of their speed of execution and the nominal level of hardware required for them to operate. The evaluation of millions of sets of input variables to produce predicted output values, and the subsequent SA using either Morris or a Sobol methodology was shown to be a very achievable task. The interpretation of the results, however, is not so straight forward. The analysis of the MLEs showed variations between the ANN and RF classes of model. While not overtly incorrect, the results show that no certain conclusions can be drawn about the internal functioning of the underlying process-driven model that the ML models were emulating. For the detailed SA of input parameters of process-driven models, this research has shown that the approach of using MLEs for the analysis using the Sobol method has limitations and may not provide the analysis expected. Using this approach to apply the Morris method is more robust than applying the Sobol method with respect to consistency of results from the underlying process-driven model. The approach does offer advantages in the areas of computational speed, and the ability to redefine or extend the functionality of the underlying process-driven model through the inclusion of new or modified data sources. In situations where very large numbers of simulations are required to be run and the computational burden of running the required number of simulations is a limiting factor, then the utilisation of an MLE to improve the computational speed is worth consideration. This approach also opens the possibility of modified and expanded experimental designs, with the SA of these experiments potentially contributing to new knowledge in the research area.

CHAPTER 6: GENERAL DISCUSSION AND CONCLUSIONS

With simulation modelling playing an increasingly important role in diverse disciplines spanning science to medicine to public policy to defence, the need to have carefully validated models and certified outputs continues to grow in significance. Sensitivity analysis (SA) is one of the critical tools used to validate models (Razavi et al. 2021). As models and modelling systems become more complex, the computational burden of undertaking thorough analyses becomes more challenging. At the same time, the area of machine learning (ML), a discipline rooted in the manipulation and analysis of large data sets, continues to develop, and provide solutions to previously intractable problems such as remote image analysis (Chowdhury et al. 2015; Gilbertson & Niekerk 2016; Pantazi et al. 2016). Rather than attempting to replace the process-driven biophysical models that are the building blocks of larger and more complex modelling systems, this thesis has investigated the potential use of ML emulators to undertake the SA of the underlying process driven models. The example process-driven modelling system used in this research was the APSIM-NextGen framework. The goal here has been to avail the model developer and end user of the most useful and powerful features of each modelling approach, the flexibility, robustness and proven fit-for-purpose qualities of the process-driven modelling system, and the computational efficiency, data processing capacity and innate flexibility inherent in the ML approaches. Through a review of the literature (Chapter 2) it was identified that SA, although considered as best-practice for simulation model development and validation, is not as widely used as it could be, and has very limited adoption of use in the development of process-driven biophysical crop models. The approach of using MLEs to assist in performing SA of process-driven models by alleviating the computational burden, including for SA methods as computationally intensive as variance decomposition, has not been assessed. This applies to process-driven biophysical models specifically, but also to process-driven modelling systems more generally.

In agricultural systems modelling, the two most commonly used SA methods are the Morris and the Sobol methods (Morris 1991; Sobol' 1993, 2001; Campolongo et al.

2007). The first objective of this research project, the focus of Chapter 3, was to establish if MLEs, generated to predict the outputs of a process-driven model, in this case APSIM-NextGen chickpea, could be shown to have the same or very similar predictions of the sensitivity of output parameters to the input parameters as was shown by the Morris and Sobol methods for the underlying process-driven model. This has not previously been demonstrated in literature relating to process-driven models. Due to the need to be able to determine exactly how the results were being calculated, the MARS algorithm was selected as the ML approach to be used in the testing. The RF and ANN algorithms were considered, but their 'black-box' qualities for how output values are calculated meant that they were considered as not ideal for this phase of the experiment, and so were not included to generate MLEs. The results of Chapter 3 revealed that there is a strong correlation between the sensitivity indices calculated for the MLE for seven output parameters, and those calculated for the Morris ($R^2 = 0.89$) and Sobol ($R^2 = 0.82$) methods. The implication of this is that a SA of a MLE should give comparable, though not identical, results as a SA of the process-driven upon which the emulator as based. While this was the desired, and expected, outcome, it was not a forgone conclusion as ML is not driven by processes that cause the output value, as is the case for process-driven models, but rather by statistical analysis of inferred correlations between inputs values and output values (Razavi et al. 2021). This approach of generating a MLE of the process-driven model does not guarantee creating a method with improved computational efficiency for performing SA, nor does it offer a one size fits all solution to performing SA. The Morris method was shown to be at least as computationally efficient as the MLE. So, for tasks such as screening for important inputs, the Morris method would remain as the preferred choice as it does not incur the additional costs of building the MLE. However, in the case of the Sobol method, the computational efficiency of the MLE showed promising results for potential efficiency gains.

Having established that using MLEs for SA of process-driven models has the potential to deliver computational efficiency gains, the next research objective of this thesis, the focus of Chapter 4, was to determine which ML algorithms produced the best emulators based on accuracy of predictions (outputs) and speed of development. In reviewing the use of ML in the areas of agricultural systems modelling (Chapter 2), it was found that there has been no research published which reference the use of

ML based emulators for process-driven biophysical models. Emulators have been used in ecological modelling disciplines, such as hydrology, and these include, linear regression and support vector machines (Villa-Vialaneix et al. 2012), variable importance measures, random forests (Wei et al. 2015), and polynomial chaos expansion. One feature that stands out in each of these approaches, except for the random forest approach, is that computational gains are obtained primarily by reducing the complexity of the problem which is achieved by reducing the number of dimensions being considered in addressing the problem. The issue of dimensionality reduction is not a trivial one. For an emulator to be suitable for use in SA, it needs to respond appropriately to less-common mixes of parameters, as is particularly the case in environmental factors like climate, where extreme weather events need to be handled. This is because a thorough SA will use a full factorial of parameter values, and less common scenarios are likely to be included. Some factors that are of little, or no significance generally, might become critically important in extreme situations. Eliminating them from the emulator will compromise some outputs. It was this aspect of eliminating some input factors from those used to develop the MLE that led to questions about the suitability of the MARS algorithm for undertaking SA. It was decided to include the MARS algorithm in the experiments for Chapter 4 for continuity with the previous chapter, and to provide a base for comparison against algorithms that were to be newly included in Chapters 4 and 5.

Machine learning is more commonly used to generate models of systems directly, rather than generate emulators of existing process-driven models. The ML algorithms used for the generation of predictive agricultural models include MARS (Fortin et al. 2014; Deo et al. 2015), ANN (Kuwata & Shibasaki 2016; Pantazi et al. 2016; Nettleton et al. 2019), and RF (Villa-Vialaneix et al. 2012; Newlands et al. 2014; Thessen 2016; Kouadio et al. 2018). These algorithms, having been shown to be adaptable and appropriate for the generation of models to simulate complex agricultural systems and predict crop yields, were selected for the investigation into generating ML emulators (Chapter 4). These three ML algorithms represented three distinctly different classes of algorithm, (a) MARS a regression-based approach, (b) ANN a neural network based approach, and (c) RF a decision tree based approach. The extreme learning machine (ELM), an advanced form of ANN that was used by Deo et al. (2015) and Kouadio et al. (2018) for ML modelling of crops and drought

indices, was also considered for inclusion. However, after a thorough review of the performance of an implementation of an ELM in the R computing environment, it was decided that the available implementation of the ELM was not fit-for-purpose, and a standard ANN was used in its place.

Each of the three classes of algorithm produced emulators that performed impressively well in terms of accuracy for each of the six model outputs. The least accurate predictor was the MARS emulator for GrainWt with an accuracy of R^2 = 0.88, while each of the RF emulators had an accuracy of prediction between R^2 = 0.98 and R^2 = 1.00. The computational effort as reflected in the execution times required to generate the emulators showed significant variations with the MARS emulator being the fastest to create and train, while the ANN emulators took approximately three times longer, and the RF emulators taking approximately 500 times longer. So, although slower to generate and execute, the RF emulators were consistently the most accurate. These details are not often reported in literature, and while they represent only a very specific instance of a diverse and complex modelling system, the comparison of the performance of the ML approaches may well be of interest to biophysical model developers.

Models built on ML algorithms need data, and lots of it (Buchanan & Miller 2017). The MLEs generated in this research were developed on data sets generated from the underlying process-driven modelling system. As such, there was almost limitless amounts of data available to train these emulators. Any issues around the lack of data for model generation (Montesino-San Martin et al. 2018) was avoided. However, the defining of modelling scenarios to run in the process-driven biophysical modelling system in order to create the data to generate the MLEs, is not without its issues. One issue that was encountered early in the experimental design process, but not recognised fully until much later on, was that of valid parameter ranges. While parameters were all well within the valid ranges for the process-driven biophysical models, some of the combinations of values resulted in a failure to produce crops. Most of the combinations of parameter settings that did fail to simulate a mature crop were valid combinations of values, but due to other uncontrolled factors, such as weather, the crop failed. This is not an issue in the process-driven modelling system and can be used to define acceptable boundaries for parameter setting in given

situations, such as time of sowing experiments. For ML models, however, the presence in the data of multiple diverse parameter settings which all result in the same target value, zero in the case of yield for a failed crop, simply creates noise and the model becomes less accurate (Gollapudi 2016). The resolution of this issue, without unduly limiting the acceptable ranges of input parameters values, is not an easy task, but is one that cannot be ignored if attempting to create MLEs.

A second issue with generating data sets was clearly shown by the simulation of the two additional test locations in Chapter 4. Although the MLEs had performed extremely well on the training locations (Figure 4-1, locations 1 to 7), and this was assessed using previously unseen test scenario data for those locations, the performance of the emulators at predicting model outputs at completely new locations (Figure 4-1, locations 8 and 9) was very poor. This was because the parameter values used, or the pattern of relationship between input values and output values, had not been encountered in the training data set. The MLEs simply had no 'knowledge' as to how to predict the output values. This is a known limitation of ML predictive algorithms (Buchanan & Miller 2017). This issue highlights the need to ensure all scenarios and all parameter value combinations of interest are covered by the values and patterns in the training data set. While process-driven models are likely to give reasonably accurate answers for scenarios that fall outside the 'usual' settings, but within the model limits, ML models will not fail gracefully, but are more prone to give completely unrealistic predictions.

A third data related issue that became evident during this research was one related to dimensionality reduction. The MARS algorithm, by its design, eliminates the use of parameters that are deemed to not have sufficient influence on the value of the output (Friedman 1991a; Friedman & Roosen 1995). This effectively reduces the dimensionality of the model and approximates the details of the system that it is trying to predict. While this improves the computational speed of the model execution, it also removes details about the system being modelled. In the case of SA, this dimensionality reduction can remove some of the parameters that are of interest to the analysis, either due to direct influence, or by interaction. It was for this reason that the MARS model was deemed to be unsuitable for inclusion in Chapter 5 of this research, the comparison of the performance of MLEs in undertaking SA.

The final part of this research was to compare the sensitivity analyses of the different MLEs. The first part of this assessment is contained in Chapter 4 and is covered by Figure 4-4 which presents a heat map of variable importance indices. As discussed in Chapter 4, this analysis showed clearly that the different MLEs vary in the importance that is placed on input variables for calculating the output value. By using the Morris or Sobol method to undertake the SA (Chapter 5), these variations were less pronounced, but the MLEs still showed variations between their results. These results show that none of these measures of sensitivity taken from the MLEs provide a reliable analysis of the input-output linkages of the underlying process-driven model. The comparison would benefit from undertaking both a Morris and Sobol analysis on the APSIM-NextGen chickpea model, though a direct comparison between the SA of the APSIM model and the MLEs would not be possible due to differences between the input parameter sets.

The potential of MLEs to provide a way of alleviating the computational burden of running large simulation experiments in APSIM itself was clearly demonstrated. The combination of highly accurate predictions, especially from the RF emulators, and the extraordinary speed of execution of the MLEs, which produced results many thousands of times faster than is possible using a process-driven biophysical model, highlights this potential. In relation to using this approach for the SA of the underlying process-driven models, the analysis shows that the MLEs generate results based on differing patterns of input variable importance, so the approach is not an ideal fit for analysing the underlying model. The issue of MLEs being 'black-boxes' of functionality is part of the problem. It might be summarised as, we can teach the machine to learn from the data that we supply it, but it is not easy for humans to learn from the machine and be certain about why the results are as they are. To generate well validated MLEs from within APSIM-NextGen and use them for undertaking simulation experiments or SA, would require a thorough knowledge of the development process for generating MLEs and high levels of automation of the procedures involved. Insights from this research have revealed that the automation of these processes would not be a simple task. Of particular complexity would be the selection and cleaning of the data needed to build accurate and robust MLEs that fully meet the requirements of the research being undertaken. Molnar et al. (2020)

raises a number of concerns around the issue of data selection and input data interactions. APSIM can be used to generate very large data sets quite easily, but ensuring all parameters used contribute to 'successful' outcomes, or determining which inputs have resulted in failed outcomes, is not straight forward. The work by Shastry et al. (2016) also comments on the essential need to carefully clean data before generating ML models. The solution to the problem may well involve convolutional ML networks where inputs and outputs are clustered to allow the establishment of limits to the ranges of values that will work, and the identification of combinations that will fail. Other forms of deep learning networks might also be required to ensure the required input parameters are retained if undertaking SA but discarded if not needed for other analyses. In short, the potential for automated systems appears to be present, but the path to fulfilling them may be a difficult one.

An alternative way of using MLEs for SA might lie in a new approach to generating simulation experiments. The almost limitless potential of process-driven models to produce data sets of input and output values goes a long way to overcoming one of the greatest limitations to developing ML models, that of the lack of data. Provided that data sets are well designed and include all relevant inputs and matching outputs over the complete range that a ML model is desired to work, then very accurate and computationally efficient MLEs can be developed, as demonstrated by this research. In addition, novel data sources, such as remote imagery, can be included in the generation of the MLEs, provided validated outputs can be sourced or generated with which to train the MLEs. These would be hybrid models which would incorporate the outputs of process-driven models and inputs from other data sources, as demonstrated in the recent work by Paudel et al. (2021) This opens options for simulation experiments that are simply not possible with process-driven models alone. The research included in this thesis is at the juncture of the fields of process-driven biophysical model development, agronomy, plant physiology, machine learning emulators, and global sensitivity analysis. The outcomes of this work have implications for model development and model application in all these disciplines. Firstly, the Morris method remains a more computationally efficient choice, when compared with the development and use of MLEs, for the screening of importance of parameters of process-driven models. Secondly, the results show that, while both Morris and Sobol analyses produce very similar results across different MLEs, the discrepancies indicate that great caution is needed if interpreting these results as a

way of understanding the underlying process-driven model and its input-output sensitivities. The results suggest that by using the computational efficiency of an MLE, SA of large-scale simulation experiments becomes more feasible, and this can contribute to efficiency gains for scientific research. The SA of enhanced forms of simulation experiments produced by hybrid models, which use the outputs of process-driven models and combine these with other sources of data to create new forms of ML based agro-ecological models, is suggested by this research as a direction that could be pursued to advance agro-ecological modelling.

Emulator and surrogate models' use is well documented in literature (Razavi et al. 2012), although the use of any form of emulator in the field of agricultural systems modelling continues to be a relative rarity. The research included in this thesis attests to the significant potential gains offered by MLEs for agro-ecological modelling. A recent publication (Razavi et al. 2021) highlights the considerable interest by experts in a wide range of disciplines in supporting the notion that SA is such an important and integral part of modern simulation modelling, that it deserves to have its own discipline in the science and mathematics communities, and that it be taught in higher education with formalised curricula. In addition, there is increasing recognition of the need to address the questions and concerns around the functioning of ML models and the need to offer formal validation of the outputs. The specialist research area of 'Interpretable machine learning models' (Molnar et al. 2020) has the potential to address some of the concerns raised in the discussions of Chapters 4 and 5 of this thesis about the use of ML in SA. In particular, these concerns include the interpretation of ML outputs, variable importance values, and the effect of data selection on the reliability of output values. Many of these issues are also raised by Razavi et al. (2021) in their review of the current state of knowledge of SA involving the use of ML. These research areas will play key roles in advancing the use of ML in SA for a wide range of science disciplines.

To summarise: Chapter 3 addressed the question of whether an ML based emulator developed to simulate the outputs of a process-driven biophysical model, shows the same, or at least comparable, mappings between input parameter values and output values when subjected to some form of SA as an analysis performed on the underlying process-driven model. A method has been developed that allows the

sensitivity indices produced by the Morris and the Sobol methods run on the process-driven model to compared with the variable importance indices produced by the MLE for its input parameters. The results of this analysis showed that the variable importance indices of the MLE were very similar to the SA indices produced by, particularly, the Morris method, and to a lesser extent to the indices produced by the Sobol method. This answered the first research question in the affirmative. Subsequent research required for addressing the second research question in Chapter 4, however, highlighted significant potential limitations in this approach. Specifically, if the ML algorithm discards input variables assessed to be of insignificant importance (dimensionality reduction), as many of the approaches do to maximise computational efficiency, then the SA can no longer be considered to represent a true analysis of the underlying process-driven model that is being driven by all the input parameters.

In Chapter 4, I addressed the second research question: Can ML based emulators be built to accurately predict the outputs of process-driven biophysical models, and if so, are some ML algorithms more suited to the task than others based on accuracy of predictions and computational effort incurred? This research question was answered in the affirmative, with the results of the research showing that each of the three ML algorithms tested, ANN, MARS, and RF, all produced emulators that generated accurate predictions, though RF was shown to be the most consistently accurate across all output parameters, while the MARS emulators were slightly less accurate and less consistent in their accuracy. The computational burden of generating and running the emulators was basically the opposite, with the MARS emulators being the least expensive to build and run in terms of computational effort, while the RF emulators were many hundreds of times more computationally expensive. Additional testing of the emulators to generate predictions for test sites that were not included in the training data sets but were within the geographical boundaries of the chickpea production zones being simulated, showed the need to ensure ML model training data sets comprehensively cover all value ranges and combinations that might occur in live production data sets. In the case of the test locations used in Chapter 4, soil profile values fell outside the ranges for which the MLEs had been trained and as a result none of the MLEs were able to generate sensible predictions. The implication of this is that MLEs need to be generated to specifically address the problem that

they are going to be used to answer. This requirement then focuses attention on the need for the careful design of the simulation experiments to be run on the process-driven modelling system that will generate the required data sets used to train the MLEs. As for future studies, the choice of an ML algorithm for developing emulators is not a clear-cut decision. Certainly, the robustness and accuracy of the RF algorithm should not be undervalued, even with a higher computational cost than the ANN and MARS algorithms. As this is a rapidly changing field, newer RF based algorithms, such as Gradient Boosted Decision Trees (GBDT) or XGBoost decision trees would be well worth investigating. The trends in development are almost always towards greater efficiencies. Also, the trend to utilise additional coding libraries which are compatible with multiple compliant ML models, may provide additional attractive features for models, such as the generation of surface response curves for visualising input parameter interactions, as was demonstrated for the MARS model in Chapter 3. As no single ML algorithm is the most suitable to solve all problems, assessment at the time of solving a problem as to what options are available and which approach is best, is a requirement for the foreseeable future.

In Chapter 5, I addressed the third research question: Does performing a Morris or a Sobol SA on an MLE yield the same or comparable results as preforming the SA on the original process-driven model? This research question was answered in the negative, with the SA results reflecting the input/output mappings that were occurring in the MLE, rather than those of the original process-driven model. While these results were similar and not inconsistent between MLEs, it is not possible to make any justified assumptions about these results and what is produced by a SA of the original process-driven model. A positive outcome of Chapter 5's research, however, was the demonstration of just how computationally efficient MLEs are, even the relatively computationally expensive RF emulators. This observation, along with the demonstrated potential to generate highly accurate MLEs for specific purposes, as shown by the results of Chapter 4, opens the possibility of undertaking other forms of SA using simulation experiments which require numbers of simulations that have previously been beyond the scope of most research budgets. Also, data sources that are difficult or impossible to include as data sources for process-driven models may, in some cases, be readily included as input data for ML models. Combining both the process-driven model and additional features that could

be added to an MLE opens the possibility of creating hybrid simulation models. These possibilities present some very exciting options for further research in the area of agro-ecological modelling.

REFERENCES

Acharya, N, Shrivastava, NA, Panigrahi, BK & Mohanty, UC 2013, 'Development of an artificial neural network based multi-model ensemble to estimate the northeast monsoon rainfall over south peninsular India: an application of extreme learning machine', *Climate Dynamics*, vol. 43, no. 5-6, pp. 1303-10.

Ali, I, Cawkwell, F, Dwyer, E & Green, S 2016, 'Modeling managed grassland biomass estimation by using multitemporal remote sensing data—A machine learning approach', *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 7, pp. 3254-64.

Ali, I, Greifeneder, F, Stamenkovic, J, Neumann, M & Notarnicola, C 2015, 'Review of machine learning approaches for biomass and soil moisture retrievals from remote sensing data', *Remote Sensing*, vol. 7, no. 12, pp. 16398-421.

Allen, RG, Pereira, LS, Raes, D & Smith, M 1998, 'Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56', *Fao, Rome*, vol. 300, no. 9, p. D05109.

Ascough II, JC, Green, TR, Ma, L & Ahjua, LR 2004, *Key Criteria and Selection of Sensitivity Analysis Methods Applied to Natural Resource Models*, USDA-ARS.

Asseng, S, Ewert, F, Rosenzweig, C, Jones, JW, Hatfield, JL, Ruane, AC, Boote, KJ, Thorburn, PJ, Rotter, RP, Cammarano, D, Brisson, N, Basso, B, Martre, P, Aggarwal, PK, Angulo, C, Bertuzzi, P, Biernath, C, Challinor, AJ, Doltra, J, Gayler, S, Goldberg, R, Grant, R, Heng, L, Hooker, J, Hunt, LA, Ingwersen, J, Izaurralde, RC, Kersebaum, KC, Muller, C, Naresh Kumar, S, Nendel, C, O/'Leary, G, Olesen, JE, Osborne, TM, Palosuo, T, Priesack, E, Ripoche, D, Semenov, MA, Shcherbak, I, Steduto, P, Stockle, C, Stratonovitch, P, Streck, T, Supit, I, Tao, F, Travasso, M, Waha, K, Wallach, D, White, JW, Williams,

JR & Wolf, J 2013, 'Uncertainty in simulating wheat yields under climate change', *Nature Climate Change*, vol. 3, no. 9, pp. 827-32.

Bachoc, F, Helbert, C & Picheny, V 2020, 'Gaussian process optimization with failures: classification and convergence proof', *Journal of Global Optimization*, vol. 78, no. 3, pp. 483-506.

Balakrishnan, N & Muthukumarasamy, G 2016, 'Crop Production-Ensemble Machine Learning Model for Prediction', *International Journal of Computer Science and Software Engineering (IJCSSE)*, vol. 5, no. 7, pp. 148-53.

Behmann, J, Mahlein, A-K, Rumpf, T, Römer, C & Plümer, L 2015, 'A review of advanced machine learning methods for the detection of biotic stress in precision crop protection', *Precision Agriculture*, vol. 16, no. 3, pp. 239-60.

Belgiu, M & Drăguţ, L 2016, 'Random forest in remote sensing: A review of applications and future directions', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 24-31.

Bellocchi, G, Rivington, M, Donatelli, M & Matthews, K 2010, 'Validation of biophysical models: issues and methodologies. A review', *Agronomy for Sustainable Development*, vol. 30, no. 1, pp. 109-30.

Biau, G & Scornet, E 2016, 'A random forest guided tour', *TEST*, vol. 25, no. 2, pp. 197-227.

Bocca, FF & Rodrigues, LHA 2016, 'The effect of tuning, feature engineering, and feature selection in data mining applied to rainfed sugarcane yield modelling', *Computers and Electronics in Agriculture*, vol. 128, pp. 67-76.

Boehmke, B & Greenwell, BM 2019, *Hands-On Machine Learning with R*, CRC Press.

Breiman, L 2001, 'Random forests', *Machine learning*, vol. 45, no. 1, pp. 5-32.

Brennan, LE, Lisson, SN, Poulton, PL, Carberry, PS, Bristow, KL & Khan, S 2008, 'A farm-scale, bio-economic model for assessing investments in recycled water for irrigation', *Australian Journal of Agricultural Research*, vol. 59, no. 11, pp. 1035-48.

Brisson, N, Gary, C, Justes, E, Roche, R, Mary, B, Ripoche, D, Zimmer, D, Sierra, J, Bertuzzi, P, Burger, P, Bussière, F, Cabidoche, YM, Cellier, P, Debaeke, P, Gaudillère, JP, Hénault, C, Maraux, F, Seguin, B & Sinoquet, H 2003, 'An overview of the crop model stics', *European Journal of Agronomy*, vol. 18, no. 3, pp. 309-32.

Brown, HE, Huth, NI, Holzworth, DP, Teixeira, EI, Zyskowski, RF, Hargreaves, JNG & Moot, DJ 2014, 'Plant Modelling Framework: Software for building and running crop models on the APSIM platform', *Environmental Modelling & Software*, vol. 62, pp. 385-98.

Buchanan, B & Miller, T 2017, *Machine Learning for Policymakers - What it is and why it matters*, The Cyber Security Project, TCS Project, et al., President and Fellows of Harvard College, Harvard Kennedy School, Cambridge, MA 02138, <https://www.belfercenter.org/Cyber/>.

Campolongo, F & Braddock, R 1999, 'The use of graph theory in the sensitivity analysis of the model output: a second order screening method', *Reliability Engineering & System Safety*, vol. 64, no. 1, pp. 1-12.

Campolongo, F, Cariboni, J & Saltelli, A 2007, 'An effective screening design for sensitivity analysis of large models', *Environmental Modelling & Software*, vol. 22, no. 10, pp. 1509-18.

Carberry, PS, Hochman, Z, McCown, RL, Dalgliesh, NP, Foale, MA, Poulton, PL, Hargreaves, JNG, Hargreaves, DMG, Cawthray, S, Hillcoat, N & Robertson, MJ 2002, 'The FARMSCAPE approach to decision support: farmers', advisers', researchers' monitoring, simulation, communication and performance evaluation', *Agricultural Systems*, vol. 74, no. 1, pp. 141-77.

Casadebaig, P, Zheng, B, Chapman, S, Huth, N, Faivre, R & Chenu, K 2016, 'Assessment of the Potential Impacts of Wheat Plant Traits across Environments by Combining Crop Modeling and Global Sensitivity Analysis', *PloS one*, vol. 11, no. 1, p. e0146385.

Castelletti, A, Galelli, S, Ratto, M, Soncini-Sessa, R & Young, PC 2012, 'A general framework for Dynamic Emulation Modelling in environmental problems', *Environmental Modelling & Software*, vol. 34, pp. 5-18.

Chekole, A 2019, 'Application of Machine Learning Tools for Predicting Determinant Factors', *International Journal of Advanced Research in Computer Science*, vol. 10, no. 4, pp. 40-4.

Chen, C, Berger, J, Fletcher, A, Lawes, R & Robertson, M 2016, 'Genotype × environment interactions for phenological adaptation in narrow-leafed lupin: A simulation study with a parameter optimized model', *Field Crops Research*, vol. 197, pp. 28-38.

Chowdhury, S, Verma, B & Stockwell, D 2015, 'A novel texture feature based multiple classifier technique for roadside vegetation classification', *Expert Systems with Applications*, vol. 42, no. 12, pp. 5047-55.

Christopher Frey, H & Patil, SR 2002, 'Identification and Review of Sensitivity Analysis Methods', *Risk Analysis*, vol. 22, no. 3, pp. 553-78.

Cravero, A & Sepúlveda, S 2021, 'Use and Adaptations of Machine Learning in Big Data—Applications in Real Cases in Agriculture', *Electronics*, vol. 10, no. 5, p. 552.

Dahms, T, Seissiger, S, Conrad, C & Borg, E 2016, 'Modelling biophysical parameters of maize using LandSat 8 times series', *XXIII ISPRS Congress*: *Proceedings of the XXIII ISPRS Congress* The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Prague, Czech Republic.

Dayal, K, Weaver, T, Bange, M & CSD Ltd. Extension & Development Team 2019, 'Using machine learning to sharpen agronomic insights to improve decision making in Australian cotton systems ', *19th Australian Society of Agronomy Conference*: *Proceedings of the 19th Australian Society of Agronomy Conference*, J Pratley (ed.), Wagga Wagga New South Wales.

DeJonge, KC, Ascough II, JC, Ahmadi, M, Andales, AA & Arabi, M 2012, 'Global sensitivity and uncertainty analysis of a dynamic agroecosystem model under different irrigation treatments', *Ecological Modelling*, vol. 231, pp. 113-25.

Deo, RC & Şahin, M 2015a, 'Application of the extreme learning machine algorithm for the prediction of monthly Effective Drought Index in eastern Australia', *Atmospheric Research*, vol. 153, pp. 512-25.

Deo, RC & Şahin, M 2015b, 'Application of the Artificial Neural Network model for prediction of monthly Standardized Precipitation and Evapotranspiration Index using hydrometeorological parameters and climate indices in eastern Australia', *Atmospheric Research*, vol. 161, pp. 65-81.

Deo, RC, Samui, P & Kim, D 2015, 'Estimation of monthly evaporative loss using relevance vector machine, extreme learning machine and multivariate adaptive regression spline models', *Stochastic Environmental Research and Risk Assessment*, vol. 30, no. 6, pp. 1769-84.

Deo, RC, Tiwari, MK, Adamowski, JF & Quilty, JM 2017, 'Forecasting effective drought index using a wavelet extreme learning machine (W-ELM) model', *Stochastic Environmental Research and Risk Assessment*, vol. 31, no. 5, pp. 1211-40.

Dokic, K, Blaskovic, L & Mandusic, D 2020, 'From machine learning to deep learning in agriculture – the quantitative review of trends', *IOP Conference Series: Earth and Environmental Science*, vol. 614, p. 012138.

Donatelli, M, Van Ittersum, MK, Bindi, M & Porter, JR 2002, 'Modelling cropping systems—highlights of the symposium and preface to the special issues', *European Journal of Agronomy*, vol. 18, no. 1, pp. 1-11.

Dumancas, GG & Bello, GA 2015, 'Comparison of machine-learning techniques for handling multicollinearity in big data analytics and high-performance data mining', *SC15: The International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 41-2.

Fajardo, M & Whelan, BM 2021, 'Within-farm wheat yield forecasting incorporating off-farm information', *Precision Agriculture*, vol. 22, no. 2, pp. 569-85.

Feng, P, Wang, B, Liu, DL, Waters, C & Yu, Q 2019, 'Incorporating machine learning with biophysical model can improve the evaluation of climate extremes impacts on wheat yield in south-eastern Australia', *Agricultural and Forest Meteorology*, vol. 275, pp. 100-13.

Feng, P, Wang, B, Liu, DL, Waters, C, Xiao, D, Shi, L & Yu, Q 2020, 'Dynamic wheat yield forecasts are improved by a hybrid approach using a biophysical model and machine learning technique', *Agricultural and Forest Meteorology*, vol. 285, p. 107922.

Fortin, JG, Morais, A, Anctil, F & Parent, LE 2014, 'Comparison of Machine Learning Regression Methods to Simulate NO3 Flux in Soil Solution under Potato Crops', *Applied Mathematics*, vol. 5, no. 5, pp. 832-41.

Friedman, JH 1991a, 'Multivariate Adaptive Regression Splines', *The Annals of Statistics*, vol. 19, no. 1, pp. 1-67.

Friedman, JH 1991b, *Estimating functions of mixed ordinal and categorical variables using adaptive splines.*, Technical Report No. 108, DoS-S University, Laboratory for Computational Statistics, Department of Statistics, Stanford University.

Friedman, JH & Roosen, CB 1995, 'An introduction to multivariate adaptive regression splines', *Statistical Methods in Medical Research*, vol. 4, no. 3, pp. 197-217.

Gebauer, A, Brito Gómez, VM & Ließ, M 2019, 'Optimisation in machine learning: An application to topsoil organic stocks prediction in a dry forest ecosystem', *Geoderma*, vol. 354, p. 113846.

Ghimire, S, Deo, RC, Downs, NJ & Raj, N 2018, 'Self-adaptive differential evolutionary extreme learning machines for long-term solar radiation prediction with remotely-sensed MODIS satellite and Reanalysis atmospheric products in solar-rich cities', *Remote Sensing of Environment*, vol. 212, pp. 176-98.

Gilbertson, J & Niekerk, A 2016, 'Value of feature reduction for crop differentiation using multi-temporal imagery, machine learning, and object-based image analysis', *GEOBIA 2016 : Solutions and Synergies,*: *Proceedings of the GEOBIA 2016 : Solutions and Synergies,* University of Twente Faculty of Geo-Information and Earth Observation (ITC).

Gollapudi, S 2016, *Practical Machine Learning*, Packt Publishing Ltd, 2016, Livery Place, 35 Livery Street, Burmingham B3 2PB, UK.

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

Guo, Y, Fu, Y, Hao, F, Zhang, X, Wu, W, Jin, X, Robin Bryant, C & Senthilnath, J 2021, 'Integrated phenology and climate in rice yields prediction using machine learning methods', *Ecological Indicators*, vol. 120, p. 106935.

Hieronymi, A 2013, 'Understanding Systems Science: A Visual and Integrative Approach', *Systems Research and Behavioral Science*, vol. 30, no. 5, pp. 580-95.

Hochman, Z, Gobbett, D, Horan, H & Navarro Garcia, J 2016, 'Data rich yield gap analysis of wheat in Australia', *Field Crops Research*, vol. 197, pp. 97-106.

Holzworth, D, Huth, NI, Fainges, J, Brown, H, Zurcher, E, Cichota, R, Verrall, S, Herrmann, NI, Zheng, B & Snow, V 2018, 'APSIM Next Generation: Overcoming challenges in modernising a farming systems model', *Environmental Modelling & Software*, vol. 103, pp. 43-51.

Holzworth, DP, Snow, V, Janssen, S, Athanasiadis, IN, Donatelli, M, Hoogenboom, G, White, JW & Thorburn, P 2015, 'Agricultural production systems modelling and software: Current status and future prospects', *Environmental Modelling & Software*, vol. 72, pp. 276-86.

Holzworth, DP, Huth, NI, deVoil, PG, Zurcher, EJ, Herrmann, NI, McLean, G, Chenu, K, van Oosterom, EJ, Snow, V, Murphy, C, Moore, AD, Brown, H, Whish, JPM, Verrall, S, Fainges, J, Bell, LW, Peake, AS, Poulton, PL, Hochman, Z, Thorburn, PJ, Gaydon, DS, Dalgliesh, NP, Rodriguez, D, Cox, H, Chapman, S, Doherty, A, Teixeira, E, Sharp, J, Cichota, R, Vogeler, I, Li, FY, Wang, E, Hammer, GL, Robertson, MJ, Dimes, JP, Whitbread, AM, Hunt, J, van Rees, H, McClelland, T, Carberry, PS, Hargreaves, JNG, MacLeod, N, McDonald, C, Harsdorf, J, Wedgwood, S & Keating, BA 2014, 'APSIM – Evolution towards a new generation of agricultural systems simulation', *Environmental Modelling & Software*, vol. 62, pp. 327-50.

Hussein, EA, Thron, C, Ghaziasgar, M, Bagula, A & Vaccari, M 2020, 'Groundwater Prediction Using Machine-Learning Tools', *Algorithms*, vol. 13, no. 11.

Huth, N & Holzworth, D 2005, 'Common sense in model testing', *MODSIM 2005 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand*, pp. 170-6.

Iooss, B & Lemaître, P 2015, 'A review on global sensitivity analysis methods', *Operations Research/ Computer Science Interfaces Series*, vol. 59, pp. 101-22.

Iooss, B, Janon, A, Pujol, G, Broto, B, Boumhaout, K, Da Veiga, S, Delage, T, El Amri, R, Fruth, J, Gilquin, L, Guillaume, JHA, Le Gratiet, L, Lemaitre, P,

Marrel, A, Meynaoui, A, Nelson, BL, Monari, F, Oomen, R, Rakovec, O, Ramos, B, Roustant, O, Song, E, Staum, J, Sueur, R, Touati, T & Weber , F 2020, *sensitivity: Global Sensitivity Analysis of Model Outputs*, 1.18.1, Cran R Project, <https://cran.r-project.org/web/packages/sensitivity>.

Janssen, SJC, Porter, CH, Moore, AD, Athanasiadis, IN, Foster, I, Jones, JW & Antle, JM 2017, 'Towards a new generation of agricultural system data, models and knowledge products: Information and communication technology', *Agricultural Systems*, vol. 155, pp. 200-12.

Jaxa-Rozen, M & Kwakkel, J 2018, 'Tree-based ensemble methods for sensitivity analysis of environmental models: A performance comparison with Sobol and Morris techniques', *Environmental Modelling & Software*, vol. 107, pp. 245-66.

Jekabsons, G 2016, *ARESLab : Adaptive Regression Splines toolbox for Matlab/Octave.*, Available at http://www.cs.rtu.lv/jekabsons/.

Jeong, JH, Resop, JP, Mueller, ND, Fleisher, DH, Yun, K, Butler, EE, Timlin, DJ, Shim, K-M, Gerber, JS & Reddy, VR 2016, 'Random forests for global and regional crop yield predictions', *PloS one*, vol. 11, no. 6, p. e0156571.

Jones, JW, Hoogenboom, G, Porter, CH, Boote, KJ, Batchelor, WD, Hunt, LA, Wilkens, PW, Singh, U, Gijsman, AJ & Ritchie, JT 2003, 'The DSSAT cropping system model', *European Journal of Agronomy*, vol. 18, no. 3, pp. 235-65.

Jones, JW, Antle, JM, Basso, B, Boote, KJ, Conant, RT, Foster, I, Godfray, HCJ, Herrero, M, Howitt, RE, Janssen, S, Keating, BA, Munoz-Carpena, R, Porter, CH, Rosenzweig, C & Wheeler, TR 2016, 'Brief history of agricultural systems modeling', *Agricultural Systems*, vol. 155, pp. 240-54.

Jordan, MI & Mitchell, TM 2015, 'Machine learning: Trends, perspectives, and prospects', *Science*, vol. 349, no. 6245, pp. 255-60.

Kaelbling, LP, Littman, ML & Moore, AW 1996, 'Reinforcement Learning: A Survey', *Journal of Artificial Intelligence Research*, vol. 4, pp. 237-85.

Karandish, F & Šimůnek, J 2016, 'A comparison of numerical and machine-learning modeling of soil water content with limited input data', *Journal of Hydrology*, vol. 543, pp. 892-909.

Keating, BA, Carberry, PS, Hammer, GL, Probert, ME, Robertson, MJ, Holzworth, D, Huth, NI, Hargreaves, JNG, Meinke, H, Hochman, Z, McLean, G, Verburg, K, Snow, V, Dimes, JP, Silburn, M, Wang, E, Brown, S, Bristow, KL, Asseng, S, Chapman, S, McCown, RL, Freebairn, DM & Smith, CJ 2003, 'An overview of APSIM, a model designed for farming systems simulation', *European Journal of Agronomy*, vol. 18, no. 3, pp. 267-88.

Kersebaum, KC, Boote, KJ, Jorgenson, JS, Nendel, C, Bindi, M, Frühauf, C, Gaiser, T, Hoogenboom, G, Kollas, C, Olesen, JE, Rötter, RP, Ruget, F, Thorburn, PJ, Trnka, M & Wegehenkel, M 2015, 'Analysis and classification of data sets for calibration and validation of agro-ecosystem models', *Environmental Modelling & Software*, vol. 72, pp. 402-17.

King, DM & Perera, BJC 2013, 'Morris method of sensitivity analysis applied to assess the importance of input variables on urban water supply yield – A case study', *Journal of Hydrology*, vol. 477, pp. 17-32.

Kouadio, L, Deo, RC, Byrareddy, V, Adamowski, JF, Mushtaq, S & Phuong Nguyen, V 2018, 'Artificial intelligence approach for the prediction of Robusta coffee yield using soil fertility properties', *Computers and Electronics in Agriculture*, vol. 155, pp. 324-38.

Kursa, MB & Rudnicki, WR 2010, 'Feature Selection with the Boruta Package', *Journal of Statistical Software*, vol. 36, pp. 1-13.

Kuwata, K & Shibasaki, R 2016, 'Estimating Corn Yield In The United States With Modis Evi And Machine Learning Methods', *ISPRS Annals of*

Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 3, no. 8, pp. 131-6.

Lawes, RA, Oliver, YM & Huth, NI 2019, 'Optimal Nitrogen Rate Can Be Predicted Using Average Yield and Estimates of Soil Water and Leaf Nitrogen with Infield Experimentation', *Agronomy Journal*, vol. 111, no. 3, pp. 1155-64.

Legates, DR & Davis, RE 1997, 'The continuing search for an anthropogenic climate change signal: Limitations of correlation-based approaches', *Geophysical Research Letters*, vol. 24, no. 18, pp. 2319-22.

Legates, DR & McCabe Jr, GJ 1999, 'Evaluating the use of 'goodness-of-fit' measures in hydrologic and hydroclimatic model validation', *Water Resources Research*, vol. 35, no. 1, pp. 233-41.

Leng, G & Hall, JW 2020, 'Predicting spatial and temporal variability in crop yields: an inter-comparison of machine learning, regression and process-based models', *Environmental Research Letters*, vol. 15, no. 4, p. 044027.

Li, J, Tran, M & Siwabessy, J 2016, 'Selecting Optimal Random Forest Predictive Models: A Case Study on Predicting the Spatial Distribution of Seabed Hardness', *PloS one*, vol. 11, no. 2, p. e0149089.

Liaw, A & Wiener, M 2018, *Random Forests for Classification and Regression - Breiman and Cutler's implementation*, 4.6-14, R, <https://www.stat.berkeley.edu/~breiman/RandomForests/>.

Lippmann, R 1987, 'An introduction to computing with neural nets', *IEEE Assp magazine*, vol. 4, no. 2, pp. 4-22.

Liu, S, Mcgree, J, Ge, Z & Xie, Y 2015, *Computational and Statistical Methods for Analysing Big Data with Applications*, Elsevier Science.

Liu, Z & Choe, Y 2021, 'Data-driven sensitivity indices for models with dependent inputs using polynomial chaos expansions', *Structural Safety*, vol. 88, p. 101984.

Luo, Q, Bange, M, Braunack, M & Johnston, D 2016, 'Effectiveness of agronomic practices in dealing with climate change impacts in the Australian cotton industry — A simulation study', *Agricultural Systems*, vol. 147, pp. 1-9.

Ma, C, Zhang, HH & Wang, X 2014, 'Machine learning for Big Data analytics in plants', *Trends in Plant Science*, vol. 19, no. 12, pp. 798-808.

McCown, RL, Hammer, GL, Hargreaves, JNG, Holzworth, D & Huth, NI 1995, 'APSIM: an agricultural production system simulation model for operational research', *Mathematics and Computers in Simulation*, vol. 39, no. 3, pp. 225-31.

Milborrow, S 2019, *CRAN - R Package 'earth': Multivariate Adaptive Regression Splines*, 5.1.2, <available at http://www.milbo.users.sonic.net/earth>.

Milborrow, S 2020, *earth: Multivariate Adaptive Regression Splines. Derived from mda:mars by Trevor Hastie and Rob Tibshirani. Uses Alan Miller's Fortran utilities with Thomas Lumley's leaps wrapper.*, R package version 5.3.0, <https://CRAN.R-project.org/package=earth>.

Molnar, C, Konig, G, Herbinger, J, Freiesleben, T, Dandl, S, Scholbeck, CA, Casalicchio, G, Grosse-Wentrup, M & Bischl, B 2020, 'Pitfalls to Avoid when Interpreting Machine Learning Models', *arXiv*.

Montesino-San Martin, M, Wallach, D, Olesen, JE, Challinor, AJ, Hoffman, MP, Koehler, AK, Rötter, RP & Porter, JR 2018, 'Data requirements for crop modelling—Applying the learning curve approach to the simulation of winter wheat flowering time under climate change', *European Journal of Agronomy*, vol. 95, pp. 33-44.

Morris, MD 1991, 'Factorial Sampling Plans for Preliminary Computational Experiments', *Technometrics*, vol. 33, no. 2, pp. 161-74.

Nettleton, DF, Katsantonis, D, Kalaitzidis, A, Sarafijanovic-Djukic, N, Puigdollers, P & Confalonieri, R 2019, 'Predicting rice blast disease: machine learning versus process-based models', *BMC Bioinformatics*, vol. 20, no. 1, p. 514.

Newlands, NK, Zamar, DS, Kouadio, LA, Zhang, Y, Chipanshi, A, Potgieter, A, Toure, S & Hill, HSJ 2014, 'An integrated, probabilistic model for improved seasonal forecasting of agricultural crop yield under environmental uncertainty', *Frontiers in Environmental Science*, vol. 2, pp. 17-35.

Niazian, M & Niedbała, G 2020, 'Machine Learning for Plant Breeding and Biotechnology', *Agriculture*, vol. 10, no. 10, p. 436.

Norton, J 2015, 'An introduction to sensitivity assessment of simulation models', *Environmental Modelling & Software*, vol. 69, pp. 166-74.

Obsie, EY, Qu, H & Drummond, F 2020, 'Wild blueberry yield prediction using a combination of computer simulation and machine learning algorithms', *Computers and Electronics in Agriculture*, vol. 178, p. 105778.

Pantazi, XE, Moshou, D, Alexandridis, T, Whetton, R & Mouazen, AM 2016, 'Wheat yield prediction using machine learning and advanced sensing techniques', *Computers and Electronics in Agriculture*, vol. 121, pp. 57-65.

Pardon, L, Huth, NI, Nelson, PN, Banabas, M, Gabrielle, B & Bessou, C 2017, 'Yield and nitrogen losses in oil palm plantations: Main drivers and management trade-offs determined using simulation', *Field Crops Research*, vol. 210, pp. 20-32.

Paudel, D, Boogaard, H, de Wit, A, Janssen, S, Osinga, S, Pylianidis, C & Athanasiadis, IN 2021, 'Machine learning for large-scale crop yield forecasting', *Agricultural Systems*, vol. 187, p. 103016.

Peake, AS, Huth, NI, Carberry, PS, Raine, SR & Smith, RJ 2014, 'Quantifying potential yield and lodging-related yield gaps for irrigated spring wheat in sub-tropical Australia', *Field Crops Research*, vol. 158, pp. 1-14.

Pembleton, KG, Cullen, BR, Rawnsley, RP, Harrison, MT & Ramilan, T 2016, 'Modelling the resilience of forage crop production to future climate change in the dairy regions of Southeastern Australia using APSIM', *The Journal of Agricultural Science*, vol. 154, no. 7, pp. 1131-52.

Phelan, DC, Harrison, MT, McLean, G, Cox, H, Pembleton, KG, Dean, GJ, Parsons, D, do Amaral Richter, ME, Pengilley, G & Hinton, SJ 2018, 'Advancing a farmer decision support tool for agronomic decisions on rainfed and irrigated wheat cropping in Tasmania', *Agricultural Systems*, vol. 167, pp. 113-24.

Pianosi, F, Beven, K, Freer, J, Hall, JW, Rougier, J, Stephenson, DB & Wagener, T 2016, 'Sensitivity analysis of environmental models: A systematic review with practical workflow', *Environmental Modelling & Software*, vol. 79, pp. 214-32.

Plischke, E, Borgonovo, E & Smith, CL 2013, 'Global sensitivity measures from given data', *European Journal of Operational Research*, vol. 226, no. 3, pp. 536-50.

Prasad, R, Deo, RC, Li, Y & Maraseni, T 2017, 'Input selection and performance optimization of ANN-based streamflow forecasts in the drought-prone Murray Darling Basin region using IIS and MODWT algorithm', *Atmospheric Research*, vol. 197, pp. 42-63.

Prasad, R, Deo, RC, Li, Y & Maraseni, T 2018, 'Soil moisture forecasting by a hybrid machine learning technique: ELM integrated with ensemble empirical mode decomposition', *Geoderma*, vol. 330, pp. 136-61.

Priestley, CHB & Taylor, RJ 1972, 'On the assessment of surface heat flux and evaporation using large-scale parameters', *Monthly weather review*, vol. 100, no. 2, pp. 81-92.

Qureshi, ME, Whitten, SM, Mainuddin, M, Marvanek, S & Elmahdi, A 2013, 'A biodphysical and economic model of agriculture and water in the Murray-Darling Basing, Australia', *Environmental Modelling & Software*, vol. 41, pp. 98-106.

R Core Team 2020, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, <https://www.R-project.org>.

R Core Team 2021, *R: A language and environment for statistical computing.*, R Foundation for Statistical Computing, Vienna, Austria, <https://www.R-project.org/>.

Raghavendra. N, S & Deka, PC 2014, 'Support vector machine applications in the field of hydrology: A review', *Applied Soft Computing*, vol. 19, pp. 372-86.

Ramstein, GP, Jensen, SE & Buckler, ES 2019, 'Breaking the curse of dimensionality to identify causal variants in Breeding 4', *Theoretical and Applied Genetics*, vol. 132, no. 3, pp. 559-67.

Ratto, M, Pagano, A & Young, P 2007, 'State Dependent Parameter metamodelling and sensitivity analysis', *Computer Physics Communications*, vol. 177, no. 11, pp. 863-76.

Ratto, M, Castelletti, A & Pagano, A 2012, 'Emulation techniques for the reduction and sensitivity analysis of complex environmental models', *Environmental Modelling & Software*, vol. 34, pp. 1-4.

Razavi, S & Gupta, HV 2015, 'What do we mean by sensitivity analysis? the need for comprehensive characterization of "global" sensitivity in Earth and

Environmental systems models', *Water Resources Research*, vol. 51, no. 5, pp. 3070-92.

Razavi, S, Tolson, BA & Burn, DH 2012, 'Review of surrogate modeling in water resources', *Water Resources Research*, vol. 48, no. 7, p. W07401.

Razavi, S, Jakeman, A, Saltelli, A, Prieur, C, Iooss, B, Borgonovo, E, Plischke, E, Lo Piano, S, Iwanaga, T, Becker, W, Tarantola, S, Guillaume, JHA, Jakeman, J, Gupta, H, Melillo, N, Rabitti, G, Chabridon, V, Duan, Q, Sun, X, Smith, S, Sheikholeslami, R, Hosseini, N, Asadzadeh, M, Puy, A, Kucherenko, S & Maier, HR 2021, 'The Future of Sensitivity Analysis: An essential discipline for systems modeling and policy support', *Environmental Modelling & Software*, vol. 137, p. 104954.

Robertson, MJ, Rebetzke, GJ & Norton, RM 2015, 'Assessing the place and role of crop simulation modelling in Australia', *Crop and Pasture Science*, vol. 66, no. 9, pp. 877-93.

Robertson, MJ, Carberry, PS, Huth, NI, Turpin, JE, Probert, ME, Poulton, PL, Bell, M, Wright, GC, Yeates, SJ & Brinsmead, RB 2002, 'Simulation of growth and development of diverse legume species in APSIM', *Australian Journal of Agricultural Research*, vol. 53, no. 4, pp. 429-46.

RStudio Team 2021, *RStudio: Integrated Development Environment for R*, RStudio, PBC Boston, MA <http://www.rstudio.com/>.

Rudy, JC 2017, *A Python implementation of Jerome Friedman's Multivariate Adaptive Regression Splines* https://github.com/scikit-learn-contrib/py-earth, <https://github.com/scikit-learn-contrib/py-earth>.

Ryan, E, Wild, O, Voulgarakis, A & Lee, L 2018, 'Fast sensitivity analysis methods for computationally expensive models with multi-dimensional output', *Geoscientific Model Development*, vol. 11, no. 8, pp. 3131-46.

Şahin, M, Kaya, Y, Uyar, M & Yıldırım, S 2014, 'Application of extreme learning machine for estimating solar radiation from satellite data', *International Journal of Energy Research*, vol. 38, no. 2, pp. 205-12.

Saiaa, SM, Nelsona, N, Husethb, AS, Griegerc, K & Reich, BJ 2020, 'Transitioning Machine Learning from Theory to Practice in Natural Resources Management', *Ecological Modelling*, vol. 435, p. 109257.

Salcedo-Sanz, S, Pastor-Sánchez, A, Prieto, L, Blanco-Aguilera, A & García-Herrera, R 2014, 'Feature selection in wind speed prediction systems based on a hybrid coral reefs optimization – Extreme learning machine approach', *Energy Conversion and Management*, vol. 87, pp. 10-8.

Saltelli, A & Annoni, P 2010, 'How to avoid a perfunctory sensitivity analysis', *Environmental Modelling & Software*, vol. 25, no. 12, pp. 1508-17.

Saltelli, A, Tarantola, S & Campolongo, F 2000, 'Sensitivity Anaysis as an Ingredient of Modeling', *Statistical Science*, vol. 15, no. 4, pp. 377-95.

Samarasinghe, S 2016, *Neural networks for applied sciences and engineering: from fundamentals to complex pattern recognition*, Auerbach publications.

Sanikhani, H, Deo, RC, Yaseen, ZM, Eray, O & Kisi, O 2018, 'Non-tuned data intelligent model for soil temperature estimation: A new approach', *Geoderma*, vol. 330, pp. 52-64.

Sarrazin, F, Pianosi, F & Wagener, T 2016, 'Global Sensitivity Analysis of environmental models: Convergence and validation', *Environmental Modelling & Software*, vol. 79, pp. 135-52.

Sexton, J & Laake, P 2009, 'Standard errors for bagged and random forest estimators', *Computational Statistics & Data Analysis*, vol. 53, no. 3, pp. 801-11.

Sexton, J, Everingham, YL & Inman-Bamber, G 2017, 'A global sensitivity analysis of cultivar trait parameters in a sugarcane growth model for contrasting production environments in Queensland, Australia', *European Journal of Agronomy*, vol. 88, pp. 96-105.

Shahhosseini, M, Hu, G, Archontoulis, SV & Huber, I 2021, 'Coupling Machine Learning and Crop Modeling Improves Crop Yield Prediction in the US Corn Belt', *Scientific Reports* vol. 11 no. 1, pp. 1-15.

Shakoor, MT, Rahman, K, Rayta, SN & Chakrabarty, A 2017, 'Agricultural production output prediction using Supervised Machine Learning techniques', *2017 1st International Conference on Next Generation Computing Applications (NextComp)*, pp. 182-7, <http://ieeexplore.ieee.org/document/8016196/>.

Shastry, KA, Sanjay, HA & Deshmukh, A 2016, 'A Parameter Based Customized Artificial Neural Network Model for Crop Yield Prediction.', *Journal of Artificial Intelligence*, vol. 9, pp. 23-32.

Shin, M-J, Guillaume, JHA, Croke, BFW & Jakeman, AJ 2015, 'A review of foundational methods for checking the structural identifiability of models: Results for rainfall-runoff', *Journal of Hydrology*, vol. 520, pp. 1-16.

Shukr, HH, Pembleton, KG, Zull, AF & Cockfield, GJ 2021, 'Impacts of Effects of Deficit Irrigation Strategy on Water Use Efficiency and Yield in Cotton under Different Irrigation Systems', *Agronomy*, vol. 11, no. 2, p. 231.

Singh, V, Sarwar, A & Sharma, V 2017, 'Analysis of soil and prediction of crop yield (Rice) using Machine Learning approach.', *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, p. 1254.

Sobol', IM 1993, 'Sensitivity analysis for non-linear mathematical models', *Mathematical Modelling and Computational Experiments*, vol. 1, no. 4, pp. 407-14.

Sobol', IM 2001, 'Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates', *Mathematics and Computers in Simulation*, vol. 55, no. 1, pp. 271-80.

Stanfill, B, Mielenz, H, Clifford, D & Thorburn, P 2015, 'Simple approach to emulating complex computer models for global sensitivity analysis', *Environmental Modelling & Software*, vol. 74, pp. 140-55.

Stas, M, Van Orshoven, J, Dong, Q, Heremans, S & Zhang, B 2016, 'A comparison of machine learning algorithms for regional wheat yield prediction using NDVI time series of SPOT-VGT', *Agro-Geoinformatics (Agro-Geoinformatics), 2016 Fifth International Conference on*, IEEE, pp. 1-5.

Stöckle, CO, Martin, SA & Campbell, GS 1994, 'CropSyst, a cropping systems simulation model: water/nitrogen budgets and crop yield', *Agricultural Systems*, vol. 46, no. 3, pp. 335-59.

Stöckle, CO, Donatelli, M & Nelson, R 2003, 'CropSyst, a cropping systems simulation model', *European Journal of Agronomy*, vol. 18, no. 3, pp. 289-307.

Taormina, R & Chau, K-W 2015, 'Data-driven input variable selection for rainfall–runoff modeling using binary-coded particle swarm optimization and Extreme Learning Machines', *Journal of Hydrology*, vol. 529, pp. 1617-32.

Thessen, A 2016, 'Adoption of Machine Learning Techniques in Ecology and Earth Science', *One Ecosystem*, vol. 1.

Thorp, KR, DeJonge, KC, Marek, GW & Evett, SR 2020, 'Comparison of evapotranspiration methods in the DSSAT Cropping System Model: I. Global sensitivity analysis', *Computers and Electronics in Agriculture*, vol. 177, p. 105658.

Venables, WN & Ripley, BD 2002, *Modern Applied Statistics with S*, Fourth edn, Springer.

Villa-Vialaneix, N, Follador, M, Ratto, M & Leip, A 2012, 'A comparison of eight metamodeling techniques for the simulation of N2O fluxes and N leaching from corn crops', *Environmental Modelling & Software*, vol. 34, pp. 51-66.

Wallach, D & Thorburn, PJ 2017, 'Estimating uncertainty in crop model predictions: Current situation and future prospects', *European Journal of Agronomy*, vol. 88, pp. 1-7.

Wei, P, Lu, Z & Song, J 2015, 'A comprehensive comparison of two variable importance analysis techniques in high dimensions: Application to an environmental multi-indicators system', *Environmental Modelling & Software*, vol. 70, pp. 178-90.

Wickham, H 2016, *ggplot2: Elegant Graphics for Data Analysis.*, Springer-Verlag New York.

Wickham, H, Averick, M, Bryan, J, Chang, W, D'Agostino-McGowan, L, François, R, Grolemund, G, Hayes, A, Henry, L, Hester, J, Kuhn, M, Lin Pedersen, T, Miller, E, Milton Bache, S, Müller, K, Ooms, J, Robinson, D, Paige Seidel, D, Spinu, V, Takahashi, K, Vaughan, D, Wilke, C, Woo, K & Yutani, H 2019, 'Welcome to the tidyverse', *Journal of Open Source Software*, vol. 4, no. 43, p. 1686.

Willmott, CJ 1981, 'On the Validation of Models', *Physical Geography*, vol. 2, no. 2, pp. 184-94.

Willmott, CJ, Ackleson, SG, Davis, RE, Feddema, JJ, Klink, KM, Legates, DR, O'Donnell, J & Rowe, CM 1985, 'Statistics for the evaluation and comparison of models', *Journal of Geophysical Research: Oceans*, vol. 90, no. C5, pp. 8995-9005.

Yang, J 2011, 'Convergence and uncertainty analyses in Monte-Carlo based sensitivity analysis', *Environmental Modelling & Software*, vol. 26, no. 4, pp. 444-57.

Young, PC & Ratto, M 2009, 'A unified approach to environmental systems modeling', *Stochastic Environmental Research and Risk Assessment*, vol. 23, no. 7, pp. 1037-57.

Zhang, J, Chen, Y & Zhang, Z 2020, 'A remote sensing-based scheme to improve regional crop model calibration at sub-model component level', *Agricultural Systems*, vol. 181, p. 102814.

APPENDIX A

Sample listing of MARS model generation log

Log file of build of ARES (MARS) model for output: 'Emergence Days After Sowing'
for Sowing #3 at Warwick, Australia.

```
    'Building the model (T:EmergDAS)
================================================='

Building ARES model...
Approx number of available knot locations (controlled by useMinSpan
and useEndSpan): x1:3 x2:4 x3:4 x4:4 x5:4 x6:4 x7:3
Forward phase
.....................................................
Termination condition is met: R2 improvement is below threshold.
Number of basis functions in the model after forward phase: 56
Backward phase
.....................................................
Number of basis functions in the final model: 40
Total effective number of parameters: 40.0
Highest degree of interactions: 3
Number of input variables in the model: 3 (x3, x5, x6)
Execution time: 38.92 seconds

model =

  <a href="matlab:helpPopup struct" style="font-
weight:bold">struct</a> with fields:

            MSE: 7.8715e-05
            GCV: 7.9576e-05
          coefs: [40×1 double]
       knotdims:
      knotsites: [39×1 cell]
       knotdirs: [39×1 cell]
        parents: [39×1 double]
    trainParams: [1×1 struct]
             t1: [39×7 double]
             t2: [39×7 double]
           minX: [0 0 0 0 0 0 0]
           maxX: [1 1 1 1 1 1 1]
      isBinary: [0 0 0 0 0 0 0]


    'Output Target:EmergDAS'

Info on the basis functions
=======================================
Type: piecewise-cubic
MSE: 7.87152e-05
GCV: 7.95764e-05
```

```
R2GCV: 0.998614
Total number of basis functions (including intercept): 40
Total effective number of parameters: 40
Basis functions:
BF          MSE          GCV        R2GCV           coef    hinges
basis function
0             -            -            -      -0.092418
(intercept)
3       0.00173      0.00175      0.96954       -0.33044    \_
C(x5|-1,0.5,0.667,0.833)
1       0.00132      0.00133      0.97677       -0.31735    \_
C(x3|-1,0.5,0.667,0.833)
2       0.00034      0.00035      0.99398         1.0091    _/
C(x5|+1,0.5,0.667,0.833)
6       0.00025      0.00026      0.99555         2.8202    _/ _/
C(x3|+1,0.5,0.667,0.833) * C(x5|+1,0.167,0.333,0.5)
15      0.00024      0.00025      0.99572        -3.6111    _/ _/
C(x3|+1,0.5,0.667,0.833) * C(x6|+1,0.5,0.667,0.833)
25      0.00020      0.00020      0.99647         10.291    _/ \_
C(x3|+1,0.5,0.667,0.833) * C(x6|-1,0.167,0.333,0.5)
22      0.00017      0.00018      0.99695        -17.908    _/ _/ _/
BF20 * C(x3|+1,0.167,0.333,0.5)
30      0.00016      0.00016      0.99718        -9.3634    \_ \_ _/
C(x5|-1,0.5,0.667,0.833) * C(x6|-1,0.167,0.333,0.5) *
C(x3|+1,0.5,0.667,0.833)
26      0.00016      0.00016      0.99725        -6.5903    _/ \_
C(x3|+1,0.167,0.333,0.5) * C(x6|-1,0.167,0.333,0.5)
37      0.00015      0.00015      0.99739          1.583    \_ _/
BF5 * C(x3|+1,0.167,0.333,0.5)
5       0.00014      0.00014      0.99750         1.5005    \_
C(x6|-1,0.5,0.667,0.833)
39      0.00014      0.00014      0.99751         5.5776    \_ \_ _/
BF38 * max(0, x5 +0)
27      0.00014      0.00014      0.99751        -3.2237    \_ \_
BF1 * C(x6|-1,0.167,0.333,0.5)
33      0.00014      0.00014      0.99752         2.6298    \_ \_ \_
C(x6|-1,0.5,0.667,0.833) * C(x5|-1,0.5,0.667,0.833) * C(x3|-
1,0.5,0.667,0.833)
12      0.00014      0.00014      0.99761        -9.9709    \_ _/ \_
BF9 * C(x3|-1,0.5,0.667,0.833)
16      0.00013      0.00013      0.99766        -3.1265    _/ \_
C(x3|+1,0.5,0.667,0.833) * C(x6|-1,0.5,0.667,0.833)
31      0.00013      0.00013      0.99767        -5.1669    \_ \_ \_
C(x5|-1,0.5,0.667,0.833) * C(x6|-1,0.167,0.333,0.5) * C(x3|-
1,0.5,0.667,0.833)
29      0.00013      0.00013      0.99773         10.933    \_ \_ \_
C(x5|-1,0.5,0.667,0.833) * C(x3|-1,0.167,0.333,0.5) * C(x6|-
1,0.167,0.333,0.5)
32      0.00013      0.00013      0.99779         6.9259    \_ \_ _/
C(x6|-1,0.5,0.667,0.833) * C(x5|-1,0.5,0.667,0.833) *
C(x3|+1,0.5,0.667,0.833)
13      0.00012      0.00013      0.99781        -30.502    _/ \_ _/
C(x5|+1,0.5,0.667,0.833) * C(x3|-1,0.5,0.667,0.833) *
C(x6|+1,0.167,0.333,0.5)
23      0.00012      0.00013      0.99782         30.512    _/ _/ \_
BF20 * C(x3|-1,0.167,0.333,0.5)
7       0.00012      0.00012      0.99784        -2.3035    _/ \_
C(x3|+1,0.5,0.667,0.833) * C(x5|-1,0.167,0.333,0.5)
28      0.00012      0.00012      0.99785         1.5122    _/
C(x6|+1,0.167,0.333,0.5)
```

```
4       0.00012      0.00012      0.99794        -1.4285    _/
C(x6|+1,0.5,0.667,0.833)
38      0.00012      0.00012      0.99796        -2.1263    \_ \_
BF5 * C(x3|-1,0.167,0.333,0.5)
11      0.00011      0.00011      0.99800         -14.68    \_ _/ _/
BF9 * C(x3|+1,0.5,0.667,0.833)
24      0.00011      0.00011      0.99802         1.0286    _/ _/
C(x3|+1,0.5,0.667,0.833) * C(x6|+1,0.167,0.333,0.5)
19      0.00011      0.00011      0.99803         3.6606    \_ _/ _/
C(x5|-1,0.5,0.667,0.833) * C(x6|+1,0.167,0.333,0.5) *
C(x3|+1,0.5,0.667,0.833)
20      0.00011      0.00011      0.99805         8.6149    _/ _/
BF2 * C(x6|+1,0.167,0.333,0.5)
14      0.00011      0.00011      0.99810         9.2566    _/ \_ \_
C(x5|+1,0.5,0.667,0.833) * C(x3|-1,0.5,0.667,0.833) * C(x6|-
1,0.167,0.333,0.5)
8       0.00011      0.00011      0.99811         7.6232    _/ _/
BF2 * C(x3|+1,0.5,0.667,0.833)
36      0.00011      0.00011      0.99813         27.859    _/ _/ _/
BF8 * C(x6|+1,0.5,0.667,0.833)
10      0.00010      0.00010      0.99827        -4.9552    _/ _/
BF2 * C(x3|+1,0.167,0.333,0.5)
18      0.00008      0.00009      0.99852         4.2098    _/ _/ \_
BF10 * C(x6|-1,0.5,0.667,0.833)
17      0.00008      0.00008      0.99854        -7.4774    _/ _/ _/
BF10 * C(x6|+1,0.5,0.667,0.833)
34      0.00008      0.00008      0.99855         2.3485    _/ _/ _/
BF6 * C(x6|+1,0.5,0.667,0.833)
9       0.00008      0.00008      0.99856         1.1825    \_ _/
BF5 * C(x5|+1,0.5,0.667,0.833)
21      0.00008      0.00008      0.99860       -0.99092    _/ \_
BF2 * C(x6|-1,0.167,0.333,0.5)
35      0.00008      0.00008 !    0.99861       0.042691    _/ _/ \_
BF6 * C(x6|-1,0.5,0.667,0.833)


The model
============================================================
BF1 = C(x3|-1,0.5,0.66667,0.83333)
BF2 = C(x5|+1,0.5,0.66667,0.83333)
BF3 = C(x5|-1,0.5,0.66667,0.83333)
BF4 = C(x6|+1,0.5,0.66667,0.83333)
BF5 = C(x6|-1,0.5,0.66667,0.83333)
BF6 = C(x3|+1,0.5,0.66667,0.83333) * C(x5|+1,0.16667,0.33333,0.5)
BF7 = C(x3|+1,0.5,0.66667,0.83333) * C(x5|-1,0.16667,0.33333,0.5)
BF8 = BF2 * C(x3|+1,0.5,0.66667,0.83333)
BF9 = BF5 * C(x5|+1,0.5,0.66667,0.83333)
BF10 = BF2 * C(x3|+1,0.16667,0.33333,0.5)
BF11 = BF9 * C(x3|+1,0.5,0.66667,0.83333)
BF12 = BF9 * C(x3|-1,0.5,0.66667,0.83333)
BF13 = C(x5|+1,0.5,0.66667,0.83333) * C(x3|-1,0.5,0.66667,0.83333) *
C(x6|+1,0.16667,0.33333,0.5)
BF14 = C(x5|+1,0.5,0.66667,0.83333) * C(x3|-1,0.5,0.66667,0.83333) *
C(x6|-1,0.16667,0.33333,0.5)
BF15 = C(x3|+1,0.5,0.66667,0.83333) * C(x6|+1,0.5,0.66667,0.83333)
BF16 = C(x3|+1,0.5,0.66667,0.83333) * C(x6|-1,0.5,0.66667,0.83333)
BF17 = BF10 * C(x6|+1,0.5,0.66667,0.83333)
BF18 = BF10 * C(x6|-1,0.5,0.66667,0.83333)
BF19 = C(x5|-1,0.5,0.66667,0.83333) * C(x6|+1,0.16667,0.33333,0.5) *
C(x3|+1,0.5,0.66667,0.83333)
BF20 = BF2 * C(x6|+1,0.16667,0.33333,0.5)
```

```
BF21 = BF2 * C(x6|-1,0.16667,0.33333,0.5)
BF22 = BF20 * C(x3|+1,0.16667,0.33333,0.5)
BF23 = BF20 * C(x3|-1,0.16667,0.33333,0.5)
BF24 = C(x3|+1,0.5,0.66667,0.83333) * C(x6|+1,0.16667,0.33333,0.5)
BF25 = C(x3|+1,0.5,0.66667,0.83333) * C(x6|-1,0.16667,0.33333,0.5)
BF26 = C(x3|+1,0.16667,0.33333,0.5) * C(x6|-1,0.16667,0.33333,0.5)
BF27 = BF1 * C(x6|-1,0.16667,0.33333,0.5)
BF28 = C(x6|+1,0.16667,0.33333,0.5)
BF29 = C(x5|-1,0.5,0.66667,0.83333) * C(x3|-1,0.16667,0.33333,0.5) *
C(x6|-1,0.16667,0.33333,0.5)
BF30 = C(x5|-1,0.5,0.66667,0.83333) * C(x6|-1,0.16667,0.33333,0.5) *
C(x3|+1,0.5,0.66667,0.83333)
BF31 = C(x5|-1,0.5,0.66667,0.83333) * C(x6|-1,0.16667,0.33333,0.5) *
C(x3|-1,0.5,0.66667,0.83333)
BF32 = C(x6|-1,0.5,0.66667,0.83333) * C(x5|-1,0.5,0.66667,0.83333) *
C(x3|+1,0.5,0.66667,0.83333)
BF33 = C(x6|-1,0.5,0.66667,0.83333) * C(x5|-1,0.5,0.66667,0.83333) *
C(x3|-1,0.5,0.66667,0.83333)
BF34 = BF6 * C(x6|+1,0.5,0.66667,0.83333)
BF35 = BF6 * C(x6|-1,0.5,0.66667,0.83333)
BF36 = BF8 * C(x6|+1,0.5,0.66667,0.83333)
BF37 = BF5 * C(x3|+1,0.16667,0.33333,0.5)
BF38 = BF5 * C(x3|-1,0.16667,0.33333,0.5)
BF39 = BF38 * max(0, x5 +0)
y = -0.092418 -0.31735*BF1 +1.0091*BF2 -0.33044*BF3 -1.4285*BF4
+1.5005*BF5 +2.8202*BF6 -2.3035*BF7 +7.6232*BF8 +1.1825*BF9 -
4.9552*BF10 -14.68*BF11 -9.9709*BF12 -30.502*BF13 +9.2566*BF14 -
3.6111*BF15 -3.1265*BF16 -7.4774*BF17 +4.2098*BF18 +3.6606*BF19
+8.6149*BF20 -0.99092*BF21 -17.908*BF22 +30.512*BF23 +1.0286*BF24
+10.291*BF25 -6.5903*BF26 -3.2237*BF27 +1.5122*BF28 +10.933*BF29 -
9.3634*BF30 -5.1669*BF31 +6.9259*BF32 +2.6298*BF33 +2.3485*BF34
+0.042691*BF35 +27.859*BF36 +1.583*BF37 -2.1263*BF38 +5.5776*BF39


Variable Importance of the model
====================================
Estimated input variable importance:
Variable    delGCV      nSubsets       subsRSS       subsGCV
1            0.000             0         0.000         0.000
unused
2            0.000             0         0.000         0.000
unused
3          100.000            38       100.000       100.000
4            0.000             0         0.000         0.000
unused
5           78.064            37        57.576        57.591
6           35.184            35        11.581        11.590
7            0.000             0         0.000         0.000
unused

    'Output Target:EmergDAS'

Testing on test data
================================================

Running Prediction Tests
==========================================
Legates =
    0.9704
    0.9592
```

```
'End of Output Target:EmergDAS'
```