AUTOMATING LAND COVER CLASSIFICATION USING TIME SERIES NDVI: A CASE STUDY IN THE BERG RIVER CATCHMENT AREA

AYODEJI STEVE ADESUYI (BSc Honours Geoinformatics)

Thesis submitted in partial fulfilment of the requirements for the degree Masters in Geoinformatics in the Faculty of Science at Stellenbosch University.

SUPERVISOR: MRS Z MŰNCH

December 2016

DEPARTMENT OF GEOGRAPHY AND ENVIRONMENTAL STUDIES

DECLARATION

By submitting this Master's thesis report electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Signature: ..... Date: 06/06/2016

ABSTRACT

The processing of large volumes of geographic information system (GIS) and remote sensing (RS) data necessitates the development of automated techniques which are cost-effective, faster and user-friendly in order to aid spatial decision making. In this study, an automated technique for identifying agricultural land cover was developed using a custom tool. Multiple ensemble classifiers in ArcGIS workflow automation tool (MEAWAT) was tested on time-series MODIS normalised difference vegetation (NDVI) data using the Berg River catchment area of Western Cape, South Africa as a case study. Although the tool was developed to perform agricultural land cover classification using MODIS input data, the tool was subsequently applied to Landsat NDVI data of the same study extent. A few modifications to the tool were implemented to accommodate the different satellite imagery. The tool was built on an ArcGIS/Python platform, and various GIS & RS functions usually performed in a variety of different software packages were integrated, including study area selection, reprojection, classification and accuracy assessment.

The NDVI phenology curve was used to create training data for the classification. Different parameters were tested which allow users to engage with different rules and derive a suitable land cover map for their purpose. MEAWAT uses decision tree and ensemble classifiers such as random forest and extra-tree as well as boosting using a meta-estimator (AdaBoost). Classification accuracies of 70.5%, 75.5%, 76.3% and 78.7% were achieved respectively with MODIS data, while an accuracy of 89% was achieved using the boosted random forest classifier on the Landsat data. It was observed that a better classification output can be derived using MEAWAT on higher resolution satellite imagery provided good training data are available. These findings highlight the potential of MEAWAT for large dataset land cover classification using different satellite imagery. In addition, it exposed limitations of the tool, indicating that various adjustments will be needed on the tool when working with other satellite imagery different from MODIS and Landsat.

KEY WORDS AND PHRASES

MODIS, Landsat 8, NDVI, MEAWAT, land cover, image classification, decision tree, ensemble classifiers, remote sensing.

OPSOMMING

Die verwerking van groot volumes geografiese inligtingstelsel- (GIS) en afstandswaarnemingsdata noodsaak die ontwikkeling van outomatiese tegnieke wat koste-doeltreffend, vinnig en gebruikersvriendelik is ten einde ruimtelike besluitneming te ondersteun. In hierdie studie is ŉ geoutomatiseerde tegniek vir die identifisering van landbou-verwante landbedekking met behulp van ŉ pasgemaakte instrument ontwikkel. Veelvuldige geheelklassifiseerder in ArcGIS outomatiese instrument (MEAWAT) is op die MODIS genormaliseerde verskil plantegroei-indeks (GVPI) tydreeksgegewens van die Bergrivieropvangsarea in die Wes-Kaap, Suid-Afrika, getoets. Alhoewel die instrument ontwikkel is om landbou-verwante landbedekking met behulp van MODIS-data te klassifiseer, is die instrument ook op Landsat GVPI-data vir dieselfde studiegebied toegepas. Die instrument is effens aangepas sodat verskillende satellietbeeldtipes geakkommodeer kon word. Die instrument is op die ArcGIS/Python-platform gebou en die GIS- en afstandswaarnemingfunksies wat gewoonlik deur ŉ verskeidenheid sagtewarepakkette vervul word, is geïntegreer, insluitende die seleksie van die studie-area, herskatting, klassifikasie en assessering van akkuraatheid.

Die GVPI-fenologiekurwe is gebruik om opleidingsdata vir die klassifikasie te skep. Verskillende parameters, watgebruikers in staat stel om verskeie reëls te gebruik om ŉ geskikte grondbedekkingkaart vir hulle doeleindes te ontwikkel, is getoets. Die MEAWAT-instrument gebruik beslissingsbome en geheelklassifiseerders soos ewekansige-woud en ekstra boom, asook versterking deur middel van ŉ meta-beramer (AdaBoost). Klassifikasie-akkuraatheid van onderskeidelik 70.5%, 75.5%, 76.3% en 78.7% is met die MODIS-data verkry, terwyl 89% akkuraatheid van die Landsat-data met behulp van die versterkte ewekansige-woudklassifiseerder verkry is. Dit is waargeneem dat ŉ beter klassifikasie afgelei kan word deur MEAWAT op hoër resolusie satellietbeelde toe te pas, maar slegs indien goeie opleidingsdata beskikbaar is. Hierdie bevindinge beklemtoon die potensiaal van MEAWAT vir die klassifikasie van groot landbedekkingdatastelle deur van verskillende satellietbeelde gebruik te maak. Dit het ook beperkings van die instrument aan die lig gebring, wat aandui dat verskeie aanpassings nodig sal wees wanneer satellietbeelde wat van MODIS en Landsat verskil gebruik word.

TREFWOORDE EN -FRASES

MODIS, Landsat 8, GVPI, MEAWAT, landbedekking, klassifikasie van beelde, beslissingbome, geheelklassifiseerders, afstandswaarneming.

ACKNOWLEDGEMENTS

I would sincerely like to thank:

- Almighty God for helping me to complete the study.
- My family, Tsholofelo Legoale for their constant support and encouragement.
- Mrs Zahn Münch (Stellenbosch University) for her incredible support, encouragement and constant guidance as my supervisor as well as motherly advice throughout the study.
- Centre for Geographic Analysis (CGA) Stellenbosch University, especially Divan Vermeulen for his input and insights on Python.
- Gert Wessels (CSIR) for training on how to use Python.
- Friends (David Adeyemi, Timilehin Okeowo, Olabanji Asekun, Pulane Sehloho, Reanne Olivier) for their support and encouragement.
- TEAMGIS and church members (RCCGDON) for their prayers.
- Stellenbosch University for their support and use of facilities.
- The National Research Foundation (NRF) for sponsoring the research.

CONTENTS

TABLES

FIGURES

ACRONYMS AND ABBREVIATIONS

AVHRR	Advanced very high resolution radiometer
BRDF	Bidirectional reflectance distribution function
CART	Classification and regression trees
CFR	Cape floristic region
CHAID	Chi-square automatic interaction detectors
DWA	Department of Water Affairs
EKF	Extended Kalman filter
EMS	Electromagnetic spectrum
ESRI	Environmental Systems Research Institute
EVI	Enhanced vegetation index
FAST	Fast algorithm for classification trees
FOSS	Free and open source software
GDAL	Geospatial data abstraction library
GIS	Geographic information system
GLC	Global land cover classification
GPP	Gross primary production
HDF	Hierarchical data format
ICM	Iterated conditional modes
ISO	International Organization for Standardization
ISODATA	Iterative self-organizing data analysis technique
IVGE	Informed virtual geographic environment
LPDAAC	Land processes distributed active archive centre
LULC	Land use land cover
MEAWAT	Multiple ensemble classifiers in ArcGIS workflow automation
MLC	Maximum likelihood classifier
MODIS	Moderate resolution imaging spectro-radiometer
MRT	MODIS reprojection tool
MVC	Maximum value compositing
NDVI	Normalized difference vegetation index
NIR	Near infra-red
OBIA	Object-based image analysis
OLI	Operational land imager

PBIA	Pixel-based image analysis
PSU	Pensylvania State University
QUEST	Quick unbiased, efficient statistical tree
RHP	River health programme
RMSE	Root mean square error
RS	Remote sensing
SDSS	Spatial decision support system
SPOT	Satellite Pour l'Observation de la Terre
SVM	Support vector machine
TIRS	Thermal infrared sensor
TM	Thematic mapping
TWOPAC	Twined object & pixel-based automated classification chain
VBA	Visual basic for application
VCI	Vegetation condition index
VI	Vegetation index
WEKA	Waikato environment for knowledge analysis

CHAPTER 1: **INTRODUCTION**

The processing of large volumes of data has necessitated the development of automated techniques in geographic information system (GIS) and remote sensing (RS). Over the years, using GIS and RS for land cover mapping has proven to be effective in enhancing our understanding of the real world through mapping and spatial modelling (Van Niekerk 2008). With the continual introduction of powerful software, both affordable and user-friendly, the GIS industry is receiving accolades, as better research is conducted and more real life problems are being solved (Da Silva Brum et al. 2013). GIS provides the necessary tools and techniques that help to support strategic decision making (Van Wyngaarden & Waters 2007: s.p.).

With the evolution of using GIS and RS for land cover mapping, techniques such as majority filtering, unsupervised classification and layer stacking have been developed, which can be regarded as the evolution of an automation process in a basic way (Luccio 2013). These techniques have replaced the traditional techniques of image analysis such as tape rule measurement and paper maps (Goodchild 2006). Furthermore, with the aid of automated techniques, image processing is conducted within a GIS framework in near real-time and on demand, as opposed to the traditional labour intensive techniques of processing imagery (Luccio 2013). In order to efficiently meet the near real-time demand for GIS products and services, and provide solutions to real world problems in the shortest possible time, the need for automation is clear.

Technological advances have also increased the demand for automated processes in GIS and RS. An increase in affordable, available data and file storage has brought about a gradual change from the file-based structures towards cloud-based services, accessed by subscribing to such services. As a result, the user can access data automatically without the rigours of manual data collection and processing (Luccio 2013). Due to continuous transformation and changes in the Earth's surface, innovative methods or techniques are required to ascertain vegetation activities and land surface changes (Colditz 2007). Moderate resolution imaging spectroradiometer (MODIS) vegetation index (VI) can be used for mapping land cover and timing of seasonal activities of the vegetation cover (Wardlow & Egbert 2010). Land cover change can also be addressed and understood by using time-series MODIS normalized difference vegetation index (NDVI) data (Lunneta et al. 2006).

This chapter sets out to introduce automation and its role in the evolution of GIS and RS for land cover mapping. It concludes with the problem statement, aim and objectives of this study, a description of the study area and introduces the research methodology implemented in the thesis.

1.1 AUTOMATION AND GIS FOR LAND COVER MAPPING

Information technologists describe automation as the linking of disparate systems and software in such a way that they become self-operating (Sucic & Capuder 2016). These operations can range from a simple process such as printing from a laptop to performing complex tasks such as voice command applications on a phone. Automation involves the technique of making a process run automatically with minimal or no human (manual) interference (PSU 2014).

Various geo-processing tools in software packages such as ArcGIS (Maree et al. 2003), Environment for Visualizing Images (ENVI) (Exelis 2013b), Waikato environment for knowledge analysis (WEKA) (Witten & Eibe 2000) and eCognition (Benz et al. 2004) allow automation. Geo-processing can be described as operations within GIS that are used to manipulate spatial data. Geo-processing tools can enable automation of different GIS tasks, including spatial analysis and modelling. This facilitates GIS tasks involving repetition, and provides built-in documenting methods for multiple steps as workflows become complex (ESRI 2013).

Automation within GIS may involve the combination of sequences of tools through models and scripts. GIS users who wish to perform automated GIS operations for their respective analysis can also carry out such automated analyses with the aid of a customized tool, by using the model tool (ModelBuilder) in ArcGIS (Sugumaran & Degroote 2010), data mining in WEKA (Hall et al. 2009), or scripting tools (Python, Arc macro language (AML) executable files) (ESRI 2013). Geo-processing can also be performed within ENVI software through layer stacking and regression trees.

Spatial knowledge of land cover information is essential for planning, management and monitoring of natural resources (Yacouba, Guangdao & Xingping 2009). With the increase in remotely sensed data – as a result of more satellites being launched and lower cost of imagery – there is increased demand for automation techniques to handle large volumes of data (PSU 2014). Various industries are interested in the capturing, streamlining and processing of recent information about conditions and the continuous dynamic changes of the earth surface. This can effectively be expedited using remote sensing technology, and automated processes will ease processing of large volumes of such data (Yacouba, Guangdao & Xingping 2009).

Over the years remotely sensed data from traditional sources such as Landsat thematic mapper (TM) and enhanced thematic mapper plus (ETM+), advanced very high resolution radiometer (AVHRR) and Satellite Pour l'Observation de la Terre (SPOT) have proven functional for land

use and land cover (LULC) classification because of their synoptic and continuous coverage (Wardlow & Egbert 2008). Significant progress has been made in classifying LULC at different spectral and spatial resolutions (Jiang et al. 2012; Loveland et al. 2000; Wardlow & Egbert 2008; Wessels et al. 2004).

Automated approaches for image analysis aid in efficient extraction of useful information from an image (Campbell & Wynne 2011). Mekru, Moulin & Bergeron (2012) used an automated approach resulting in accurate, enhanced representation of the real world by land cover classification through the generation of informed virtual geographic environment (IVGE). The object and pixel-based approach, both in supervised and unsupervised mode, have been widely used for image analysis which entails basic automation (Abburu & Golla 2015) with minimal human intervention. In addition, machine learning algorithms such as decision tree (DT), ensemble algorithms and artificial neural networks are effective for image analysis, which entails automation (Sharma, Ghosh & Joshi 2013, Torma 2013). Jiang et al. (2013) observed that using a semi-automatic approach in obtaining training samples with GIS for land cover classification is also advantageous, as this reduces error derived from using spectral pattern alone for classifying remotely sensed images. Another advantage of an automated approach is the ability to make analysis easier (Cleveland 2009) by storing parameters used during iterations, thereby reducing error introduced by human action. This in turn may reduce error propagation which can lead to an increased workflow efficiency (PSU 2014).

Automation is therefore essential when dealing with large volumes of data and complex models. Various GIS and RS software enables automation in different ways and makes image analysis for land cover mapping easier and faster. This study will focus on demonstrating the usefulness of automation for efficiently, solving the problem stated in the next section.

1.2 PROBLEM STATEMENT

Recent research has shown that using time-series data to model land cover could be time-consuming and that it often involves many manual tasks (Tyrallora & Gonschorek 2012). In a recent land cover classification study focused on identifying different agricultural crops, Adesuyi & Münch (2015) observed that processing large volumes of satellite data in different software packages was not only cumbersome and time-consuming but increased the likelihood for human errors. Consequently, an integrative platform is needed to incorporate different software tools for image processing and classification, that would save time, cost and reduce possible errors.

For such integration of geo-processing techniques, a suitable automation process and an interactive software platform is required. This will require knowledge and understanding of geo-processing tools, interaction between software packages, and a flexible programming language to interpret software functions. Ideally, the derived integrative platform should be effective to undertake land cover classification of large areas more rapidly than the traditional manual classification. The following questions arise from the problem stated above: How can an automated technique be developed for land cover classification ensuring seamless integration between different software packages? How can this process be used to effectively identify agricultural land cover classes?

1.3 AIM AND OBJECTIVES

Following on from the research questions, the primary aim of this study is to develop an automated workflow on an appropriate software platform to integrate different software packages for classifying agricultural land cover using time-series NDVI data.

The following objectives must be achieved for this research to reach its stated aim:

1. Review relevant literature on geo-processing tools and automated techniques for image classification.
2. Collect applicable reference and input data e.g. aerial photographs, ancillary data and MODIS and Landsat imagery.
3. Select integrative platform and software tools.
4. Develop the automated workflow.
5. Demonstrate the use of the automated workflow using MODIS and Landsat imagery to classify agricultural land cover.
6. Evaluate the usefulness of the automated process to identify agricultural land cover classes.

To achieve these objectives and demonstrate the workflow, a suitable study area, the Berg River catchment, was identified.

1.4 STUDY AREA

The Berg River catchment area in the Western Cape Province of South Africa (Figure 1.1) covers an area of approximately 9000 km^2, and is the largest catchment in the Province. Its center coordinate is on latitude 33.9024S and longitude 19.0570E. The catchment can be classified into three major parts based on topography; 1) the flat and extensive portion which lies west of Moorreesburg and Koringberg, 2) the river valley area which lies east of Koringberg to the south of Paarl, and 3) the upper mountainous area which is south of Paarl (Figure 1.1). The

terrain is mostly flat, with average topographic gradient of 0.001% between Paarl and the mouth of the Berg River at Laaiplek. The Berg River itself has a length of 285 km (DWA 2013).

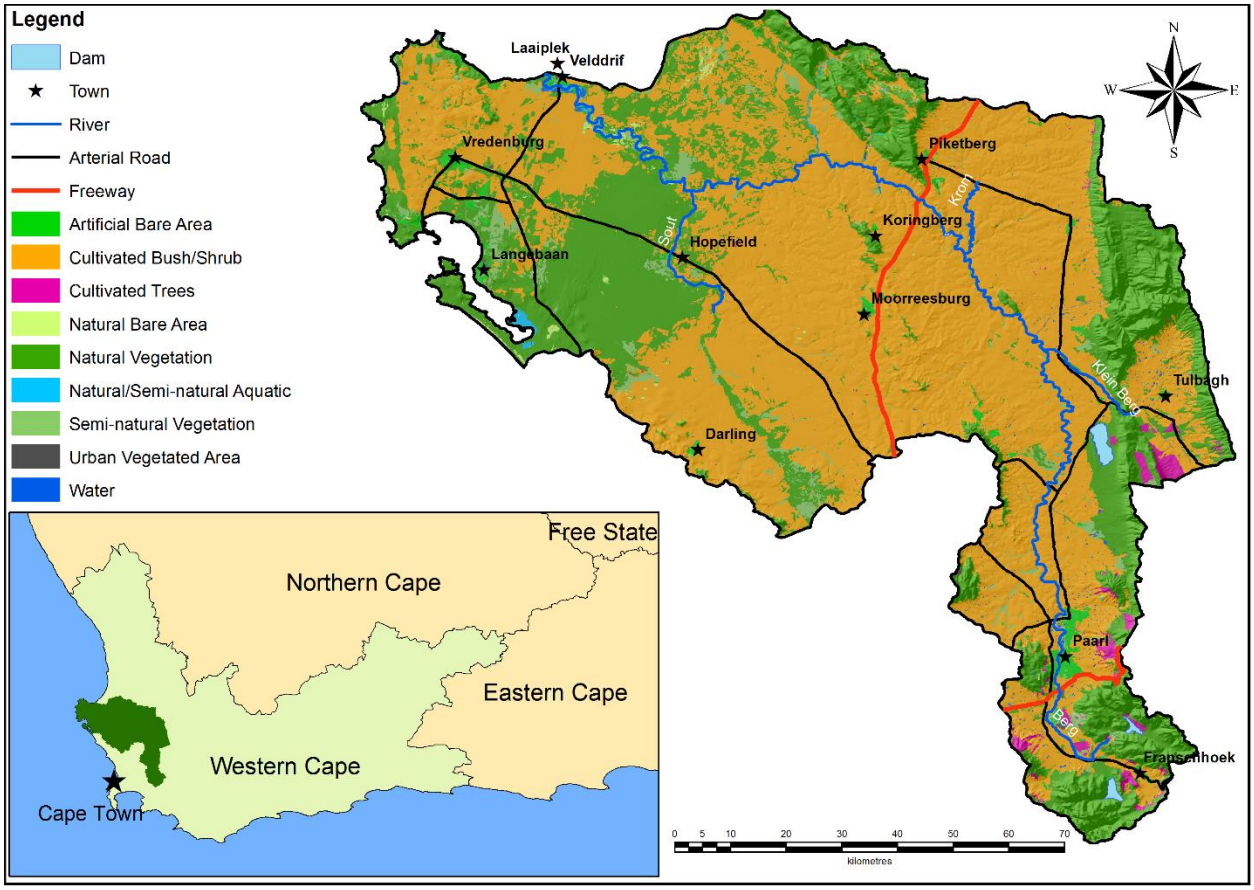


Figure 1.1 Study Area: Major towns, roads, and vegetative land cover class of the Berg River catchment

The western region of the catchment has finely textured and fertile soils which emanate from the Cape Granite suite, making it very suitable for agriculture. The presence of sandy sediments and rich clayey soils in the lowlands and middle catchment makes it possible for crop production to thrive in the region and thus enhances agricultural development (Clark & Ratcliffe 2007). Mountains with elevations in excess of 1000m flank the eastern part of the catchment around the north-orientated valley (eWISA 2008), dominated by acidic soils of the Table Mountain Group (TMG). Since the presence of large rocks and mountains renders part of the catchment unsuitable for agricultural use (eWISA 2008; RHP 2004), natural vegetation, especially fynbos, abounds in this region (Stuckenberg 2012).

The mean annual temperature in the study area varies between 16°C in the east and 18°C in the west, with a maximum in January of 29.4°C and a minimum of 4.5°C in July. The mean annual rainfall ranges between 300mm in the lower catchment to 1400mm in the mountainous upper catchment (eWISA 2008) and occurs predominantly in winter (May – August).

Due to the temperate climate and fertile soils, land use in the catchment is predominantly agricultural (~60%), and the area is commonly regarded as the food basket of the Western Cape Province. However, the advent of agriculture has reduced the presence of the natural vegetation species in the catchment (RHP 2004) and alien vegetation such as *Acacia Cyclops* compete for water from the Berg River. The road network in the catchment is sparse with most roads servicing major urban areas while large swathes of agricultural and natural areas remain inaccessible. The catchment has a well-developed drainage network which supplies irrigation for agriculture, dams and urban centres. Besides many small farm dams, Wemmershoek Dam, Voëlvlei Dam, Misverstand Dam and the Berg River Dam, at the base of the Franschhoek Mountains, provide significant water storage (DWA 2013).

Agricultural products include wheat, wine grapes, deciduous fruit, vegetables, grain and sheep farming. Grapes and deciduous fruits are cultivated intensively in the colder eastern regions, generating significant foreign revenue (R1.3 billion) annually from the export of fruit, wine and spirits, as most of the production is exported. Dry land, small grains (e.g. wheat, canola) and extensive livestock (cattle and sheep) dominate the middle to lower catchment to the west. Agricultural practices include crop rotation and fallowing which enables soils to recover (RHP 2004). In addition, indigenous 'fynbos' flowers (e.g. Protea), olives, dairy products, pigs and poultry are produced. Agriculture also drives much of the secondary economy in the form of fruit and vegetable processing, including canning, drying, juicing, and jam production.

The majority (79%) of the population resides in urban areas, the largest of which are Velddrif and Laaiplek near the coast, Mooreesburg, Piketberg, Hopefield and Darling further inland and Paarl and Wellington in the upper catchment (RHP 2004). Fisheries at Laaiplek and Velddrif constitute important industries in these areas. Owing largely to the perceived aesthetic appeal of the area, tourism and recreation are substantial and rapidly growing industries, driven partially by conservation efforts (Haiden et al. 2014; RHP 2004; Stuckenberg 2012).

As part of the Cape Floristic Region (CFR), this biodiversity 'hotspot' not only has immense intrinsic value for the healthy functioning of the ecosystems of the catchment, but also has economic value associated with wildflower harvesting and ecotourism. Conservation efforts have been aided by the proclamation of numerous protected areas, concentrated in the mountains and the West Coast area (Turner et al. 2012).

1.5 RESEARCH METHODOLOGY AND AGENDA

This research is investigative and experimental in nature, with the conceived outcome being an automated tool through which land cover mapping of large volumes of satellite imagery can be carried out on a suitable platform (to be identified). The applicability of the tool will be tested for automating agricultural land cover mapping over large areas in order to reduce extensive manual processing and thereby reducing human error.

The research comprised four phases namely: 1) knowledge building, 2) planning, 3) execution, 4) evaluation and conclusion, as outlined in the research design (Figure 1.2). The knowledge building phase, which entailed identification and formulation of the problem and stating the research aim and objectives, has been addressed in this chapter. The second research activity for the knowledge building phase was conducting a literature review to substantiate what automation in GIS and RS for land cover mapping entails, applicable throughout each methodological phase. The literature review (Chapter 2) also covers different image analysis approaches, use of different satellite imagery and automated approaches for land cover classification.

Knowledge building leads into the planning phase during which data collection and development of a conceptual and methodological framework for the research was carried out. Chapter 3 provides a review of the technologies, scripts and coding available for the development of a customized tool thereby affording a basis for the execution phase.

The execution phase included tool development, pre-processing of data and implementation of land cover classification using the automated workflow, which will be discussed in Chapter 4. Successful implementation of the execution phase would enable a proper evaluation of the process (evaluation phase). The demonstration of the tool and evaluation of the classification will be discussed in Chapter 5.

Finally, in the conclusion phase, the research was critically assessed regarding the achievement of its stated objectives. The prospects and limitations of automating land cover mapping using time-series NDVI using the automated workflow are discussed in Chapter 6. The dissertation concludes with recommendations for further research.

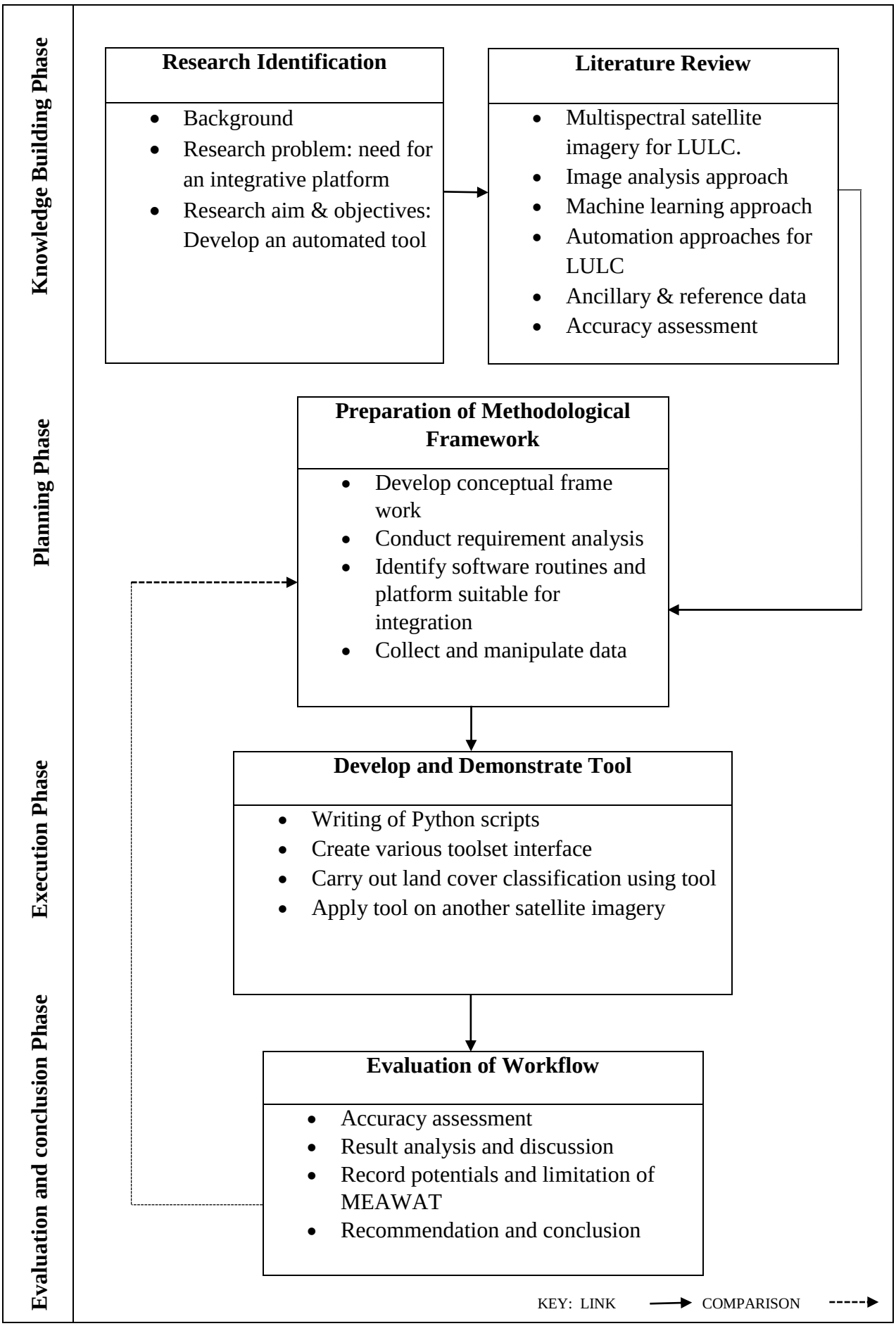


Figure 1.2 Research design

CHAPTER 2: LITERATURE REVIEW

In this chapter the following concepts will be addressed according to the research design: (1) multispectral satellite imagery for land cover mapping; (2) image analysis approaches; (3) machine learning using decision trees and ensemble methods; (4) concepts of automation; (5) ancillary and reference data; and (6) accuracy assessment.

Remote sensing (RS) can be described as a science where an object's physical characteristics are determined from afar without contact with the object (Campbell & Wynne 2011). Blaschke (2010) states, that for RS imagery to work with other datasets, conversion of such data into useful information must take place within relevant geographic information systems (GIS). GIS can be described as a tool for managing collected spatial data. Technological advances have led to a need for greater integration between GIS and RS. Such integration can be further enhanced and made efficient through automation.

An automated approach is faster than the conventional way of data analysis as it helps to reduce error propagation. GIS integration with RS can make data analysis easier, as RS (supplier) and GIS (consumer) can co-exist and function from a single platform (Rogan & Miller 2006). Furthermore, integration of both through automated techniques yielded high accuracies in forest change detection (Rogan & Miller 2006) as well as environmental monitoring and change analysis (Ehlers, Gachter & Janowsky 2006). Researchers such as Aitkenhead & Aalders (2001); Rogan & Miller (2006) and Tyrallora & Gonschorek (2012) see automation as a process of integrating GIS and RS by combining different satellite images and analysing them in a GIS environment. However, notable challenges such as incompatible data types and lack of an integrative method for spatial data analysis may arise during the integration of RS and GIS (Rogan & Miller 2006).

Various types of imagery have been used for land cover mapping. This includes multispectral, hyper-spectral and radar data. In this study the emphasis will be on multispectral imagery, which is discussed in more detail in the next section.

2.1 MULTISPECTRAL IMAGERY FOR LULC MAPPING

In multispectral imagery, specific frequencies across the electromagnetic spectrum (EMS) are captured and distributed in constant wavelength intervals. The EMS ranges are influenced by electromagnetic radiation received in the multispectral sensor (Lillesand, Kiefer & Chipman 2004). The most frequently used band combination represent the red, green, blue (RGB) and near-infrared (NIR) regions of the EMS (Gibson 2000; Lillesand, Kiefer & Chipman 2004).

Different band combinations facilitate the identification of features invisible in a true colour (RGB) image. The significance of spectral bands to enhance the contrasting attributes of features or targets defines the capacity of multispectral datasets in generating LULC mapping and classification (Gibson 2000; Keith et al. 2009; Lillesand, Kiefer & Chipman 2004).

Multi-temporal satellite imagery can be used to gather data over a period of time and consequently provide more information about land cover change patterns than single-date imagery. Multispectral satellites used for LULC mapping include high resolution QuickBird (Frohlich et al. 2013; Her & Heatwole 2007; Kux et al. 2011; Myint et al. 2011), IKONOS (Boloorani, Erasmi & Kappas 2003; Da Silva Brum et al. 2013; Faour & Kheir 2006; Pu, Landry & Yus 2011), RapidEye (Kim & Yeom 2012; Kim, Yeom & Kim 2011; Kindu et al. 2013; Schulthess et al. 2008; Tapsall, Milenov & Tasdemir 2010) and SPOT (Bartholome & Belward 2005; Gong et al. 2008; Lewinski & Bochnek 2008; Lim, Matjafri & Abdullah 2009; Tateishi & Mukouyama 2008; Yacouba, Guangdao & Xingping 2009), medium resolution Landsat (Frost, Epstein & Walker 2014; Ioannis & Meliadis 2011; Kokalj & Ostir 2007; Wardlow & Egbert 2008; Wardlow, Egbert & Kastens 2007; Wessels et al. 2004; Yacouba, Guangdao & Xingping 2009) as well as coarser resolution AVHRR (Colditz 2007; Gopal, Woodcock & Strahler 1999; Jonsson & Eklundh 2002; Liang 2001; Liu et al. 2003; Loveland et al. 2000; Pericles 2007; Wardlow & Egbert 2008; Wardlow, Egbert & Kastens 2007; Weng 2011) and MODIS (Brown et al. 2012; Carrao, Goncalves & Caetano 2008; Chen et al. 2004; Colditz 2007; Giri & Jenkins 2005; Kleynhans et al. 2011; Lu et al. 2014; Sah et al. 2012; Spruce et al. 2011).

Very high resolution QuickBird imagery is unsuitable for application over large areas due to its costs. The use of SPOT and RapidEye data for repetitive and global mapping has been restricted because of the high cost and time associated with the acquisition and processing of required scenes (Yacouba, Guangdao & Xingping 2009). Although IKONOS imagery has a high potential for land cover mapping, it is also not efficient when carrying out classification on large areas or at global scale, due to data availability, high cost, and low temporal resolution (Da Silva Brum et al. 2013). Lower resolution AVHRR suffers from inadequate cross-calibrations, erroneous geo-location, an insufficient number of bands for full atmospheric correction and orbital drifts, undetailed classification and inconsistent results (Colditz 2007).

MODIS offers a medium spatial resolution, a wide range of bands, is atmospherically corrected, and has a high dynamic range in vegetated areas between the red and NIR band (Colditz 2007; Wessels et al. 2004; Zhang et al. 2003). The resolution, spectral and spatial characteristics of Landsat also makes it very suitable for land cover mapping (Frost, Epstein & Walker 2014). In

addition it is freely available. The use of MODIS for vegetation studies will be reviewed in the next section.

2.1.1 MODIS NDVI for vegetation studies

MODIS imagery provides scientific quality data with high temporal resolution of one to two days and intermediate spatial resolution of 250m for bands 1 and 2 (Justice & Townshend 2002), which is well suited for crop mapping and monitoring (Wessels et al. 2004). MODIS has 36 spectral bands (Table 2.1) and offers adequate and automatic atmospheric correction with thorough cloud masking (Wolfe et al. 2002).

Table 2.1 MODIS sensor characteristics

Specifications	
Orbit	705 km, 10:30a.m. descending node (Terra) or 1:30p.m. ascending node (Aqua), sun-synchronous, near-polar, circular
Scan Rate	20.3 rpm, cross track
Swath	2330 km (cross track) by 10 km (along track at nadir)
Dimensions	
Telescope	17.78 cm diam. off-axis, afocal (collimated), with intermediate field stop
Size	1.0 x 1.6 x 1.0 m
Weight	228.7 kg
Power	162.5 W (single orbit average)
Data Rate	10.6 Mbit/s (peak daytime); 6.1 Mbit/s (orbital average)
Quantization	12 bits
Spatial Resolution	250 m (bands 1–2); 500 m (bands 3–7); 1000 m (bands 8–36)
Life Span	6 years

Adapted from Geoscience Australia (2012)

The MOD13Q1 Aqua and Terra data are freely accessible and includes a time-series of red (620-670nm), NIR (841-876nm) and blue (459-479nm) surface reflectance, NDVI and enhanced vegetation index (EVI) as individual layers. It is composited at 16 day interval using a constrained view angle – maximum value composite approach to select pixels closest-to-nadir, resulting in 23 images per year with data values converted to integer format. By storing the value as a scaled integer, the file stays small retaining precision (Wardlow, Egbert & Kastens 2007). This affords the opportunity for comprehensive broad area vegetation analysis. The characteristics of the MOD13Q1 data product are provided in Table 2.2. For the purposes of this study, NDVI and EVI are the important bands to consider.

Table 2.2 MOD13Q1 MODIS vegetation index product

Science Data Set	Units	Data type	Valid Range	Scale factor
16 days NDVI	NDVI	int16	-2000, 10000	0.0001
16 days EVI	EVI	int16	-2000, 10000	0.0001
16 days VI Quality detailed QA	Bits	int16	0, 65534	NA
16 days red reflectance (Band 1)	Reflectance	int16	0, 10000	0.0001
16 days NIR reflectance (Band 2)	Reflectance	int16	0, 10000	0.0001
16 days blue reflectance (Band 3)	Reflectance	int16	0, 10000	0.0001
16 days MIR reflectance (Band 7)	Reflectance	int16	0, 10000	0.0001
16 days view zenith angle	Degree	int16	-9000, 9000	0.01
16 days sun zenith angle	Degree	int16	-9000, 9000	0.01
16 days relative azimuth angle	Degree	int16	-3600, 3600	0.1
16 days composite day of the year	Day of year	int16	1, 366	NA
16 days pixel reliability summary QA	Rank	int8	0, 3	NA

Source: Didan & Huete (2006)

NDVI is a normalized difference measure comparing the NIR and red band, and a simple, effective index for appraising green vegetation (Exelis 2013a). NDVI is one of the most successful indices to simply and quickly identify vegetated areas and their "condition". It remains the most well-known and most frequently utilized index to detect green plant canopies in multispectral remote sensing data (Agone & Bhamare 2012). Once the feasibility to detect vegetation was demonstrated, users also used the NDVI to quantify the photosynthetic capacity of plant canopies. It is defined as:

$$NDVI = (P_{NIR}-P_{RED}) / (P_{NIR}+P_{RED}) \quad \text{Equation 1}$$

where P_{NIR} and P_{RED} represent surface reflectance in the NIR and red bands respectively.

EVI is designed to reduce atmospheric and soil background that pollutes the NDVI, and is more responsive to canopy structure variation than NDVI. It is defined as:

$$EVI = G\ \{(P_{NIR} - P_{RED}) / (P_{NIR}+C_1 \times P_{RED} - C_2 \times P_{BLUE} + L)\} \quad \text{Equation 2}$$

where P_{NIR}, P_{RED} and P_{BLUE} represent surface reflectance in the NIR, red and blue bands respectively. L represents canopy background adjustment. C_1 and C_2 represent coefficients of the aerosol resistance term (Huete, Justice & Van Leeuwen 1999).

NDVI temporal profiles derived from MODIS can be used to monitor vegetation phenology thereby developing a regional landscape process model (Lunetta et al. 2006). MODIS NDVI data has been used in applications such as large area crop mapping (Wardlow & Egbert 2008), time-series generation, land cover classification and vegetation cover (Bajocco et al. 2015; Colditz 2007; Panju & Trisasongko 2012; Weng 2011; Wessels et al. 2004; Zhou, Jia & Menenti

2015). Other studies using MODIS NDVI data include land cover change detection (Lunneta et al. 2006; Wang et al. 2010), as well as change in phenology detection (Bajocco et al. 2015).

In detection of land cover changes (Zhan et al. 2002), MODIS data proved suitable as its temporal frequency is adequate to differentiate change events from phenological cycles to generate a change index. As a result of MODIS sub-pixel geo-locational accuracy of ±50m at nadir (Wolfe et al. 2002), geometric inaccuracies on the vegetation index (VI) and changes between observations in a time-series is minimal (Wardlow & Egbert 2010; Wardlow, Egbert & Kastens 2007). The VI's of MODIS (NDVI and EVI) have distinct characteristics that enhance change detection and analysis (Colditz 2007), and also often a consistent spatial and temporal coverage of vegetation conditions (Wardlow & Egbert 2008).

Using MODIS for vegetation studies has its limitations which can mostly be attributed to its coarse spatial resolution (250m). However, Wang et al. (2010) determined that MODIS imagery is effective in detecting forest health despite its relatively low spatial resolution. The first seven bands in MODIS were designed to replicate Landsat 7 sensors, irrespective of their different spatial resolutions (Wang, Hu & Hu 2009). The next section will review the use of Landsat for vegetation studies.

2.1.2 LANDSAT for vegetation studies

The Landsat satellite has various on-board sensors that have advanced with technology to enable efficient vegetation study (Li, Jiang & Feng 2014). These sensors have constantly been improved, ranging from the earlier editions of Landsat that used return beam vidicon (RBV) and multispectral scanners (MSS) to recent versions including thematic mapper (TM), enhanced thematic mapper plus (ETM+), operational land imager (OLI) and the thermal infrared sensor (TIRS) (Weng 2011). Table 2.3 provides an overview of the range of Landsat sensors, resolution, available bands, revisit days, altitude and scene size (Irons, Dwyer & Barsi 2012). ETM+ and OLI can be used as complementary data for vegetation studies (Li, Jiang & Feng 2014). Landsat offers great potential in differentiating the regeneration levels of forest vegetation using false colour combination (Rokos & Argislas 1995).

Table 2.3 Landsat sensors characteristics

Satellite	Spectral Resolution (µ)	Band	Spatial Resolution (m)	Revisit Days	Altitude	Scene Size (km)
LANDSAT 1-3	**MSS**			18	917	180 x 170
	Band 4: 0.50 - 0.60	Green	79.00			
	Band 5: 0.60 - 0.70	Red	79.00			
	Band 6: 0.70 - 0.80	Near IR	79.00			
	Band 7: 0.80 - 1.10	Near IR	79.00			
LANDSAT 4-5	**MSS**			18	705	170 x 183
	Band 4: 0.50 - 0.60	Green	82.00			
	Band 5: 0.60 - 0.70	Red	82.00			
	Band 6: 0.70 - 0.80	Near IR	82.00			
	Band 7: 0.80 - 1.10	Near IR	82.00			
	TM					
	Band 1: 0.45 - 0.52	Blue	30.00			
	Band 2: 0.52 - 0.60	Green	30.00			
	Band 3: 0.63 - 0.69	Red	30.00			
	Band 4: 0.76 - 0.90	Near IR	30.00			
	Band 5: 1.55 - 1.75	Mid IR	30.00			
	Band 6: 10.4 - 12.5	Thermal	120.00			
	Band 7: 2.08 - 2.35	Mid IR	30.00			
LANDSAT 7	**ETM+**			16	705	170 x 183
	Band 1: 0.450 - 0.515	Blue	28.50			
	Band 2: 0.525 - 0.605	Green	28.50			
	Band 3: 0.630 - 0.690	Red	28.50			
	Band 4: 0.760 - 0.900	Near IR	28.50			
	Band 5: 1.550 - 1.750	Mid IR	28.50			
	Band 6: 10.40 - 12.5	Thermal	57.00			
	Band 7: 2.080 - 2.35	Far IR	28.50			
	Band 8: 0.52 - 0.92	Pan	14.25			
LANDSAT 8	**OLI & TIRS**			16	705	185 X 180
	Band 1: 0.435 - 0.451	Aerosol	30			
	Band 2: 0.452 - 0.512	Blue	30			
	Band 3: 0.533 - 0.590	Green	30			
	Band 4-5: 0.636 - 0.879	Red-NIR	30			
	Band 6-7: 1.566 - 2.297	SWIR	30			
	Band 8: 0.503 - 0.676	PAN	15			
	Band 9: 1.363 - 1.384	Cirrus	30			
	Band 10-11: 10.60 - 12.51	TIR	100			

Adapted from: Weng (2011)

Although the previous Landsat sensors have fewer bands when compared to the Landsat 8, they were still effective for mapping vegetation changes when using NDVI as the input for classification analysis (Lillesand, Kiefer & Chipman 2004). Spectral information of vegetation contained in Landsat imagery can be determined by different spectral bands, and it offers the ability to isolate spectral trends such as urban type, vegetation type and geomorphic landforms (Frost, Epstein & Walker 2014). As with MODIS, frequently used bands for vegetation mapping with Landsat are reflectance in red and NIR bands. NDVI values can be generated from Landsat using the mean brightness values for predicting information about vegetation water content

(Jackson et al. 2004). According to Frost, Epstein & Walker (2014) time-series NDVI is used to access landscape and regional scale variability of vegetation dynamics.

Other indices such as the vegetation condition index (VCI) and temperature vegetation index (TVI) can be incorporated with NDVI to investigate multi-temporal land cover change (Maskora, Zemek & Kvet 2008; Orhan, Ekercin & Dadaser-Celik 2014). Some Landsat images contain haze, cloud cover and some have scan line errors, therefore atmospheric and radiometric correction is always essential, especially when carrying out vegetation studies that entail multi-temporal analysis and change detection (Lillesand, Kiefer & Chipman 2004).

Atmospheric effects occur as a result of the passing of electromagnetic radiation through the atmosphere, whereby scattering, absorption and refraction of the radiation takes place. Therefore, the digital value recorded by the satellite receiver is not a true representation of ground conditions. By performing an atmospheric correction on the images, the at-sensor radiance is converted to true radiance or reflectance (Guo & Zeng 2012). Some of the most common algorithms used to atmospherically correct images include the Fast Line-of-sight Atmospheric Analysis of Spectral Hypercubes (FLAASH), second simulation of the satellite signal in the solar spectrum (6S) and quick atmospheric correction (QUAC) (Guo & Zeng 2012).

ATCOR is a method that corrects remotely sensed imagery covering the solar (0.4 to 2.5 μm) and the thermal region (8 to 14 μm) (Richter 2004). ATCOR 2 provides lower root mean square error (RMSE) values, and has been shown to provide higher accuracies than ATCOR 3 (Vermeulen 2011). The ATCOR 3 algorithm is designed to perform a combination of both an atmospheric correction in combination with topographic correction by making use of a digital elevation model (DEM). Atmospheric correction is essential for vegetation analyses as the NDVI values of the data are compared and as such the true radiance is therefore required. Under normal circumstances, topographic correction techniques are applied on images taken over rugged terrains.

Landsat TM and ETM+ imagery has RGB, NIR and infrared (IR) bands commonly used to assess vegetation health and cover, but has also found applications in fields such as agriculture, botany, cartography, environmental monitoring forestry, geography, geology, geophysics, hydrology, land use planning, natural resource management and oceanography (Gibson 2000; Lillesand, Kiefer & Chipman 2004; NASA 2011). Zhao et al. (2007) proposed that vegetation indices (red-NIR bands) from narrowband spectral data are better predictions of leaf area index (LAI) and canopy chlorophyll density (CCD) than the indices from broadband spectral data, indicating that hyper-spectral imagery provides better results than multispectral remotely sensed data.

2.2 IMAGE ANALYSIS APPROACHES FOR LAND COVER MAPPING

Image analysis is a framework for examining digital images to yield a final output. It can be seen as a systematic method to extract useful information from an image and involves assigning pixels to classes, with each pixel evaluated as a discrete unit composed of values in several spectral bands (Campbell & Wynne 2011). The traditional pixel-based approach is widely used for image analysis and makes use of supervised and unsupervised methods. With increasing spatial resolution, the object-based classification approach aims to utilize spectral and contextual information in an integrative way by delineating objects from imagery for classification (Blashke 2010). In the next section image analysis following these two approaches is discussed.

2.2.1 Pixel-based classification approach

Pixel-based classification can be regarded as the traditional approach for image classification as it treats pixels as discrete units. Each class consists of spectrally similar homogenous pixels, represented by different colour symbols (Campbell & Wynne 2011). This approach was used with time-series MODIS vegetation data to classify multi-year agricultural land cover (Brown et al. 2013). The pixel-based approach uses two major classification techniques namely supervised and unsupervised classification.

Unsupervised classification groups pixels with similar multispectral responses in various spectral bands into clusters or classes that are statistically divisible, with no prior knowledge of the region (Navulur 2007). The unsupervised classification technique recognizes unique classes as distinct units. This poses a challenge as the user has limited control over the list of classes or identities and as a result and spectrally homogenous classes do not necessarily match the information categories that the analyst wants (Campbell & Wynne 2011). However, unsupervised (ISODATA) clustering for the assessment of forest defoliation detection on MODIS data products yielded good outputs with application of image thresholding (segmentation) techniques (Spruce et al. 2011).

Supervised classification makes use of training sets to develop appropriate discriminant functions that distinguish each class. Analysts have control over a selected list of informational classes, and prior knowledge of the features present (Campbell & Wynne 2011; Navulur 2007). Medingegneria (2009) used the maximum likelihood (ML) supervised classification method on MODIS data for multi-temporal monitoring of land cover changes in marshland areas. For change detection using NDVI, pixels are allocated to classes which have the most likelihood of membership (Yacouba, Guangdao & Xingping 2009). However, using high resolution imagery for ML classification of urban land cover in pixel-based image analysis (PBIA) was not as

effective (67.6%) as nearest neighbour classification using object-based image analysis (OBIA) (90.40%) (Myint et al. 2011). According to Sah et al. (2012) using a single image for land cover classification is not sufficient to generate an acceptable accuracy and as such proposed the fusion of results generated from different satellite images to ensure a high accuracy.

2.2.2 Object-based classification approach

Commonly referred to as OBIA, the object-based classification approach allocates pixels into object primitives, creates segments and designates each segment into a class (Santos, Tenedorio & Encarnacao 2007). Objects are created by grouping pixels into homogenous segments using multiresolution segmentation. It is therefore more effective using high resolution data (30m or less) as creating segments requires structure detection in the datasets (Colditz 2007). Steps in OBIA include: 1) segmentation, 2) rule-set formulation and 3) classification, which are governed by crucial parameters such as scale, shape and compactness. The scale factor aids in determining different levels of object sizes, while the shape and compactness regulates the homogeneity of objects (Myint et al. 2011). Segmentation divides the image into comparably homogenous, semantically representative groups of pixels which are analysed by further processing steps (Blaschke 2010). An object-based approach yields a higher accuracy than the pixel-based approach in land cover classification when considering texture, size, shape, hierarchical and structure (Boloorani, Erasmi & Kappas 2003).

Myint et al. (2011) indicated that a traditional pixel-based approach is not as effective in identifying urban land cover classes as OBIA. For fire scar mapping, Mallinis, Pleniou & Koutsias (2010) concluded that the observed difference between using OBIA and PBIA, was insignificant. Stuckenberg (2012) favoured an OBIA approach to classify land cover when using 30m resolution Landsat data, although problems of under-segmentation or over-segmentation are associated with OBIA. According to Desclee, Bogaert & Defourny (2006) the use of unsupervised classification for forest change detection in object-based approach is not efficient. Nevertheless, many studies have shown the effectiveness of object-based analysis in classifying urban and ecologically significant classes using multi image segmentation technology applications (Mathieu, Aryal & Chong 2007; Kong, Kai & Wu 2006). An important consideration for using OBIA is the reduction of salt-and-pepper effects synonymous with traditional pixel-based classification (Yu et al. 2006).

Studies have compared pixel-based and object-based approaches for land cover classification (Bhaskaran, Paramananda & Ramnarayan 2010). Mallinis, Pleniou & Koutsias (2010) used MODIS with both techniques, and observed no significant difference between them. Using high spatial resolution data OBIA proved very effective, but time-series data are more readily

available at medium to coarse spatial resolutions. In addition, the pixel-based approach has proven to be effective for generating time-series classifications (Colditz 2007).

Classification tasks are traditionally based on statistical methods such as minimum distance-to-mean (MDM), maximum likelihood classification (MLC) and linear discrimination analysis (LDA) (Pradhan, Ghose & Jeyaram 2010), characterized by an underlying probability model. If the data are complex in structure, then to model the data in an appropriate way can become difficult.

2.3 MACHINE LEARNING APPROACHES

Machine learning is a method of data analysis that automates analytical model building. Using algorithms that iteratively learn from data, machine learning allows computers to find hidden insights without being explicitly programmed where to look (Pedregosa et al. 2011). Various machine learning approaches can be used for image analysis, as classification carried out using these machine techniques requires less human interaction and the system can generate the classification on its own (Keuchel et al. 2003).

Ehlers (2006) developed an automated technique for environmental monitoring and change analysis using an index-based segmentation pre-classification procedure, by combining digital image data with integrated GIS processing environment to achieve an accuracy of 91.4%. This shows that the object-based classification approach supports automation and data integration. The expert-system rule-set is a set of processes, describing feature characteristics to determine if the object belongs to a class or not. Classification groups the objects into classes using supervised machine learning algorithms such as nearest neighbour (NN) and classification and regression trees (CART), a membership function approach based on shape, size, colour and pixel topology (Navulur 2007).

Machine learning algorithms used for image classification include classification and regression trees (CART) (Maree et al. 2003), decision trees (DT) (Brown et al. 2013; Colditz 2007), artificial neural network (Aitkenhead & Alders 2011) and support vector machines (SVM) (Lazar & Shellito 2009; Otukei & Blaschke 2010). Machine learning algorithms reduce the burden on expert knowledge to create decision boundaries for image classification (Lazar & Shellito 2009; Otukei & Blaschke 2010) and offer the opportunity to work directly with pixel values without precise pre-processing of the image. Machine learning and data mining methodologies (such as artificial neural networks and agent-based modelling) have been adapted for the classification of geospatial data in numerous studies (Cheng et al. 2009; Lazar & Shellito 2009; Pijanowski et al. 2002).

Maree et al. (2003) reflects on the potential of using an ensemble of DTs, SVM and extra-tree with sub-window extraction to handle information from pixels and thus generating a good result from them. A study carried out by Otukei & Blaschke (2010) on SVM noted that the outcomes were based on the type of kernel used that also showed high accuracies. Keuchel et al. (2003) combines a traditional object-based system, machine learning approach and a fuzzy prior knowledge for image classification and identification to generate a land cover map for a large region. This reveals how machine learning algorithms can be used with other approaches.

The most frequently used approach for image classification of agricultural classes with MODIS data has been the DT algorithm (Brown et al. 2013; Colditz 2007; Otukei & Blaschke 2010; Quinlan 1993; Sharma, Ghosh & Joshi 2013; Torma 2013; Wardlow & Egbert 2010, 2008). The following sub-sections therefore provide background information on the tree classifiers, their advantages, disadvantage and use for land cover mapping.

2.3.1 Decision trees

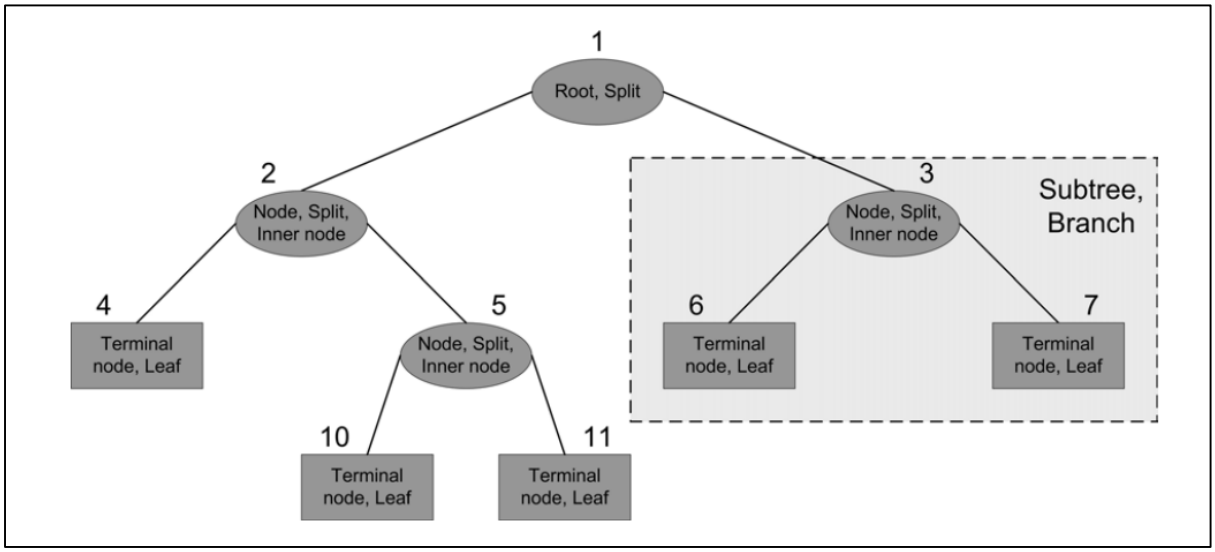
The DT method is non-parametric and therefore does not require prior assumptions of normally distributed training data (Lowry et al. 2007). It allows multi-modal distribution in input data based on threshold and rules at a multi-spectral scale (Pal & Mather 2001). DTs can readily accept various measurement scales in addition to categorical variables and have demonstrated improved accuracies over the use of traditional parametric classifiers (Hansen, Dubayah & DeFries 1996; Pal & Mather 2003).

A DT is created in two phases, namely the tree building phase and tree pruning phase. The tree building phase involves repeatedly partitioning the training data based on attribute type until all samples in each partition belongs to one class. The tree usually has a starting point, which can be regarded as the root at the top of the tree, which is further split into more homogenous groups, having a condition indicating if variable is greater or lower than threshold value, sent to right sub-tree or left sub-tree accordingly.

Each split can be referred to as nodes or inner nodes, connected to another node (branch). Splitting of the root usually terminates at the leaf, regarded as the terminal node (labelled 10 and 11 in Figure 2.1). The leaf contains results which can either be a class or a value, based on the type of tree being created (Colditz 2007). For example, Figure 2.1 is a binary tree and its nodes are sets of questions, which in turn gives either a “yes” or “no” result. The root node is denoted as node 1, while node 2 and 3, split into node 4 and 5, 6 and 7 respectively which could be terminal or node split. The numbers indicate the descendants of the next level. The tree pruning

phase removes statistical variation that may be specific only to the training set, through examining the initial tree built (Quinlan 1993).

Pruning can be used to cut unnecessary or error-prone sub-trees by defining new leaves thereby reducing tree length as it finds the optimal tree (Colditz 2007). Two strategies are often used to carry out pruning; the first is the cost complexity strategy which cuts off a rooted sub-tree with little complexity and no significant misclassification cost. The second strategy is error-based pruning which reduces tree error by computing error probability from the upper confidence limit of the binomial distribution using training data (Quinlan 1993). Having an appropriate tree size is very useful for classification accuracy.



Source: Colditz (2007: 105)

Figure 2.1 Decision tree structure with number indicating identifier of hierarchy

Several studies (Otukei & Blaschke 2010; Quinlan 1993; Sharma, Ghosh & Joshi 2013; Torma 2013; Wardlow & Egbert 2010, 2008) have been carried out on large areas using satellite data and DT algorithms with high accuracies and good classification maps. DT is functional in automatically updating land cover maps and processing of large datasets (Colditz 2007; Huth et al. 2012; Punia, Joshi & Porwal 2011) as it is fast and not sensitive to noise in the training data. A high accuracy (90%) was achieved based on using the DT algorithm for land cover classification using the S-plus statistical package (Wessels et al. 2004).

Using multi-scale analysis, the DT algorithm has proven adequate and efficient for using time-series MODIS data for land cover mapping and change detection (Colditz 2007; Knight et al. 2006). Various tree algorithms are available including iterative dichotomiser (ID3), C4.5, C5.0 and CART. ID3 was developed in 1986 by Ross Quinlan. The algorithm creates a multi-way

tree, finding for each node the categorical feature that will yield the largest information gain for categorical targets. Trees are grown to their maximum size and then a pruning step is usually applied to improve the ability of the tree to generalise to unseen data.

C4.5 is the successor to ID3 and removed the restriction that features must be categorical by dynamically defining a discrete attribute (based on numerical variables) that partitions the continuous attribute value into a discrete set of intervals (Chauhan & Chauhan 2014). C4.5 converts the trained trees (i.e. the output of the ID3 algorithm) into sets of if-then rules. These accuracies of each rule is then evaluated to determine the order in which they should be applied. Pruning is done by removing a rule's precondition if the accuracy of the rule improves without it. The J48 classifier in C4.5 software package has the advantage of using various parameters during classification such as the binary splits, confidence factor, size of tree, number of leaves, folds and objects to achieve a better result (Sharma, Ghosh & Joshi 2013).

C5.0 is Quinlan's latest version release, it uses less memory and builds smaller rule sets than C4.5 while being more accurate (Pandya & Pandya 2013). CART is very similar to C4.5, but it differs in that it supports numerical target variables (regression) and does not compute rule sets. CART constructs binary trees using the feature and threshold that yield the largest information gain at each node. The Scikit-learn Python package uses an optimised version of the CART algorithm.

The DT classification approach aided broad scale land cover mapping and better understanding of change detection using a univariate DT algorithm (Giri & Jenkins 2005). The DT supports automation and is efficient in handling noisy, missing data and non-linear relations between features and classes (Giri & Jenkins 2005). Morton, Defries & Shimabukwo (2013) favoured a DT approach in creating information for setting conservation priorities, and the evaluation of trade-offs between land use and conservation.

Multi-layer agricultural land use classification can be enhanced through DT when pruning is applied to a tree (Brown et al. 2013). Another advantage of using DT approach is the ability to edit thresholds and rules, as the user can determine the root node, internal nodes (splits) and the terminal nodes (leaves) (Pal & Mather 2001).

However, other tree algorithms are also available such as chi-square automatic interaction detectors (CHAID), the fast algorithm for classification trees (FAST) which makes use of ANOVA, the quick unbiased, efficient statistical tree (QUEST). These algorithms are computationally expensive and not vastly superior to DT (Colditz 2007).

2.3.2 Ensemble methods

The ensemble method uses a given learning algorithm such as averaging or boosting to combine predictions of several base estimators in order to improve robustness over a single estimator (Louppe & Geurts 2012) and regulate over-fitting (Pedregosa et al. 2011). Averaging differs from boosting in that during averaging several estimators are built independently and the average of their predictions used, while in boosting base estimators are built sequentially and attempt to reduce the bias of the combined estimator. Various tree algorithms can be implemented in Python using the Scikit-learn ensemble algorithms (Pedregosa et al. 2011). Scikit-learn ensemble classifiers include random forest (RF), extra-tree (ET) and Adaptive boosting (AdaBoost). The goal of an ensemble classifier is to combine the predictions of base estimators built with a given learning algorithm in order to improve the estimator. Both averaging and boosting methods are implemented with Scikit-learn. RF (Breiman 2001) and ET (Geurts, Ernst & Wehenkel 2006) are averaging algorithms, while AdaBoost (Hastie, Tibahirani & Friedman 2009) is a boosting algorithm. In the ensemble classifier, a time record for each analysis as well as cross validation information, which evaluates the estimator performance, is recorded. This holds out part of the data as a test set, and computes the cross validation score on the estimator and the dataset using the k-fold. Using the k-fold cross validation, the training set is split into smaller sets, by applying k – 1 of the folds as training data, after which the resulting model is validated on the remaining data (Pedregosa et al. 2011).

The out-of-bag (OOB) estimator generates new training sets using sampling with replacement so that, for each classifier in the ensemble, a different part of the training set is unused. This training set is kept aside to estimate the generalization error of the classification without having to rely on a separate validation set. OOB is a pessimistic estimator of the true test loss which is efficient when working with small number of trees. It can also be used to determine the optimal number of iterations and perform model selection (Bergstra & Bengio 2012). To compare the classification error of a decision stump or tree, the discrete (SAMME) and real (SAMME.R) AdaBoost algorithms are efficient. “Discrete” adapts based on errors in predicted class labels while “real” uses the predicted class probabilities factor (Hastie, Tibahirani & Friedman 2009).

When carrying out classification with AdaBoost Hastie, Tibashirani & Friedman (2009) suggest the use of small learning rate and small n_estimators because it supports better test error. Ridgeway (2007) created generalized boosted models and favoured the use of the AdaBoost algorithm to enhance classification and solve issues related to loss of function.

When working with AdaBoost a regularisation strategy is used which helps to improve the impact of a weak learner in a tree by a factor (Hastie, Tibahirani & Friedman 2009). AdaBoost is

one of the most employed boosting approaches and has been successfully used for time-series classification (Hastie, Tibahirani & Friedman 2009; Louppe & Geurts 2012; Ridgeway 2007) of AVHRR (Friedl et al. 1999) and MODIS (Friedl et al. 2002) data. According to Louppe & Geurts (2012) a high classification accuracy can be derived from an AdaBoost ensemble framework with DT, but indicated that issues of system memory might arise when working with large datasets and as such a lower bit integer (8bit) is advisable than a higher bit (32bit) integer. Land cover mapping, which involves various automated approach, will be discussed in the next section.

2.4 AUTOMATED APPROACHES FOR LAND COVER MAPPING

Models have been integrated in GIS for many years (see Huang & Jensen 1997). The purpose of the integration was to expand GIS functionality (e.g. spatial statistics, environmental modelling) while utilising the strengths of GIS databases. Maguire (2005) suggests three integration approaches: 1) loose coupling which employs common data structures; 2) moderate integration which uses remote procedure calls and shared database access; and 3) tight integration which can be achieved by object-component calls, or function calls completely integrated within GIS. In addition, visualization tools in GIS can help the user appreciate relationships between spatial variables (Mather & Koch 2011).

Automation has been used for land cover mapping in various ways using different satellite images as source data (Awwad 2003; Ozdogan et al. 2010; Verhegghen et al. 2009). Scientists have different views on what is regarded as automation (Asmat & Zamzami 2011; Comber Law & Lishman 2004). Some believe automation to be the commonly known approach of image classification such as supervised and unsupervised classification using either pixel or object-based methods (Jiang et al. 2012; Keuchel et al. 2003; Ozdogan & Gutman 2008). For others, an automated technique involves the use of machine learning algorithms such as CART and neural networks (Aitkenheads & Aalders 2011; Duong 2000; Louppe & Geurts 2012; Wehrmann, Desh & Glaser 2005).

PSU (2014) describes automation as the automatic functioning of a machine, system or process with minimal human interaction. It can also be described as a feature that allows an object that was designed for use in one application to be accessed in another application. However, for automated analysis of land cover classification to take place, there should be efficient software integration that will enable image classification. This is achievable through automation of concurrent geo-processing of multiple large datasets. Automation can also involve the creation of a new toolset or framework in order to eliminate user interaction when processing large datasets

(Asmat & Zamzami 2011; Colditz 2007; Duong 2000; Huth et al. 2011). This latter definition of automation is used in this study.

The following sub-sections provide an in-depth explanation of the two approaches for automated land cover mapping, starting with conventional image analysis as an automation technique and then discussing creation of an enhanced toolset.

2.4.1 Conventional image analysis

Conventional image classification can be regarded as being automated (Jiang et al. 2012; Keuchel et al. 2003; Ozdogan & Gutman 2008), since human interaction is eliminated by computerised image analysis. Awwad (2003) carried out a comparison between manual digitizing and automated classification of an aerial photograph using supervised classification and observed that supervised classification yielded higher accuracy than the manual digitization. Verhegghen et al. (2009) also performed automatic labelling of images by reducing the land cover legend. This was conducted in order to reduce the processing time and capital involved in carrying out such analysis. An automated technique was used to match old manual aerial photographs with satellite imagery and expert knowledge from aerial photograph interpreters (API) was used. The technique was said to be inexpensive, could perform change detection and was used for monitoring (Comber, Lawanr & Lishman 2004). Jiang et al. (2012) used a method that semi-automatically detects land cover changed pixels from satellite images compared with a prior land cover map. In addition, it automatically classifies the changed pixels based on pattern recognition and change rules. This method automatically extracted training samples with GIS and statistical technology rather than the conventional way of manual training sample selection, which is labour intensive when working with large datasets.

2.4.2 Creation of enhanced toolset

Various studies have emerged describing the creation of a new environment or new toolset to ease the manual process of image analysis (Huth et al. 2012). There is a high demand for automation of image classification due to increasing volumes of accessible data and the need to process reliable results in a shorter time (Cihlar 2000; DeFries & Chan 2000; Knorn et al. 2009; Rogan et al. 2008).

Huth et al. (2012) created a framework, referred to as twined object & pixel-based automated classification chain (TWOPAC), for automated land cover classification. The framework has the potential to classify imagery from different sensor types, using both pixel and object-based classification. It uses DT classification, calculates the accuracy of the classification, stores information in a database, and places the tool on a server to make it accessible to users. These

functions are embedded into TWOPAC using Python code and made available via a graphical user interface (GUI) (Huth et al. 2012).

Tiede, Luthje & Baraldi (2014) created a post-classification comparison (PCC) system referred to as geospatial services in support of European Union (EU) external action (G-SEXTANT). This system aids automatic post classification of change detection in irrigated areas. The system was faster, more accurate and had minimal co-registration errors.

Asmat & Zamzami (2012) performed semi-automated detection of settlement boundaries based on different densities using a custom-built house detection algorithm, extraction and delineation technique which achieved a faster and more reliable result. In the study data acquisition, processing and analysis were automated (Asmat & Zamzami 2012). Duong (2000) developed a tool that defines the naming of land cover classes based on image invariants for automated classification. The tool uses graphical analysis of the spectral reflectance curve (GASC) to define the image invariant and automatically assigns a code to each component (single, multi-temporal and auxiliary). Pedregosa et al. (2011) developed a Python module (Scikit-learn) that integrates machine learning algorithms for supervised and unsupervised classification with the aim of reaching non-specialist processing of large datasets.

GIS software packages such as ArcGIS (ESRI 2013), QGIS (Larocque, Bhatti & Arsenault 2014) and GRASS (Furlanello et al. 2003) provide access to geo-processing tools to facilitate automation. By using a graphical user interface, applications can easily be built by combining geo-processing tools, their execution can be automated and results displayed (Dobesova 2011). However, there are relatively few integrated software platforms, where different applications are integrated within the same application framework to efficiently perform customized analyses and display results in different scales or formats or conduct complex numerical analysis (Brandmeyer & Karimi 2000; Larocque, Bhatti & Arsenault 2014).

For this type of analysis, a customised workflow approach must be designed and implemented on a suitable platform often requiring customization using scripts (Dobesova 2011). The Python scripting language (docs.python.org), a freeware programming language can be used for this purpose (Dahal & Chow 2014; Kraft et al. 2010) and is readily integrated within various GIS software packages. According to Dangermond (2009) “Python is rapidly becoming the accepted standard for scientific programming, and its integration will bring a lot of advances in geographic science”. Yang et al. (2014) used Python to create a model for multi-target land use change simulation which was based on cellular automata. This model could simulate mutual transformation of multiple land use types. In a South American study (Giri & Long 2014), land cover characterization and mapping involving large datasets was required. Automated

geo-processing of the datasets was carried out using a Python library (mapPy) reducing cost and increasing processing. In a bid to create custom tools that are user friendly to non-GIS users and robust to perform complex geo-processing tasks, Kaunda-Bukenya et al. (2012) created a spatial decision support system (SDSS) with the aid of a Python-based graphical user interface (Tkinter) to provide faster solutions to environmental impacts of land use decisions. Having discovered the importance of automation and the possibilities of integration within GIS, the next section discusses the importance of ancillary data in land cover classification.

2.5 ANCILLARY AND REFERENCE DATA

Ancillary data are additional data collected independently by means other than remote sensing, which increases available information for distinguishing classes and performing other types of analysis (Campbell & Wynne 2011). Ancillary data can be used during classification combined with remote sensing layers as additional input (Rogan et al. 2003) or post classification for validation. For instance, ancillary data such as forest inventory data and topographic maps can be used to enhance classification in detecting forest change (Desclee, Bogaert & Defourny 2006). Ancillary data can assist in distinguishing features indiscernible on raw imagery (Stuckenberg 2012).

The type of reference data used to train the classifiers will influence classification outputs. The importance of accurate reference data to improve classification accuracy was emphasised by Ismail & Jusoff (2008) and Manandhar, Odeh & Ancev (2009). Erroneous reference data practically affects the accuracy of classification outputs. Therefore, users must ensure that reference data are representative because classification output is only as good as reference data (Foody 2002). The quality of thematic maps derived as classification output must be assessed and expressed meaningfully. This reflects the quality of the classification and its fitness for a particular purpose as well as understanding error and its likely implications (Foody 2002). This will be fully discussed in the next section.

2.6 ACCURACY ASSESSMENT

To ascertain the validity of the thematic map produced from image classification, an evaluation should be carried out to measure if the classification represents what is observed on the ground. The purpose of accuracy assessment is to quantify and identify mapping errors, and as such it is important as it shows the quality of a map. It also helps to determine if a new method or technique used produces better results than other methods (Congalton & Green 2009).

Validating a map created from remotely sensed data can be carried out using two types of accuracies, namely thematic accuracy and positional accuracy which can be evaluated separately

or together as they are inter-related (Congalton & Green 2009; Kohl, Magnussen & Marchetti 2006). Positional accuracy refers to the location of a point in the imagery with reference to its physical location on the ground, which is influenced by topography, sensor characteristic as well as viewing angles. Thematic accuracy refers to the accuracy of a mapped category at a particular time compared to what was actually observed on the ground at that time. The reference data must be completely accurate in order to achieve a fair assessment (Congalton & Green 2009; Kohl, Magnussen & Marchetti 2006). Positional and thematic accuracy are interlinked when working with a sampling scheme, especially when GPS are used to collect data and when sample size and unit are considered to be able to relate pixels into the correct sampling unit (Congalton & Green 2009).

In conducting accuracy assessment, three steps are essential: 1) design the assessment sample, 2) collect data for each sample (reference data & map data) and 3) analyse the result. The use of reference data offers more accurate representation of data being analysed than map data (Stuckenberg 2012). The overall accuracy of a classification can be described as a measure of match between classified and reference data without considering the errors of commission and omission (Zhao et al. 2012). Two techniques can be used to evaluate the accuracy of time-series generation and classification of MODIS data namely; a "hard" accuracy assessment which uses an independent sample set (selected homogeneous pixels), and the "soft" accuracy assessment which uses fuzzy reference (error matrix which summarizes the correspondence between the map labels assigned to the pixels and the corresponding ground condition) and classification (Colditz 2007). Reference data can include ground control points, field data, and aerial photographs. Frequently used is the confusion matrix algorithm which generates an error or confusion matrix for accuracy assessment. The confusion matrix tabulates and evaluates classification by comparing positions and classes of the reference data with the classified products (Exelis 2013). The confusion matrix is explicit in the evaluation of a classification, because user's and producer's accuracy, error of commission and omission and the kappa coefficient can be easily calculated from it.

The producer's accuracy is a measure indicating the probability that an image pixel classified corresponds with its ground truth pixel, while the user's accuracy is a measure indicating the probability that an image pixel is classified into a class based on the class the user specifies. The user's accuracy relates to error of commission, where a class pixel is erroneously included in another class, while the producer's accuracy relates to error of omission where a class pixel is omitted from its supposed class (Exelis 2013; Zhao et al. 2012). The kappa coefficient reflects the difference between actual classification agreement and the agreement expected by chance

which is rated between -0 and 1. For example, a classification with a kappa of 0.8 means that there is 80% better agreement in the classification than by chance while a kappa of 0 means no agreement in the classification (Congalton & Green 2009). The accuracy of a classification will determine the legal standing of the maps and their validity as a foundation for research (Campbell & Wynne 2011).

2.7 CONCLUSION

In this chapter literature was reviewed which described the integration of GIS and RS for land cover mapping. The use of multi-spectral imagery was discussed as being useful to analyse land use and land cover (LULC). Various image analysis approaches for land cover classification were also discussed. Approaches for automated land cover mapping were reviewed followed by machine learning approaches. The role of ancillary and training data in providing additional information needed for accurate classification was discussed. The chapter concludes with insight on accuracy assessment and its importance for map validation. The next chapter provides an in-depth description of the requirement analysis and technological considerations for creating an automated tool.

CHAPTER 3: REQUIREMENT ANALYSIS AND TECHNOLOGICAL CONSIDERATIONS

This chapter describes the requirement analysis as well as some other technological considerations for the creation of the automated tool. It is essential to describe the geo-processing data structure and the available platform in which geo-processing can be carried out. The ArcGIS platform was selected as ArcGIS enables seamless integration of different software through ModelBuilder, ArcToolbox and Python scripts (ESRI 2013). The motivation for using ArcGIS is provided in this section. As Python has been integrated into ArcGIS, it is easier to write scripts linked to tools that can call different software packages, when compared to many other GIS software. The significance of data standards for data sharing and why it is essential for tool sharing must also be considered, as the purpose of the new tool is to share it with the scientific and GIS communities to facilitate land cover mapping. The numerous advantages of performing geo-processing in an ArcGIS-Python environment will also be discussed.

3.1 REQUIREMENT ANALYSIS

As with any customized tool development, the design of the automated tool was preceded by a requirement analysis, which determines the structure and essentials of the tool as well as the identification of software routines, while being mindful that the tool is designed to solve a specific problem (Van Niekerk 2008). By examining the process of manual land cover mapping (Adesuyi & Münch 2015), potential automation needs were identified giving rise to system requirements. The software platform determines operational requirements, with new functionality introduced through integration of existing geo-processing tools and Python modules. Given that geo-processing is described as the manipulation of spatial data through operations within GIS (ESRI 2013), land cover classification can be regarded as a workflow of sequential geo-processing tasks. ArcGIS provides various ways for users to automate their geo-processing tasks by combining existing tools in a model or combining built-in tools with short computer programmes or scripts. The GIS software developer Environmental Systems Research Institute (ESRI) emphasizes Python in its documentation and includes Python with the ArcGIS installation. ArcGIS was therefore selected. In order to implement the DT and ensemble logic in a Python framework, existing scripts, data and examples from the Python Scikit-learn modules (Pedregosa et al. 2011) must be examined.

The system requirements for the automated tool can be divided into functional needs (i.e. what the system should do) and operational characteristics (i.e. how the system should do it). In addition, there is also a strong reliance on the software and data requirements (Van Niekerk

2008). Once the requirements have been determined, the specification of system requirements can take place.

3.1.1 Functional needs

The functional prerequisites of the automated tool are directly related to the objective of the study which states that the tool must enable a user to perform an automated image classification. Automation in this context is defined as the elimination of manual user interaction for land cover mapping thereby reducing processing time and analysis cost, as well as potential human and processing errors in order to achieve high accuracies with large datasets (Asmat & Zamzami 2011; Colditz 2007; Duong 2000; Huth et al. 2012). To perform land cover classification, the user must be able to:

- collect and prepare the data;
- identify the features needed;
- create training data;
- specify the different classification parameters or rules;
- carry out image analysis;
- summarize and validate; and
- create a suitability map.

3.1.2 Operational characteristics

The operational characteristics of the automated tool describe how it can be accessed, executed and presented. The tools should be user-friendly and flexible allowing multiple users simultaneous access. The user must be able to select their desired data and explore different aspects of the data. Different parameter and algorithm selections should be available and the user should interactively be able to see the effects of the different parameters used in the output map(s) created.

The tools should be executable from within the selected software framework and for this study a customized toolbox in ArcGIS was selected. For the purpose of this study geo-processing tasks will be automated in the customized toolbox created using ModelBuilder and Python (PSU 2014). An ArcToolbox is a collection of tools, structured into various toolsets within each toolbox, providing ArcGIS with the necessary analytical tools (ESRI 2013). ModelBuilder can be described as an interactive visual program that allows a user to chain tools from ArcToolbox together using the output of one tool as input in another. This facilitates a flow chart pattern.

ModelBuilder is a tool for creating a customized ArcToolbox and an automated workflow for image analysis.

The strengths of a ModelBuilder includes visual understanding of how the particular model works, automation of complex GIS workflows without programming skills and exporting of any model created to a Python script. Some of the limitations include clumsy, bulky models created in a ModelBuilder and dependency on an ArcGIS license restricting access to the tool (ESRI 2013; Pimpler 2013; PSU 2014). Additional functionality is added to a ModelBuilder via the ArcToolbox. By creating custom tools through scripting, Python can execute sequential steps, combine existing components or even iterate over sequences (ESRI 2013).

For the sake of flexibility, the toolbox should contain separate toolsets, each addressing one of the functional needs which can be executed individually by the user, or combined in a single process from a model, thereby completely automating the process. Each tool within the toolsets must be parameter-driven through interactive dialogue boxes. Each dialogue box should have a description option which will provide users with concise detail of each function within the toolset. An error message indicating an improper selection along with suggested solutions should be embedded in the tool. Moreover, customised help files can also be compiled.

As one of the functions of the toolbox is to implement image analysis using DT and ensemble classification, additional software requirements to enable this functionality are required. An important requirement for MEAWAT is to find a DT algorithm based on C.45/J48 that could easily be integrated into ArcGIS in order to produce high quality land cover classification results. ENVI and ArcMap have existing integration capabilities making ENVI a suitable candidate. However, due to the potential problem with proprietary licensing of ENVI, an alternative DT classifier to be utilized in MEAWAT is required. The use of the Scikit-learn Python module for this purpose will be highlighted in a subsequent sub-section.

3.1.3 Strengths of geo-processing in Python

Building geo-processing tasks using Python can support the data processing requests of multiple, concurrent remote users by leveraging centralized processing resources and therefore increase productivity (ArcGIS 2010). With the shift away from Visual Basic for Application (VBA) to Python as the integrated scripting language in ArcGIS, there have been continuous up-to-date and new feature implementations in ArcGIS (Wunderlich 2012). Python enables interoperability, which helps to streamline the overall analytical process. Python also provides built-in structures through dynamic typing and allows functionality to be extended using modules (Ajay 2013).

The integration of Python into ArcGIS is useful for GIS practitioners (Toms 2015). The ArcPy Python module is a wrapper module that allows programmers to use Python to interface with the core of ArcGIS (Toms 2015). This gives the programmer access to all geo-processing tools available in ArcGIS. In addition, custom modules can be built using Python and then integrated into an ArcGIS geo-processing stream which is difficult in other platforms and software (ESRI 2013). Python is easier to learn when compared to other programming language such as C++ or Visual Basic (Zandbergen 2013).

Python provides cross platform operating on a variety of operating systems such as Linux or Windows (ESRI 2013). It is a free and open source software (FOSS) and as such, is constantly being improved upon by a dedicated user community. It is an interpreted language in contrast to C++, JAVA and Visual Basic and does not need compilation to binary code for the computer to interpret and run (Zandbergen 2013). It enables easy programme and software integrations such as with the geospatial data abstraction library (GDAL/OGR) which has Python bindings (Python 2014; Zandbergen 2013). The modular nature of Python allows new modules to be created to extend the language with new or legacy code. There are a substantial number of extension modules, called packages that have been developed and are distributed by members of the Python user community (Sanner 1999). One such package is the Python Scikit-learn module which makes machine learning routines available in Python (Pedregosa et al. 2011).

Python has some limitations compared with other programming languages, including limited editing and debugging capabilities. In addition, Python cannot respond to events within the ArcGIS framework (Python 2014). It also does not enable the creation of a custom user interface tied directly to the application and not all components of ArcGIS are exposed in Python (Zandbergen 2013). Despite some limitations, the strengths of Python in combination with various built-in geo-processing tools available in ArcGIS, as well as the wide use of ArcGIS in South Africa with 880 delegates at ESRI Africa User Conference 2015 (EE Publishers 2015), provided enough motivation for selecting ArcGIS as development platform.

3.1.4 Data and software requirements

The automated tool must be user-friendly thereby providing a platform for user interaction. As specific input data are required for the tool to function effectively, these datasets in addition to the raw imagery to be used for the land cover classification, must be specified by the user. The required datasets are specified in (Table 3.1). All data must be projected and provided in the same projected coordinate system.

Table 3.1 Data requirements for the automated tool

Additional input data	Format
Study extent mask	Projected data of study area extent (shapefile or raster)
Training dataset	Point shapefile with known classes
Validation dataset	Point shapefile with known classes

The automated tool will require certain software packages to be installed on the user's computer. ArcGIS will be the only proprietary component of this tool and a Spatial Analyst license is also needed for raster analysis. Python is shipped with ArcGIS, but can also be downloaded without charge. The user must ensure that the correct versions of ArcGIS and Python are installed. In order to integrate a classifier similar to the WEKA DT classifier (C.45/J48) tested by Adesuyi & Münch (2015) for agricultural land cover classification using NDVI time-series data, the Scikit-learn Python module that uses a modified CART algorithm was selected (see Section 2.3.1). In addition, Scikit-learn gives access to various ensemble classifiers as described in Section 2.3.2. Though the possible use of ENVI DTs was investigated, the proprietary nature of the software hindered implementation in this study.

When considering all the components required for the automated workflow, the acronym MEAWAT was assigned representing multiple ensemble classifiers in ArcGIS workflow automation tool. The software requirements for MEAWAT are provided in Table 3.2.

Table 3.2 Software requirements for MEAWAT

Software Requirements	Available from
ESRI ArcMap 10.1 and above with Spatial Analyst license	Purchase from ESRI
Python 2.7 and above	Shipped with ArcGIS
NumPy 1.6.2 and above	http://sourceforge.net/projects/numpy/files/
Scipy 0. 16 and above	http://sourceforge.net/projects/scipy/files/
Scikit-learn 0.15 and above	http://scikit-learn.org/stable/
Matplotlib 1.4.3 (if not installed with Python)	https://pypi.python.org/pypi/matplotlib

Having explained the requirements needed for MEAWAT to operate effectively, some other important technological considerations include data structure, data sharing and tool sharing. Data structure will be discussed in the next section.

3.2 DATA STRUCTURE

A data structure is a specialized format for organizing and storing data. General data structure types include the array, the file, the record, the table and the tree. According to Zandbergen (2013) data structure can be described as the arrangement of data. Data structures are designed to organize data for a specific purpose and allow access and operation on the stored data in a prescribed manner. In computer programming, the data structure selected can impact on the performance of the particular algorithm (Mitchell 2003). Using NumPy, satellite images, accessed in GeoTIFF format using the ArcPy interface in ArcGIS, can be expressed as multi-dimensional arrays. This allows seamless integration with the Scikit-learn machine learning libraries required for classification. Representing an image as a NumPy array is not only computational and resource efficient, but also facilitates numerical analysis using NumPy's built-in functions. The resultant classified data would have to be converted to image file format after classification for visualization in ArcGIS. This illustrates the importance of efficient and logical computer programming code using appropriate data structures when performing automation and integration (Zandbergen 2013).

For data to be exchanged within a geospatial context, certain standards have to be adhered to as this can have a substantial influence on how data can be shared. A discussion about data standards is presented in the next section.

3.3 SIGNIFICANCE OF DATA STANDARDS FOR DATA SHARING

To enable data sharing within or across the geospatial community, data must meet certain accepted standards. These standards are rules, guidelines and conditions for products or processes and production methods must conform to the International Organization for Standardization (ISO) policies (OGC 2013). An example of such standards are ISO 19115 for geographic information–metadata and ISO 19115 for geographic information services. Data standards help to ascertain the completeness and accuracy of information. This enables the exchange of geospatial information and instructions for geo-processing. Adhering to data standards can also improve the quality of data by subjecting data to conditions and restrictions, and as such only the owner can claim ownership of the data.

Data standards encourage data availability at an affordable cost to the public, thereby facilitating data sharing. In situations where private data are being used, the owner must be credited or

acknowledged (ESRI 2003). GIS data standards allow interoperability which supports data integration and sharing among different platforms and across applications (ESRI 2003). In the GIS context, there is a move towards implementation of open standards, which will further ease sharing of data through supported formats and provision of metadata for each dataset. In other words, GIS needs to support platform independent solutions, so it can be accessed through different platforms and devices, hence the move away from the geo-relational database system to GIS-based web services framework (ESRI 2003). As the purpose of a new workflow automation tool is to share it with the scientific and GIS communities, the importance of tool sharing is considered in the next section.

3.4 ESSENTIALS OF TOOL SHARING

There are certain characteristics or components that an ArcGIS tool should exhibit before it can be shared. A toolbox within ArcGIS should have metadata before it can be shared with the public (ESRI 2013). Tool sharing in ArcGIS is efficient, as a result of ESRI's support for standard metadata representation, and their products support interoperability and web services standards that enable integration of various GIS services from different GIS vendors (ESRI 2003). A toolbox should therefore support GIS interoperability. With the advent of geo-processing packages the user can distribute shared tools and files conveniently (Zandbergen 2013).

Depending on the complexity of the tool created, the tool should have scripts, sample data, documentation and compiled help files (html) before it is shared. A common way of sharing a customized toolbox is by using a local area network to publish the toolbox on a ArcGIS server thereby making it open for users to execute. An alternative is to use a standard folder structure and making all the files available in a zip file (ESRI 2013). Appropriate toolbox documentation will entail background details of the tool which will allow the user to understand why and how to execute the tool. It is also important to make use of relative paths for file locations when constructing the toolbox. The tool should also have a scratch workspace (temporary files folder) and a tool data folder (where all the toolsets are stored). ArcGIS toolbox and ModelBuilder provide excellent documentation and metadata functionality (Zandbergen 2013).

3.5 CONCLUSION

This chapter explored the technical requirements for creating an automated tool, which will be used to generate a land cover map by integrating different workflow steps in an automated fashion. This was needed in order to understand how the design of the new tool should proceed. The strengths of geo-processing using a Python platform, particularly within ArcGIS was considered. The chapter concludes with the significance of data standards and the importance of

tool sharing. The next chapter discusses the system design and implementation in ArcGIS using ModelBuilder, ArcToolbox and Python.

CHAPTER 4: SYSTEM DESIGN AND IMPLEMENTATION

This tool development chapter comprises three major sections namely, 1) proposed workflow for MEAWAT; 2) preparation of the datasets; 3) creation of the tool using Python. This chapter describes the preparation of the dataset for analysis and methods for the image classification. The concluding paragraphs provide concise overview how evaluation of the analysis was carried out in MEAWAT.

4.1 PROPOSED WORKFLOW

An effective toolbox for generating an automated land cover classification model using time-series NDVI requires a logical workflow (Figure 4.1). Each image classification step illustrated in Figure 4.1 must flow in a systematic way to the next step for further analysis. Before image analysis can take place, the relevant imagery must be collected. The details collected (both the imagery and ancillary data) will be discussed in Chapter 5.

The first step is preparation of the MODIS and/or Landsat imagery. This process is essential to prepare the imagery in the correct format for input to MEAWAT (planning phase). Once the first step is completed, the second step starts with the creation of the MEAWAT toolbox and toolsets which include the functionalities of image resampling and projection, extracting features needed from the image, building a phenology tree from the training data and visualising it so that the data with errors or ambiguous NDVI values can be removed from the training set.

The efficiency of the image analysis technique lies in selecting appropriate training data which is facilitated by the “Create training sample” step in Figure 4.1. The third step leads to image processing where a supervised machine learning model will be used to identify two agricultural land cover classes (*Wheat* and *Vineyard*), while all other land cover classes will be classified as *Others*. The purpose of selecting *Wheat* and *Vineyard* is to demonstrate the ability of the tool to identify a winter crop (*Wheat*) and a summer crop (*Vineyard*).

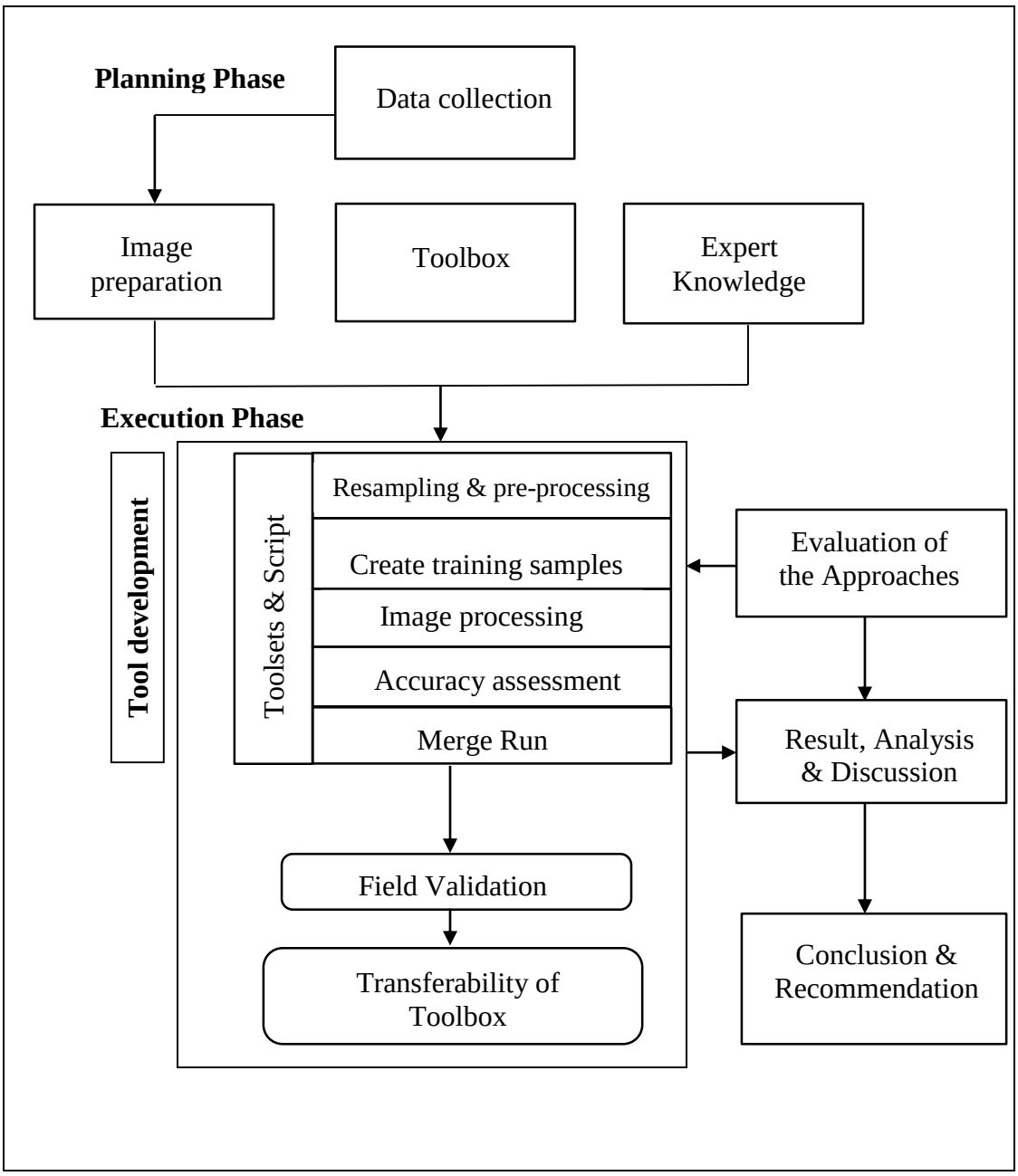


Figure 4.1 Workflow design

This is followed by the fourth step, which involves evaluation (accuracy assessment) of the output generated from image processing. In addition, a dynamic toolset that combines the functions in all the other toolsets (Merge Run) is presented.

During the execution phase, each of the different classification techniques was evaluated and they are subject to change as the image analysis process continues. By using reference data (field validation & ancillary data) the accuracy of the result was established. Subsequently, the transferability of MEAWAT was executed by executing the tool on multispectral Landsat imagery. The following section gave an explicit explanation on how the dataset was prepared for the study

4.2 DATASET PREPARATION

Image pre-processing is an essential part of image analysis. It is intended to correct for sensor and platform-specific radiometric, atmospheric effect and geometric distortions of data which in turn improves the quality of the image (Campbell & Wynne 2011). Despite lower resolution, MODIS NDVI data proved suitable in land cover studies (Sub-section 2.1.1). As described in Sub-section 2.1.2, Landsat capacity to map a medium geographical area such as the agricultural section of the Berg River catchment also makes it suitable for this study. The MODIS and Landsat images have different spatial resolutions and scene sizes. Therefore, the images need to be resampled to fit into the study extent. Below is an in-depth description of the preparation of the two datasets. Pre-processing will generally take place prior to executing MEAWAT.

Manual pre-processing required for MODIS data using the MODIS reprojection tool (MRT) is described in this section. It is a specific functional requirement for MEAWAT to integrate this step and highlight the differences expected when comparing MEAWAT and MRT.

MODIS gridded data products, MOD13Q1 in particular, have undergone radiometric, atmospheric corrections and also corrected for bidirectional reflectance distribution function (BRDF). MODIS land products use the sinusoidal grid tiling system and are downloaded from land processes distributed active archive centre (LPDAAC) in hierarchical data format (HDF). The data must be converted and projected to GeoTIFF file format to ease access for analysis through ArcMap and other spatial analysis software. The MRT was developed for this purpose (Dwyer & Schmidt 2006) (Figure 4.2).

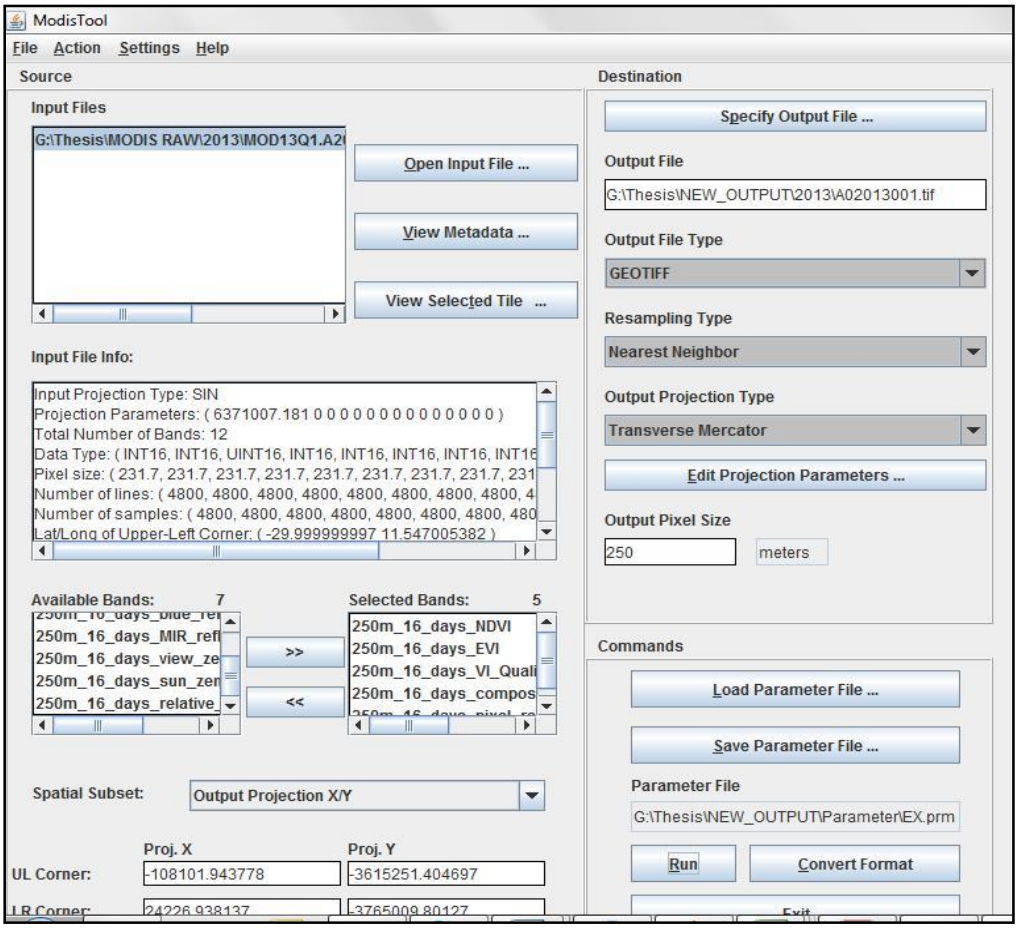


Figure 4.2 MODIS reprojection tool

MRT is efficient in converting large volumes of data by selecting the number of bands needed, spatial subset required, resampling method, output cell size and desired projection. In addition, the MRT tool parameters can be saved into a parameter file and run in batch mode.

Before image processing can take place, the study area must be selected and all images for the particular year stacked into a single image. This would traditionally be done in another software package, e.g. ENVI or ArcGIS but this was embedded into MEAWAT to reduce the possibility of user error. To concentrate on only the areas used for agricultural land cover in this study site, water, townships and Cape Nature protected areas were masked out. This produced the required study area input data set as specified in Table 3.1. Functionality of the MRT was therefore embedded into MEAWAT through a custom Python script using the ArcPy wrapper to ArcGIS geo-processing tools. The output data from the execution of the resampling and pre-processing toolset in MEAWAT was compared with that of the MRT. Landsat data preparation is discussed in the next paragraph.

PCI Geomatica (2014) and IDL ATCOR 2 (2015) (which does not require the input of an elevation model) were used to correct radiometric and atmospheric effects in the Landsat imagery covering the study area. Each Landsat image was downloaded as an OLI file with

separate bands and stacked to an ENVI BSQ file in ATCOR 2. Metadata and calibration details of the multi-spectral (MS) image (offset and gains) were obtained using PCI Focus and saved as a calibration (cal) file. ATCOR 2 was then executed to perform the radiometric and atmospheric correction on each image. Settings used include: the predefined sensor flat terrain; satellite geometry; visibility (50-70 km); ground elevation (0.3); and scale factor (4). All images affected by cloud cover were removed from the dataset resulting in 12 images instead of the expected 22 images captured for 2014. These images were removed from further analysis because cloud cover interfered with pixel values which will have a negative influence on the outcome of the classification. Preparation of the reference data used for this study is explained in the next paragraph.

Field data and the SiQ crop data obtained from the Department of Agriculture, Western Cape, were used as reference data. In 2013, SiQ, a private company specializing in crop mapping, produced an agricultural census dataset for the Western Cape. This contained crop information for each individual cultivated field. The data were generated through an extensive aerial survey combined with supporting field surveys. The agricultural census dataset includes fruit orchards, vegetables as well as annual and perennial pastures.

In an attempt to reduce the mixed pixel effect, intensified by the use of medium resolution imagery such as MODIS (Yang et al. 2015), only homogenous cultivated fields of larger than 6.25 ha which would completely cover a single MODIS pixel, were selected to represent the selected agricultural land cover classes. Of the 45 000 polygons in the SiQ database, only 16 197 polygons were therefore eligible for selection. MODIS raster data were converted to polygon and intersected with selected field boundary data. The percentage of each MODIS pixel within the field was then calculated. Observation points were selected based on a percentage of the 6.25 ha greater than 80% of areas within field covered by a single MODIS pixel. MEAWAT toolbox design is explained in the next section.

4.3 CREATING MEAWAT

MEAWAT was designed to examine the potential of using a custom toolbox in ArcGIS containing a number of toolsets to automate land cover mapping. In a custom toolbox, parameter definitions, validation code, and the source code are handled in the same interface, making it easier to create and maintain Python tools. However, a Python toolbox cannot contain ArcToolbox geo-processing tools. A custom tool can be used to link various geo-processing functions as well as perform other operations using Python programming. In addition, tool parameters can be defined through an interactive wizard and the validation code is stored in the

toolbox itself, while the Python source code is stored in a separate file. For the purpose of this study, the custom toolbox included Python script tools and a model tool built with ModelBuilder.

Figure 4.3 gives an example of a graphical user interface (GUI) that can be created using the wizard to interface with any new tools created. Each parameter in the GUI represents a parameter passed to the Python script or ArcGIS geo-processing tool.

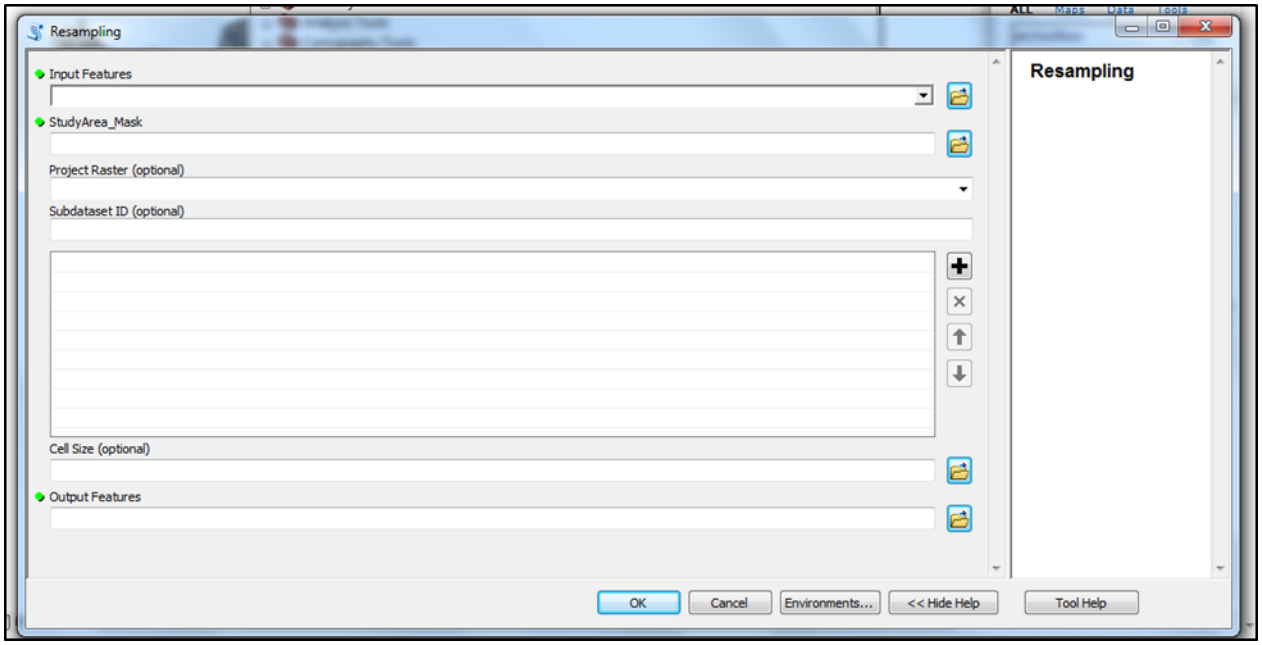


Figure 4.3 Parameter interface for the new toolbox

The MEAWAT toolbox includes toolsets to accommodate each of the steps in the analysis described in the workflow design (Figure 4.1). These include *1. Resampling & pre-processing, 2. Creating training samples, 3. Image processing, 4. Accuracy assessment* and *5. Merge Run*. Each of these five steps has one or more Python scripts associated with the corresponding tool in the toolset. These five steps and associated tools are shown in Figure 4.4. The last toolset within MEAWAT, referred to as *Merge Run* was created to execute all the steps of the image analysis thereby providing complete automation of the process.

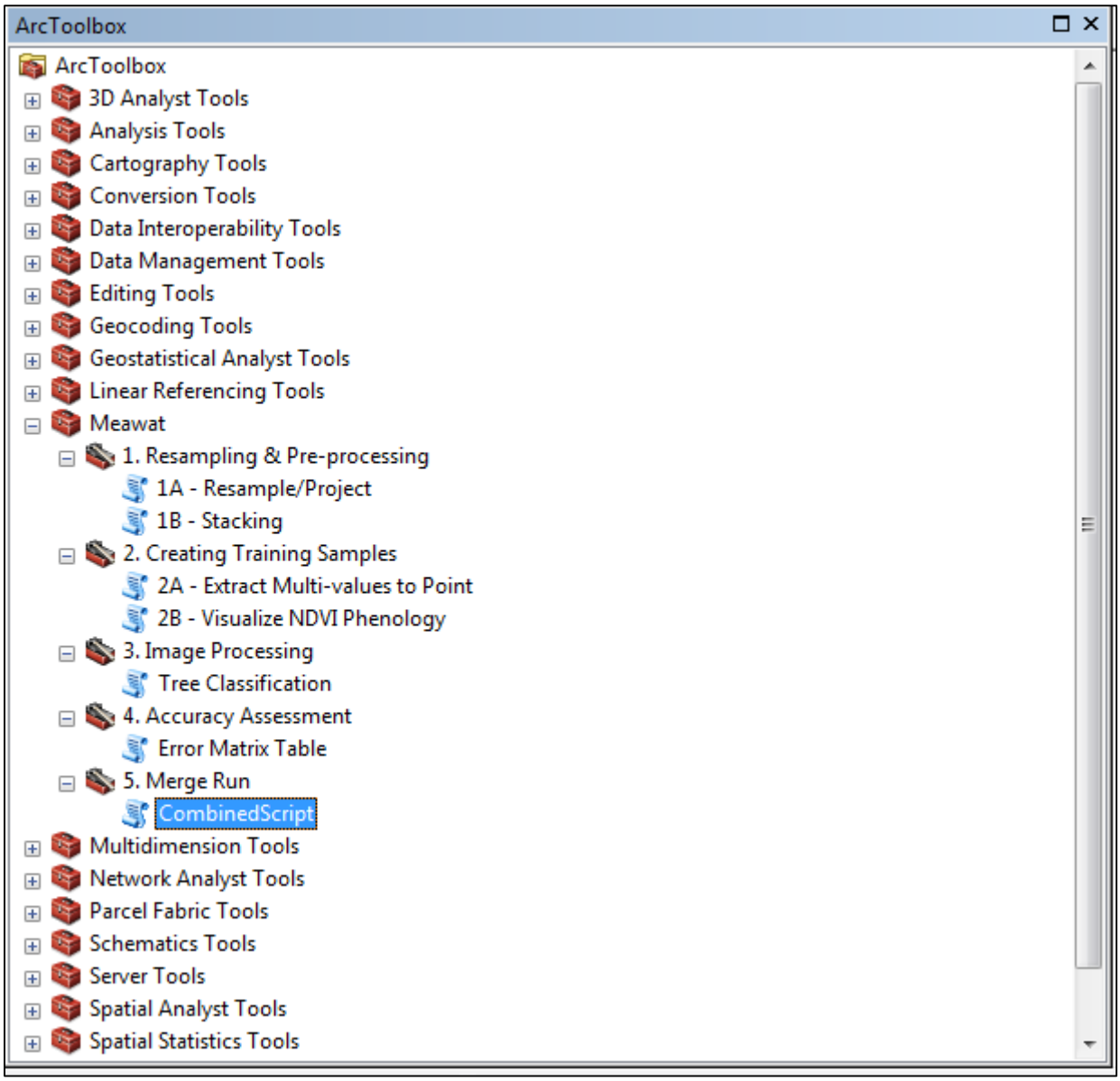


Figure 4.4 MEAWAT and its toolset

To facilitate tool sharing and following standards recommended by ESRI (2013) the MEAWAT toolbox was created in its own folder structure. The folder includes the toolbox itself, a script folder which contains all the different scripts used in MEAWAT and a map document (mxd) file that allows easy navigation to the toolbox in ArcMap. It also contains a scratch folder where temporary working files are deposited for re-use during script execution and a tool data folder, where different files used alongside with the tools are stored. A document describing details about each script in MEAWAT is available upon request. Having given background on the MEAWAT toolbox characteristics, the next sub-sections will discuss in detail the various techniques associated with each of the toolsets used in Figure 4.4.

4.3.1 Resampling & Pre-processing

The resampling and pre-processing toolset in the MEAWAT toolbox contains two custom tools: *1A - Resample/Project* and *1B - Stacking*. For 1A, a Python script was created to integrate the functionality of the MRT into MEAWAT, linked to an ArcMap tool interface (Figure 4.5). The script facilitates transformation of the raw MODIS imagery (HDF format) into GEOTIFF format in ArcMap. HDF stores multiple objects (subdatasets) within one file. ArcGIS is capable of reading HDF4 and HDF5 data based on a raster data model using a built-in tool (Extract Subdataset) (ESRI 2013). For the MODIS VI product, these subdatasets represent NDVI, EVI and various quality flags, as shown in Table 2.2. This is implemented in MEAWAT as a subdataset index (Figure 4.5) starting at 0. If no subdataset is chosen, the tool will use the default subdataset 0, which in this case indicates NDVI.

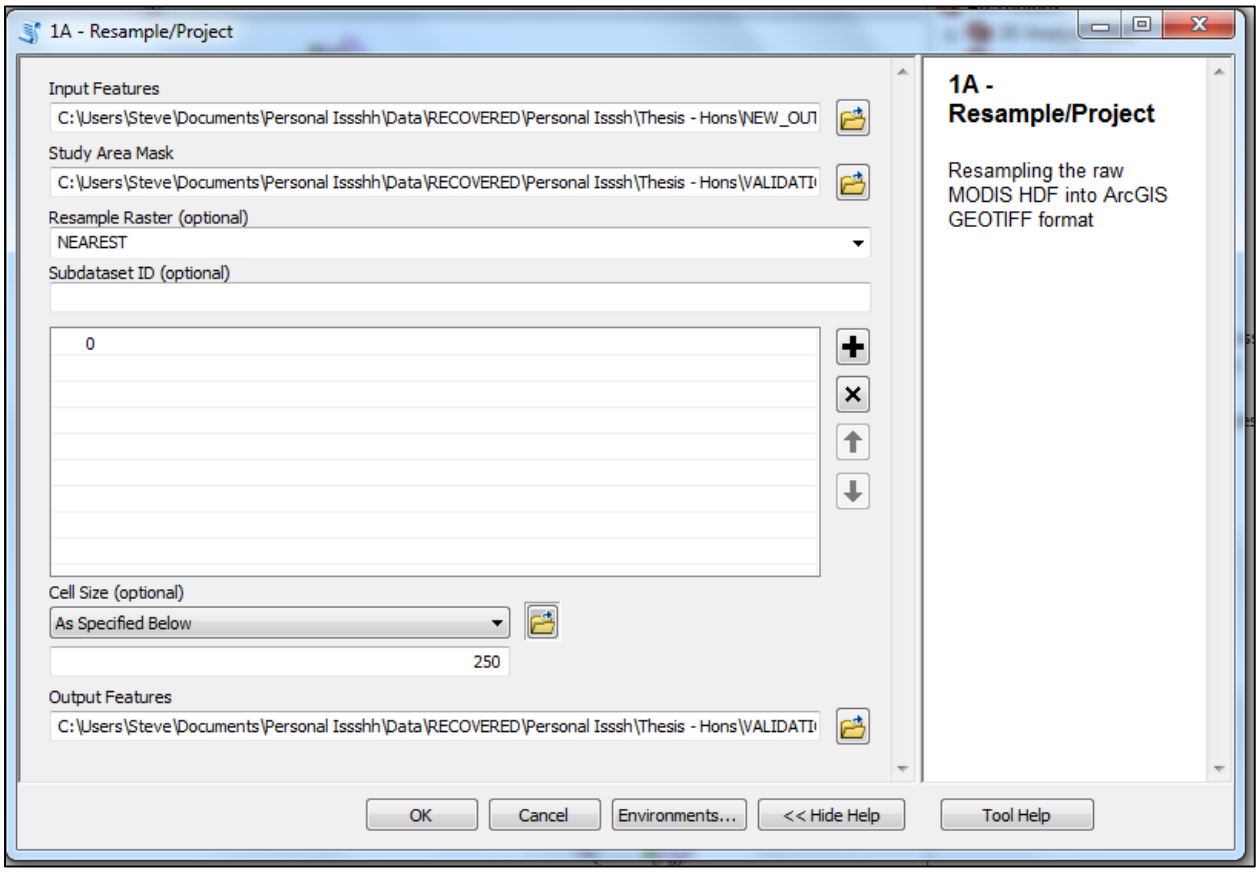


Figure 4.5 Resampling and pre-processing interface (Tool 1A)

To execute the tool when working with multiple large datasets, the user must supply both the input feature folder (location of the raster image files) and the output feature folder (where the resampled image files will be saved). The tool also requires an already projected study extent mask. This serves as input to the "Extract by Mask" geo-processing tool. The user may optionally choose a resampling method, but if not chosen the tool will use the default of nearest neighbour resampling.

The ArcGIS geo-processing tool “ProjectRaster” is used to fulfil the functionality of assigning a projection to the image. The user must also select an output cell size and the number of the subdatasets needed for their analysis. The output files are stored in GEOTIFF format (Figure 4.5) and they represent selected products of resampled reprojected data for the study area extent. Output files can be saved in a temporary folder for easy access (optional). For the purpose of this study, only the NDVI files (subdataset 0) were selected for processing.

Tool *1B* (Figure 4.6) stacks images to produce one composite image. The Python script for this function picks the images from a user-specified (or temporary folder) and stacks them together based on user-defined input such as the number of bands that would be stacked together in a large dataset.

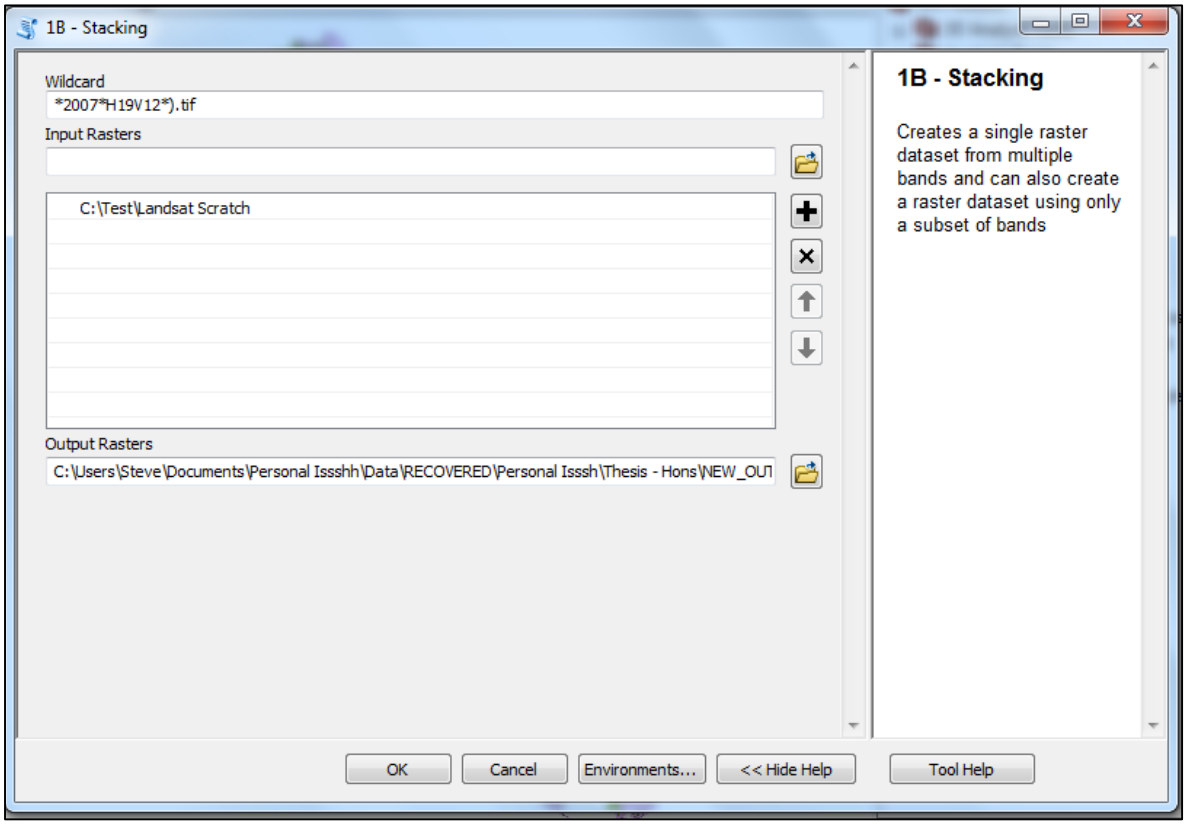


Figure 4.6 Layer stacking interface (Tool 1B)

The user can also indicate a string as mask when working with multiple files. The output file, also in GEOTIFF format, is saved in a user-specified or temporary folder. All parameters are entered via the ArcMap interface (Figure 4.6). The Python script uses ArcGIS Composite bands functionality via the ArcPy wrapper.

Having created a resampled and pre-processed image, the next step of MEAWAT is to create NDVI phenology and training data, as discussed in the next sub-section.

4.3.2 Creating training samples

The creating training data toolset in the MEAWAT toolbox contains two custom tools namely *2A- Extract multi-values to point* and *2B - Visualize phenology*. The objective of this toolset is to extract values from the stacked image dataset from the training sample points representing unique classes. These values are then used to train the DT classifier to be used in image classification. In the following paragraph creation of the toolset will be discussed.

The tool *2A* script entails creation of the training data that will be used in the classification. Using the ArcMap interface (Figure 4.7), raster values are extracted based on the point input features supplied by the user. The ArcPy geo-processing tool "Extract multi-values to points" is used in the Python script.

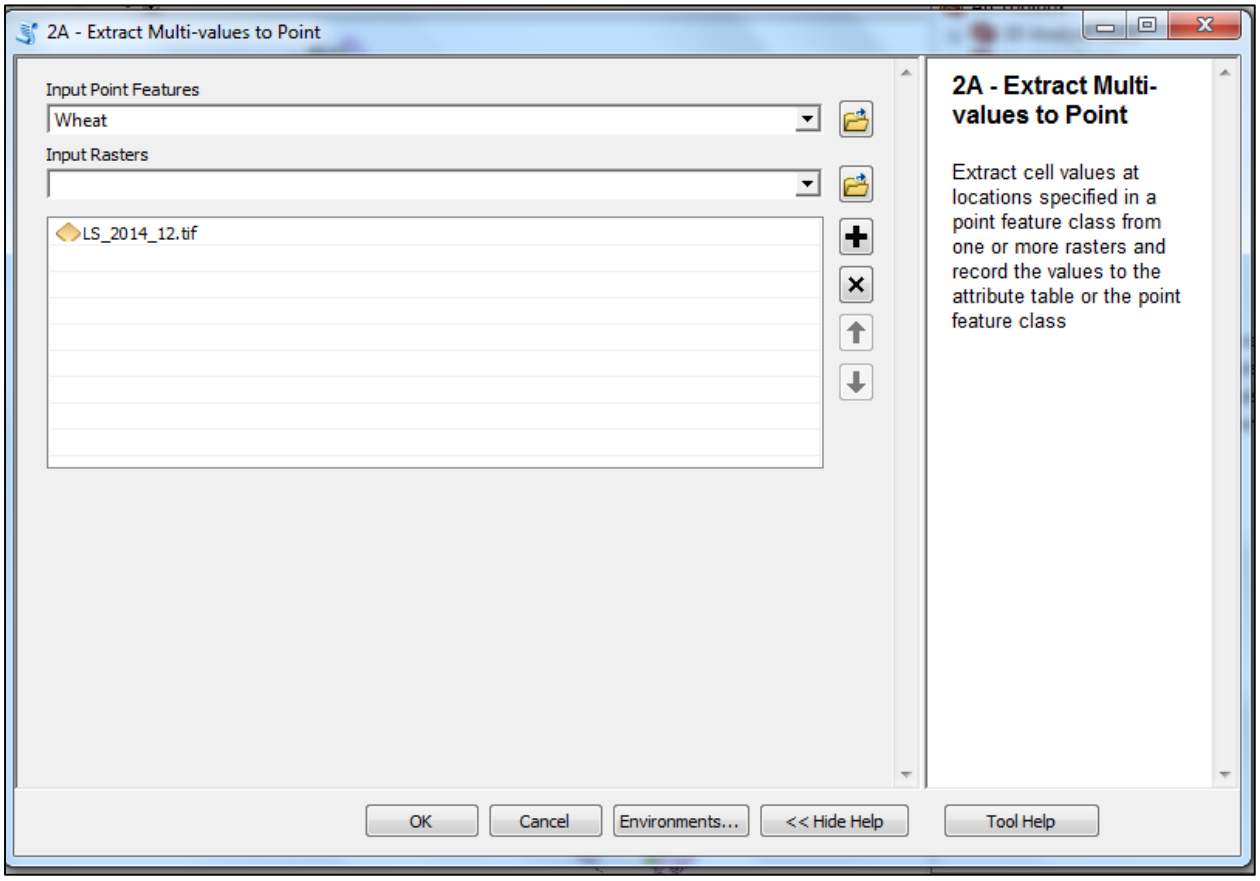


Figure 4.7 MEAWAT Extract Multi-values to point interface (Tool 2A)

The tool *2B* is used to visualize the selected training data. The output point file from tool *2A*, which should include the extracted feature values, is used to generate a graph with the day of image acquisition plotted on the X-axis against NDVI values on the Y-axis using matplotlib. Matplotlib can be used to generate complex graph within a script. Statistics are calculated for the NDVI values to show the median, 5^{th} and 95^{th} Percentile. The generated graph is in HTML format. The *2B* tool GUI is depicted in Figure 4.8.

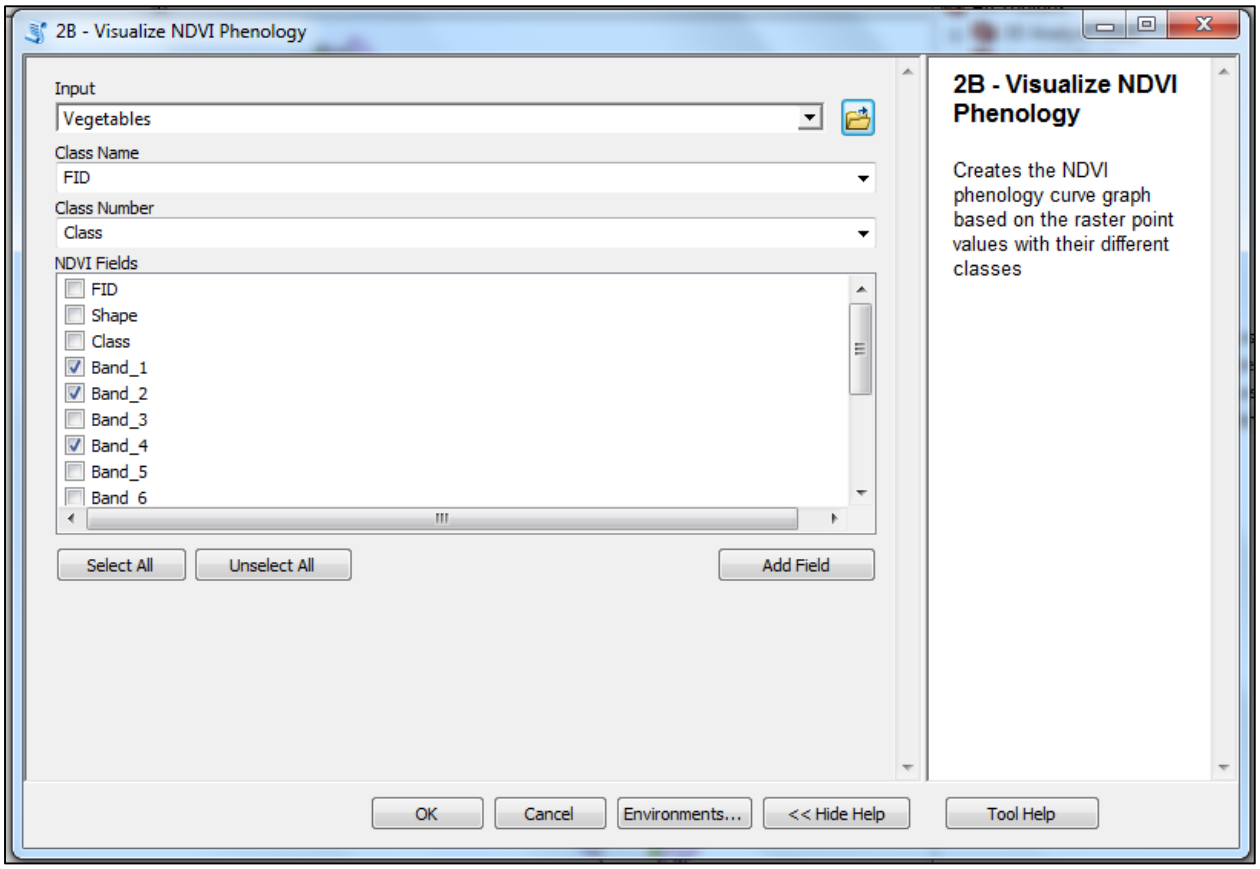


Figure 4.8 Create NDVI phenology interface from MEAWAT (Tool 2B)

The user must tick the check boxes in the NDVI Fields dialog to select the appropriate layers that will be used for building the DT. The image processing toolset is explained next.

4.3.3 Image processing

Image processing involves the manipulation of data (satellite imagery) to extract information, enhance its quality or change its format. It can involve a wide range of procedures for which specific programmes and software are needed (Chapter 2). The development of MEAWAT for image classification is described below.

First ENVI was tested and a DT was built. However, the DT classification routine in ENVI is proprietary in nature, as such no sample scripts were available to use in order to understand the mathematics behind the DT technique in ENVI. Anyone wanting to use MEAWAT would then require an ENVI license. This necessitated the search for an alternative DT classifier to be utilized in MEAWAT, therefore Scikit-learn was used.

Scikit-learn is an open source Python module integrating various machine learning algorithms. It is built on sciPy, numPy, matplotlib and it is script oriented which enables geo-processing of large volumes of data (Pedregosa et al. 2011) (For a detailed description of Scikit-learn, see Section 2.3). It is regarded as an efficient tool for data mining and data analyses such as

classification, regression, clustering, dimensionality reduction, model reduction and pre-processing. Different classifier algorithms are available within Scikit-learn, such as the DT, RF and ET, and these are all made available in the image processing toolset of tree classification in MEAWAT (Figure 4.9).

The image classification tool includes training the selected classifier, fitting the experimental model and calculating the model fit. Afterwards, the model is applied to the entire raw dataset, thus generating a land cover map for each DT algorithm used. A new tree is created for each algorithm used, and the output is saved to a folder.

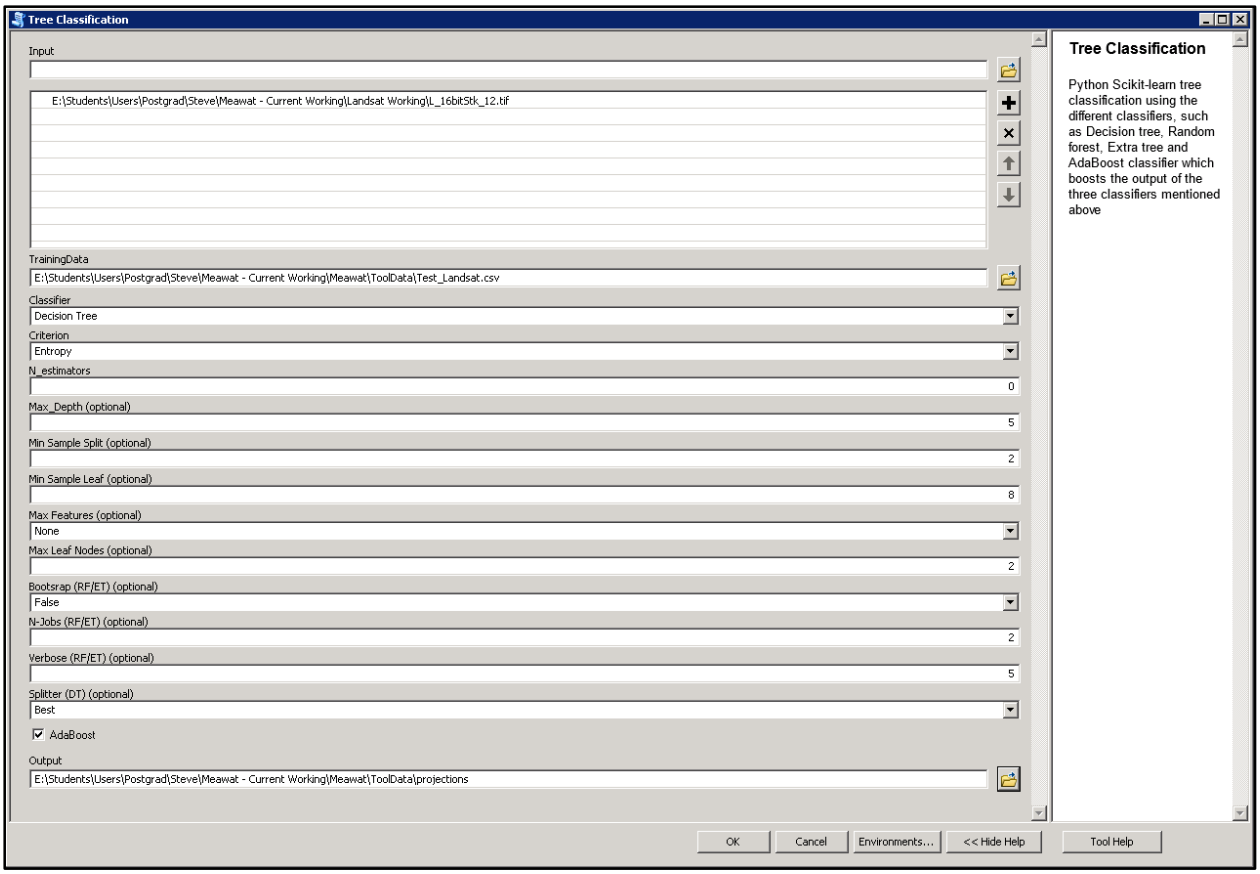


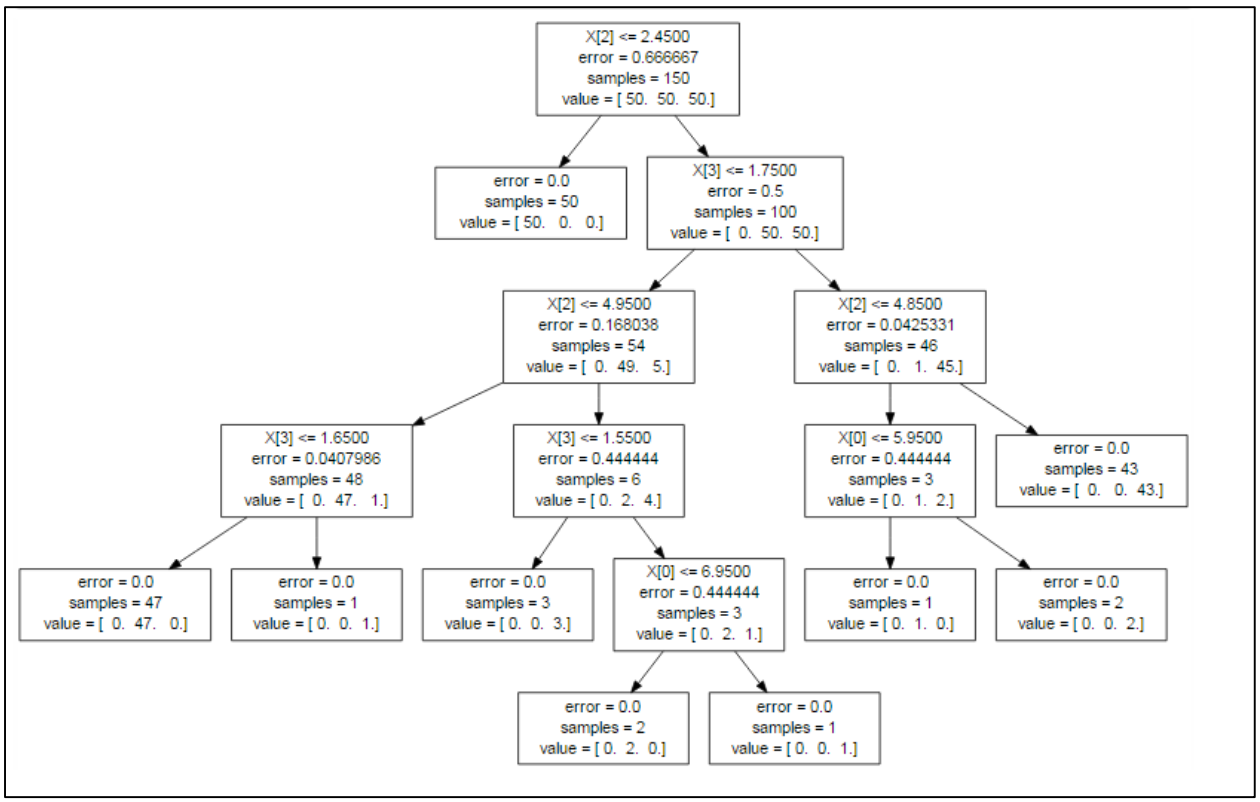
Figure 4.9 Image analysis interface

In addition to the parameters of the different classifiers, the user can manipulate the quality of the classification output. The user has the ability to select from the drop down menu which classifier to utilize, and also test any parameter until a desirable result is attained. To use the AdaBoost classifier, the user checks the AdaBoost function box to initiate it, which in turn will create an enhanced classification that generally improves upon the original classifier selected (Figure 4.9).

The user also specifies the classification output location folder. Image classification was subsequently carried out using the different algorithms supported by the Python Scikit-learn

including DT, ET, RF and AdaBoost. The techniques involved with each algorithm are explained subsequently, starting with DT.

Using the trained dataset, a tree was constructed which can be exported in Graphviz format (Figure 4.10). After the training dataset had been fitted correctly, the model was used to predict new values, thus fitting the model on the entire dataset.



Source: Pedregosa et al. (2011)

Figure 4.10 Scikit-learn decision tree graphvis format

The Scikit-learn DT classifier is based on different parameters and outputs. The parameters include *Criterion*; which is used to measure the quality of split, and can either be “gini” for the gini impurity or “entropy” for the information gain. *Splitter*; specifies the strategy to split at each node, and can either be “best” for best split or “random” for best random split. The *max features* parameter is the number of features to consider when looking for the best split, which can either be an integer, float and none assigned. The *max depth* shows how the tree fits into the data, and is used to prevent over fitting as it controls the size of the tree. The *min sample split* is the minimum number of samples needed to split an internal node. The *min sample leaf* is the required minimum number of samples at a leaf node. The *max leaf nodes* parameter grows a tree in a best-fit fashion and it is at its best when there is relative reduction in impurity. The *random state* is the seed used by the random number generator. The attributes associated with the DT

classifier also include; *tree, max features, classes, n classes and feature importance*. The extra-tree toolset is explained in the next paragraph.

The Scikit-learn ET algorithm makes use of the ensemble method, which integrates predictions of various base estimators built with a specific learning algorithm with the aim of improving robustness over a single estimator (Pedregosa et al. 2011). The extra-tree classification process starts by fitting a number of randomized extra-trees on several sub-samples of the dataset with the aim of averaging them in order to control over-fitting as well as enhance the predictive ability. The training dataset created from the NDVI phenology curve is then used to fit the model. The classifier reads the training dataset in CSV format, with the dictionary assigning numeric values to the text classes.

Upon correctly fitting the training dataset, the model is used to predict new values, thus fitting the model on the entire dataset. The Scikit-learn ET classifier uses different parameters. The parameters involved with ETs are similar to those of DTs, as such only the parameters not specified above will be further explained here to avoid repetition.

The parameters include: *n_estimators* (number of trees in the forest), *bootstraps* (indicates if bootstraps samples are used in the tree building phase), and *oob_score* (gives an option of using out-of-bag samples or not) (Hastie, Tibahirani & Friedman 2009). Also available are the *n_jobs* parameters (it specifies number of jobs to run laterally for both fit and predict), and lastly *verbose* parameter (which regulates the verbosity of tree building phase). Different parameters were explored to achieve an optimized and enhanced extra-tree classification and also to determine which method gives the best classification accuracy. The extra-tree output mean score was also determined to evaluate its performance.

In the RF classification, the process was the same as explained with extra-tree above. Upon correctly fitting the training dataset, the model was used to predict new values, by fitting the model on the entire dataset. The parameters involved with random forest are similar to those of extra-tree, as such all the parameters are already specified to avoid repetition. Different parameters were explored to achieve an optimized and enhanced random forest classification and also to determine which method gives the best classification accuracy. The random forest output mean score was also determined to evaluate its performance.

The Scikit-learn AdaBoost algorithm which is a booster widely used by researchers (see Section 2.3) makes use of the ensemble method. The AdaBoost fits an array of uncertain learners, for example small DTs, that are only slightly better than random assumptions, on repeatedly modified versions of the data (Pedregosa et al. 2011). The total predictions are then combined through a weighted sum to produce the final prediction. The AdaBoost classifier is used to boost

the three classifications mentioned above, in order to achieve an optimized and enhanced classification.

Upon correctly fitting the training dataset, the model is used to predict new values. The model used a combination of any of the three different classifier algorithm alongside with the AdaBoost model to enhance its classification. The Scikit-learn AdaBoost classifier is also based on different parameters and outputs. The parameters involved with AdaBoost are different to that of the other classifiers. The parameters are *base_estimators* (which is the base estimator from which the boosted ensemble is built i.e. the classifiers), *n_estimators* (indicates the maximum number of estimators at which boosting is stopped), and *learning rate* (shrinks the input of each classifier). Also available is the *algorithm* parameter, which could be either "samme" or "samme.R". The former uses the discrete boosting algorithm while the latter makes use of real boosting algorithm, hence supporting calculation of class probabilities. Lastly *random state*; entails the seed used by the random number generator. The AdaBoost output was also cross validated to evaluate its performance.

4.3.4 Accuracy assessment

One of the numerous advantages of automation using MEAWAT is the ability to generate accuracy assessment for the classification undertaken, thus creating an error matrix within ArcMap (Figure 4.11). Various steps (Python enabled) were followed to achieve this. The "Extract values to point" Spatial Analyst tool was integrated into a Python script to extract the raster values of the classification to points.

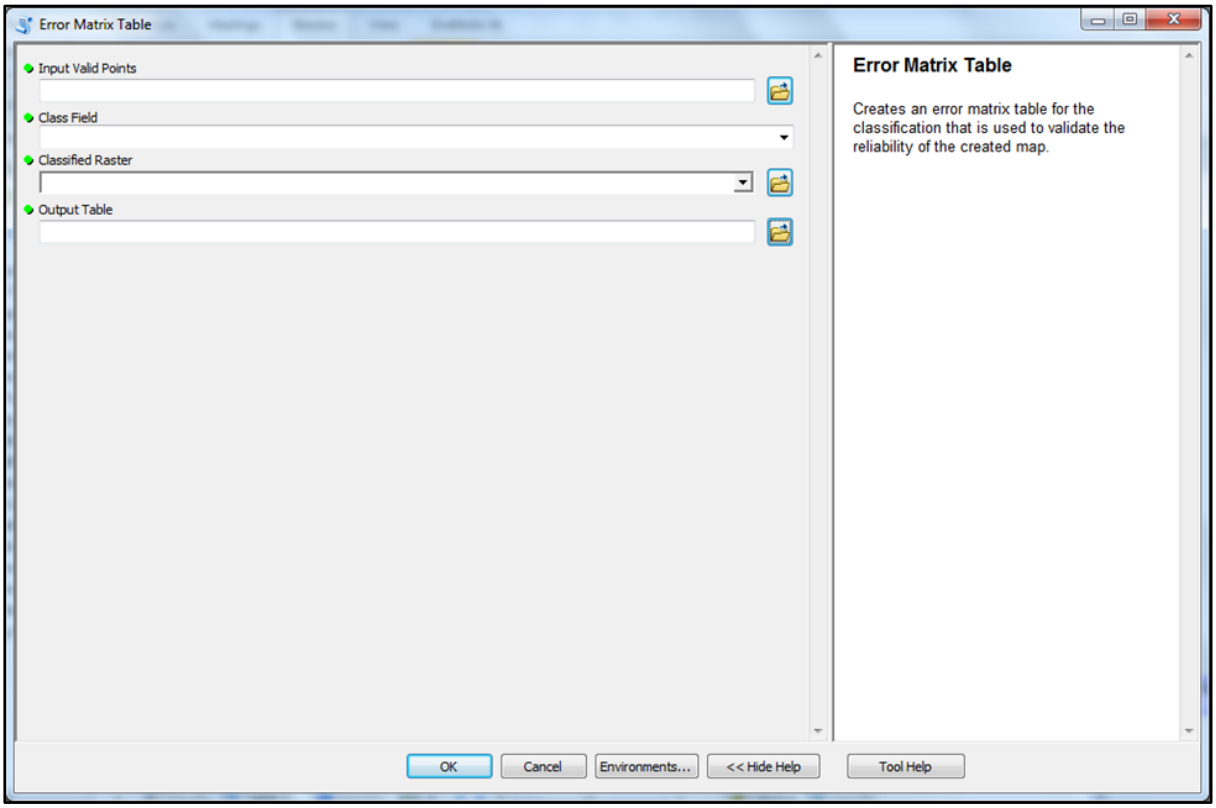


Figure 4.11 Accuracy assessment interface within MEAWAT

Negative values representing pixels with no data value and therefore no prediction were filtered out. The Frequency tool is used to create a summary table of each class to show predictions against ground truth data output to a DBF table. Summary information is rearranged into an error matrix format using the Pivot Table tool. Calculations in the Python script computes the error matrix and formats the result showing cells along the diagonal as the correct predictions against the classes as well as overall accuracy. Row and column totals, omission and commission errors, producer and consumer accuracy percentages, overall accuracy percentage and the kappa coefficient are also added to the pivot table.

4.3.5 Merge Run

The MEAWAT toolbox was subdivided into different toolsets to ensure that users can choose to run only a specific part of the toolbox that suits their purpose, instead of running all the steps. The *Merge Run* tool combines all the different scripts in the different toolsets into a single model, which will execute the toolsets sequentially to produce a classified image (Figure 4.12).

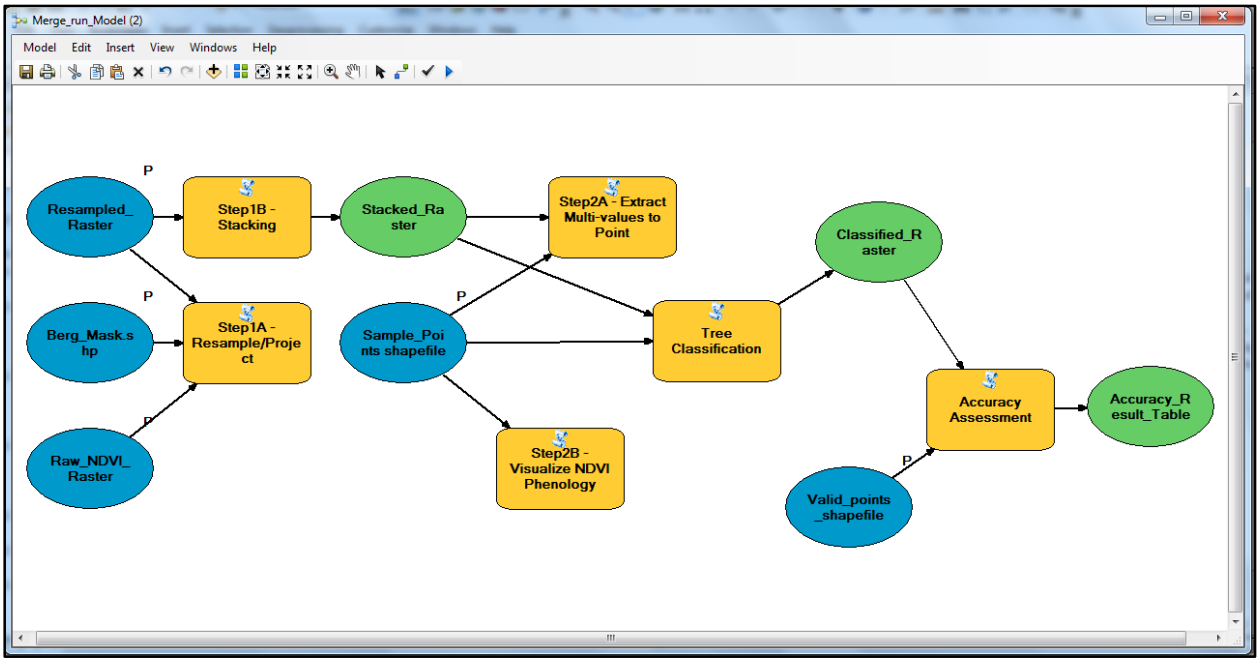


Figure 4.12 Merge Run model created to combine all the functionalities in MEAWAT

4.4 CONCLUSION

This chapter described the steps in the development of MEAWAT and its toolsets as well as the combination of these toolsets into a single model to automate classification, starting with the workflow and detailed processes. However, modifications are required in the processing chain and toolbox to ensure transferability to other imagery. The next chapter gives an in-depth demonstration of MEAWAT and example results obtained using MODIS and Landsat data.

CHAPTER 5: SYSTEM DEMONSTRATION AND EVALUATION

For MEAWAT to be demonstrated data was captured to test the tool, imagery was downloaded, pre-processed and prepared for processing. This chapter demonstrates MEAWAT used for classification with MODIS and Landsat data and validation of the tool's output by comparing it with a previous study in the same study area (Adesuyi & Münch 2015). As such, a comparison was also made between MEAWAT pre-processing output and the MODIS resampling tool (MRT) output. Furthermore, parallel investigation of ENVI software as a possible alternative classification tool is described. This chapter continues with the results obtained from MEAWAT using Python module Scikit-learn DT and ensemble classifiers as well as the accuracy assessment using both MODIS and Landsat data. Difficulties encountered during testing of transferability of MEAWAT to Landsat imagery are also described. To conclude, the transferability potential of MEAWAT is discussed.

5.1 DATA CAPTURE

Data capture is one of the most important steps in the tool development and land cover modelling process as it involves the collection and preparation of the data for image analysis. In this study, MODIS and Landsat satellite imagery datasets were used for demonstration.

5.1.1 MODIS data

MODIS 250m Terra (MOD13Q1) imagery for the period 2007 – 2014 was obtained courtesy of the NASA Earth Observing System Data and Information System (EOSDIS) LPDAAC, USGS/Earth Resources Observation and Science (EROS) Center, Sioux Falls, South Dakota (https://lpdaac.usgs.gov) using an EOSDIS user account created specifically for this project. The MODIS satellite imagery is acquired daily and composited by LPDAAC into products. NDVI and EVI were extracted as described in Sub-section 2.1.1.

5.1.2 Landsat data

Landsat 8 OLI/TIR (LC81750832014003LGN00 – 355LG00) satellite imagery data for the period January – December 2014 was acquired from United States Geological Survey's Landsat archive. The images were atmospherically corrected and NDVI values calculated from VR and NIR bands 4 and 5 respectively using the formula NDVI = (Band5 - Band4) / (Band5 + Band4). NDVI values were converted to integer values (scale factor 0.0001) to ensure compatibility with MEAWAT model and MODIS data.

5.1.3 Ancillary data

Ancillary data collected included the following:

- The Berg River catchment broad land use and Langgewens crop trial camps obtained from the Department of Agriculture Western Cape Province.
- Aerial imagery of Western Cape acquired from National Geo-spatial Information (NGI) through the Centre for Geographical Analysis (CGA) at Stellenbosch University.
- Western Cape field boundaries developed by Spatial Intelligence (SiQ), dated 2013 obtained from the Department of Agriculture Western Cape Province (used with permission).
- Two metre resolution imagery obtained through Pictometry-online to differentiate features that were indistinct on the MODIS and Landsat images.
- Field survey data for 2010 (Stuckenberg 2012) and 2013 (Adesuyi & Münch 2015) which served as reference data to guide and validate the accuracy of the classification.

5.2 RESAMPLING AND PRE-PROCESSING IN MEAWAT

Resampling and pre-processing in MEAWAT were performed using functionality similar to that of MRT. Due to differences in MEAWAT resampling versus MRT resampling some discrepancies were noted in the resulting NDVI values. Table 5.1 compares the output from MEAWAT with that of MRT. MRT which implements the GDAL resample routine resamples from the upper left corner of the raster extent to the lower right corner. In contrast, MEAWAT uses ArcGIS raster processing, which samples from lower left to upper right. Both resampling techniques use a cell size of 250m with nearest neighbour as the method. This may be an important consideration for users who want to use the output from this analysis in combination with other raster data sets. Uncertainty can be introduced in the model in this way as the raster cells may not align consistently. Table 5.1 shows the variations in the NDVI values for the same cell when using MEAWAT or MRT resampling. MEAWAT not only implements the basic functionality available within MRT for MODIS satellite imagery but allows elegant incorporation of different sensor data for further analyses.

Table 5.1 Comparison of MODIS NDVI values from resampling using MEAWAT versus MRT

MEAWAT	MRT	Absolute Difference
3289	3425	136
3010	2546	464
3915	3915	0
3454	3894	440
4130	4446	316
5220	5220	0
1959	2605	646
1955	1955	0
2483	2483	0
2724	2587	137
1911	2341	430
2666	2305	361
2044	1990	54
3250	3250	0
2174	2174	0
3895	3786	199

5.3 DECISION TREE CLASSIFICATION USING ENVI SOFTWARE

Based on earlier work done in the study area using a WEKA DT to perform land cover classification (Adesuyi & Münch 2015) ENVI (Exelis 2013a) was investigated as a possible candidate to implement into MEAWAT (see Sub-section 3.1.2).

The DT's generated by ENVI and WEKA are shown in Figure 5.1 and Figure 5.2 respectively. In the DT, the root node is represented by day of acquisition of the NDVI image, for example D001 means the NDVI image was acquired on first day of the year and D017 on 17th day of the year which are further split into nodes based on a yes and no question to further assign them into classes, and the names are automatically assigned in the software depending on column names specified by the user. For example if the NDVI value for the day of acquisition falls within the *wheat* NDVI value range (Yes), the class *wheat* is assigned to the band and if (No) it is first check to see if it falls into the other two classes NDVI range and if not it is further split until it falls within a class range. Both DT's (Figure 5.1 and 5.2) are machine generated. Although the structure of the ENVI DT (Figure 5.1) compares well with that of WEKA (Figure 5.2), the accuracy assessment results from the two classifications were very different.

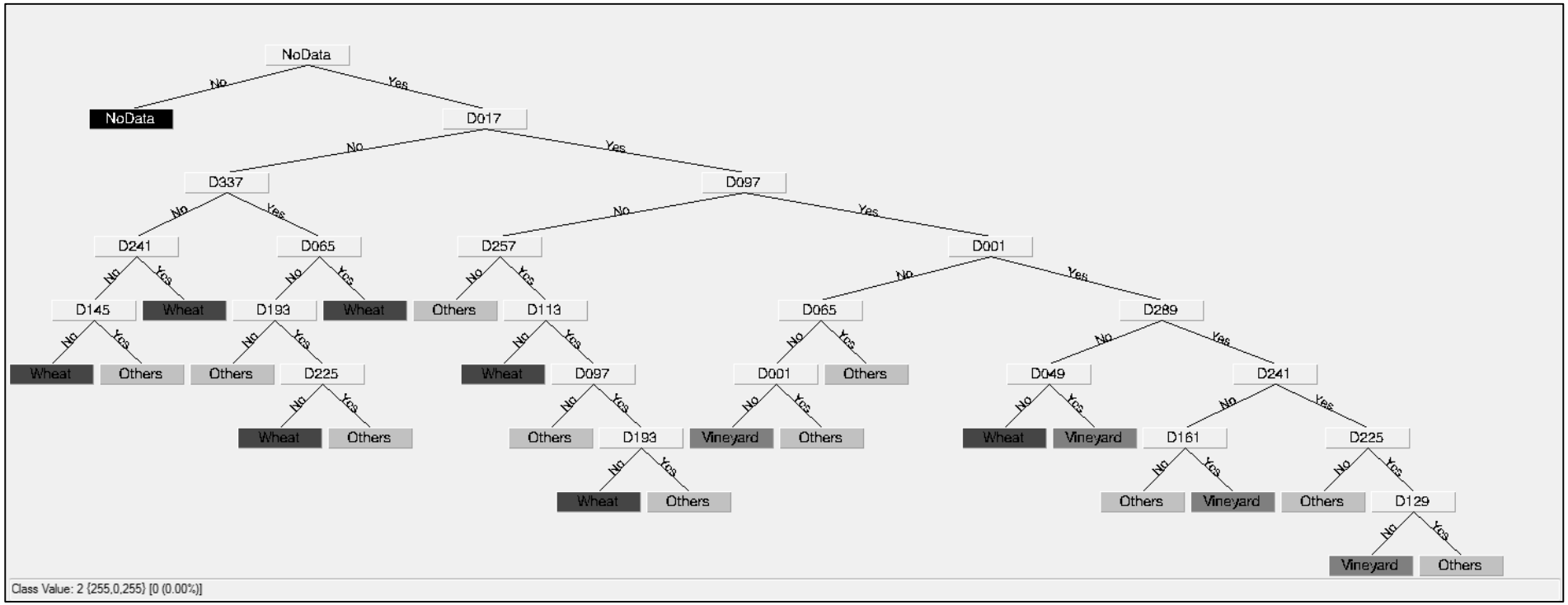


Figure 5.1 ENVI decision tree

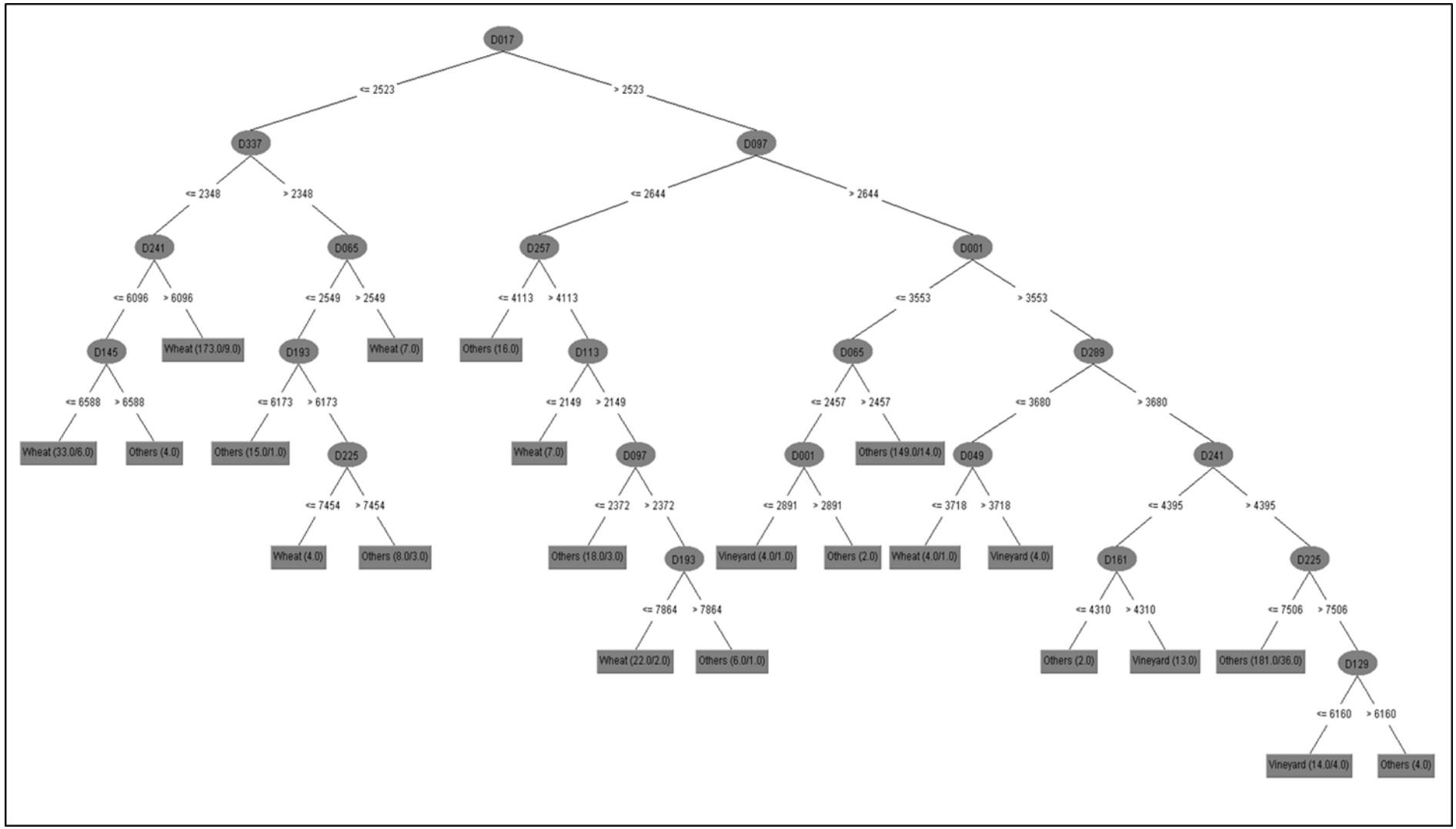


Figure 5.2 WEKA decision tree

Figure 5.3 is a comparison of the two classifications showing the error of commission and error of omission for the three classes, *wheat*, *vineyard* and *others*. The maps (Figure 5.4) are the two classifications maps generated in (A) WEKA DT and (B) ENVI DT. Although they appear similar, the accuracy assessment carried out on a hundred random points shows an accuracy of 80% (kappa 0.6) for WEKA compared to ENVI with an accuracy of 48% (kappa 0.03).

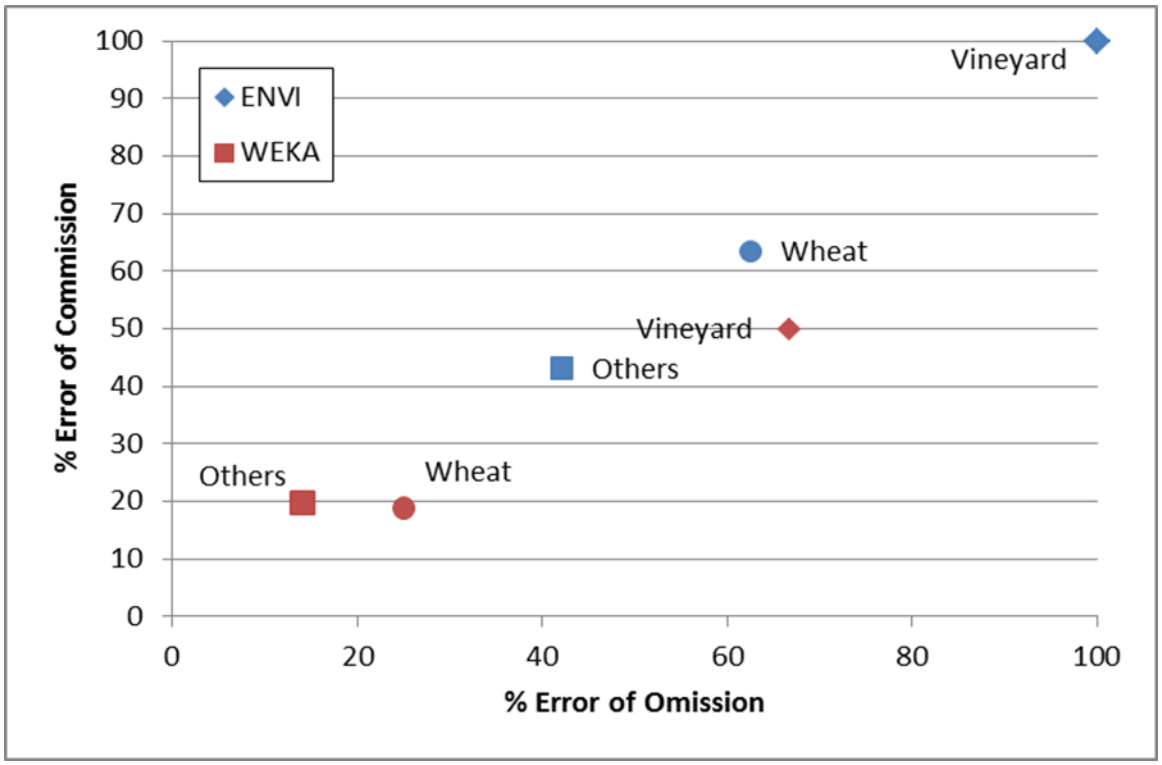


Figure 5.3 Errors of omission and commission for land cover classification using WEKA DT and ENVI DT

The *wheat* and *others* classes could be classified with greater success using either of the DTs, while *vineyard*, which has a much smaller presence in the study area, shows a particularly poor classification accuracy using the ENVI DT (100% error), with the WEKA DT not much better with 50% error. The shortcoming of this accuracy assessment is the small number of reference samples available, necessitating the use of the SiQ data set as a better source for training and validation data. Vineyards often have winter cover crops (oats, triticale and sometimes grass) which could confuse the classification.

This comparison was carried out to have a baseline of accuracy. Since better training data became available (SiQ), the DT from WEKA was recreated in ENVI. It was apparent that good training data has an influence on the classification.

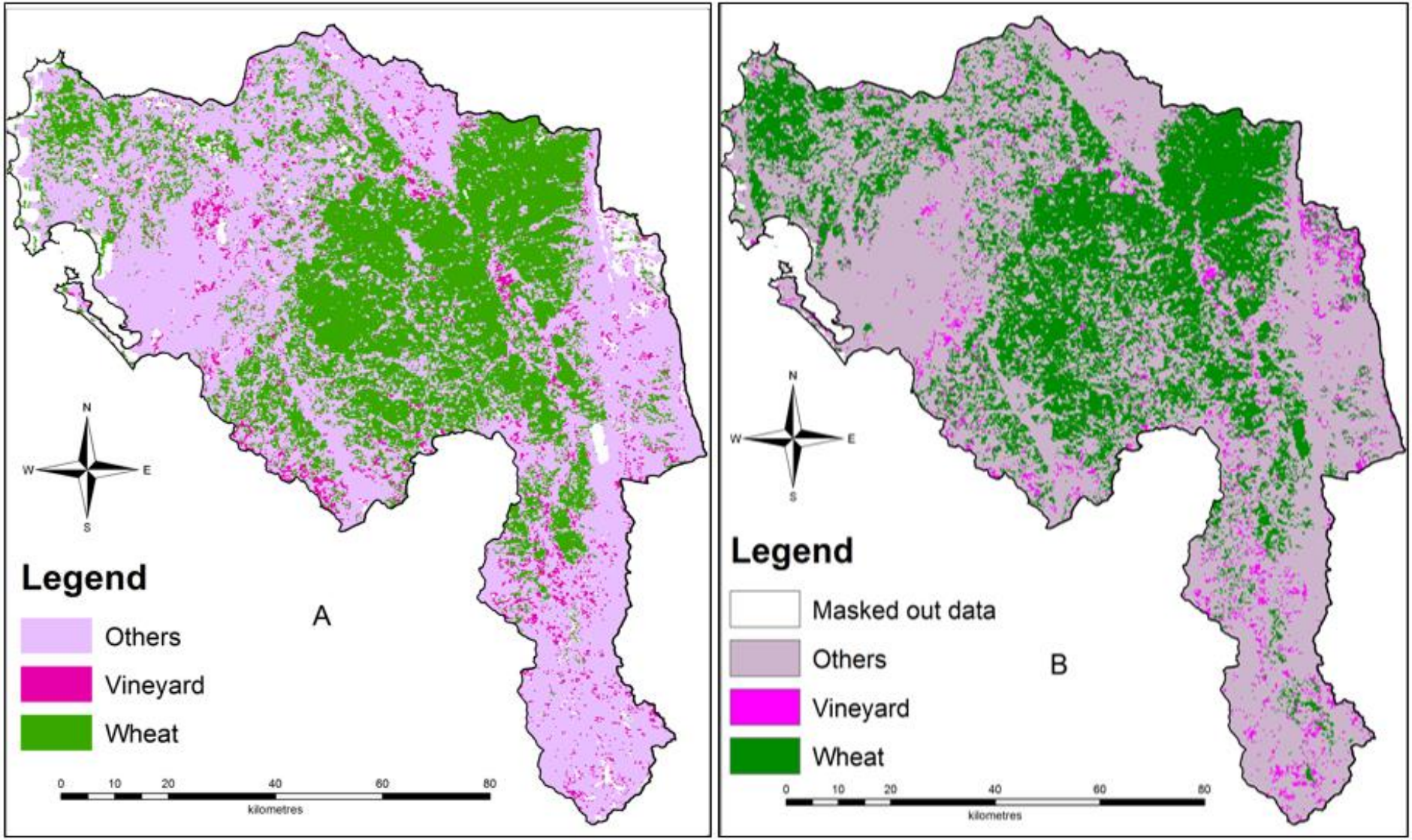


Figure 5.4 Comparison of land cover map generated in WEKA (A) against its replica recreated in ENVI (B)

According to Otukei & Blaschke (2010) the challenge in using DTs lies in determining the best tree structure to delineate the decision boundaries. The SiQ data represented a more comprehensive, accurate census of the agriculture within the study area and was therefore used to generate a new training set. This training data was used by the selected classifier to build the appropriate DT.

5.4 TRAINING DATA, DT AND CLASSIFICATION

To identify sample dominated by wheat, vineyard and other agricultural classes in the catchment, ancillary data from Langgewens agricultural research station as well as the Western Cape field boundaries, and SiQ data were used. Field sample data, with different classes identified from field photographs and pictometry, were used to create a NDVI phenology curve from the MODIS NDVI imagery representing time-series data. Using selected points and MEAWAT, the MODIS NDVI data for each of the 23 MODIS images stacked per year for period of 2007 to 2014 were extracted. Poor quality data were removed which included data representing cloud cover and water with negative NDVI. The median value per class for each year was calculated for the *wheat*, *vineyard* and *others* classes. The lower and upper percentiles of VI values for each year was empirically determined to identify specific VI tail thresholds (Brown et al. 2013).

This ENVI DT classification represented an improvement over the DT classification described in Section 5.3. Vineyard was better represented (Figure 5.5), which can be attributed to better representation of the vineyard class in the new training data, while wheat remained unchanged.

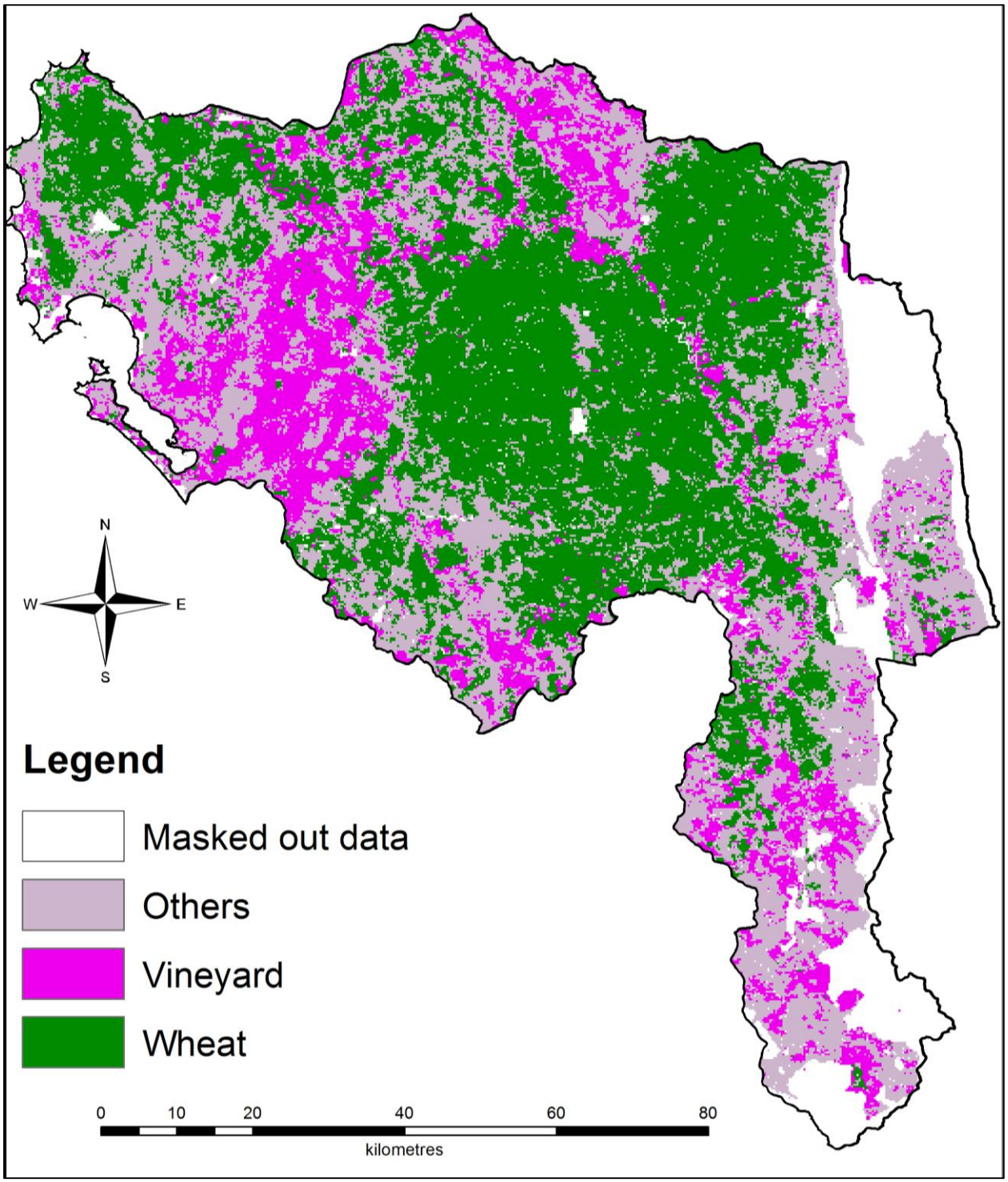


Figure 5.5 ENVI decision tree classification

The results from the ENVI accuracy assessment carried out on 16000 random points (Table 5.2) have an accuracy of 71% with a kappa of 0.5. The vineyard fields occupy a small area, generally less than 6.25 ha, therefore the MODIS resolution might not be able to pick this class up successfully. This explains the lower accuracy observed in the *vineyard* class.

Table 5.2 Confusion matrix for ENVI decision tree classification result

Classification Result									
Reference Data		Others	Vineyard	Wheat	**Total**	% Error of Omission	% Error of Commission	Producer Acc %	User Acc %
	Others	2206	325	1605	**4136**	46.66	26.17	53.34	73.83
	Vineyard	235	541	39	**815**	33.62	44.68	66.38	55.32
	Wheat	547	112	4229	**4844**	13.60	28.20	86.40	71.80
	Total	**2988**	**978**	**5829**	**9795**				
	Overall Accuracy		71%		Kappa		0.5		

With this improved accuracy the ENVI DT appears to be the perfect candidate to be used for MEAWAT. However, the DT classification routine in ENVI is proprietary in nature and anyone wanting to use MEAWAT would then require an ENVI license. This confirmed the requirement for an alternative DT classifier to be utilized in MEAWAT.

The extracted NDVI values for the selected points representing known classes were plotted on a graph using MS Excel to identify the NDVI phenology curve for classes *wheat*, *vineyard* and *others* (Figure 5.6) with NDVI values (scale factor 0.0001 for MODIS data) plotted on the vertical axis against the day of acquisition (DOA) of the imagery on the horizontal axis.

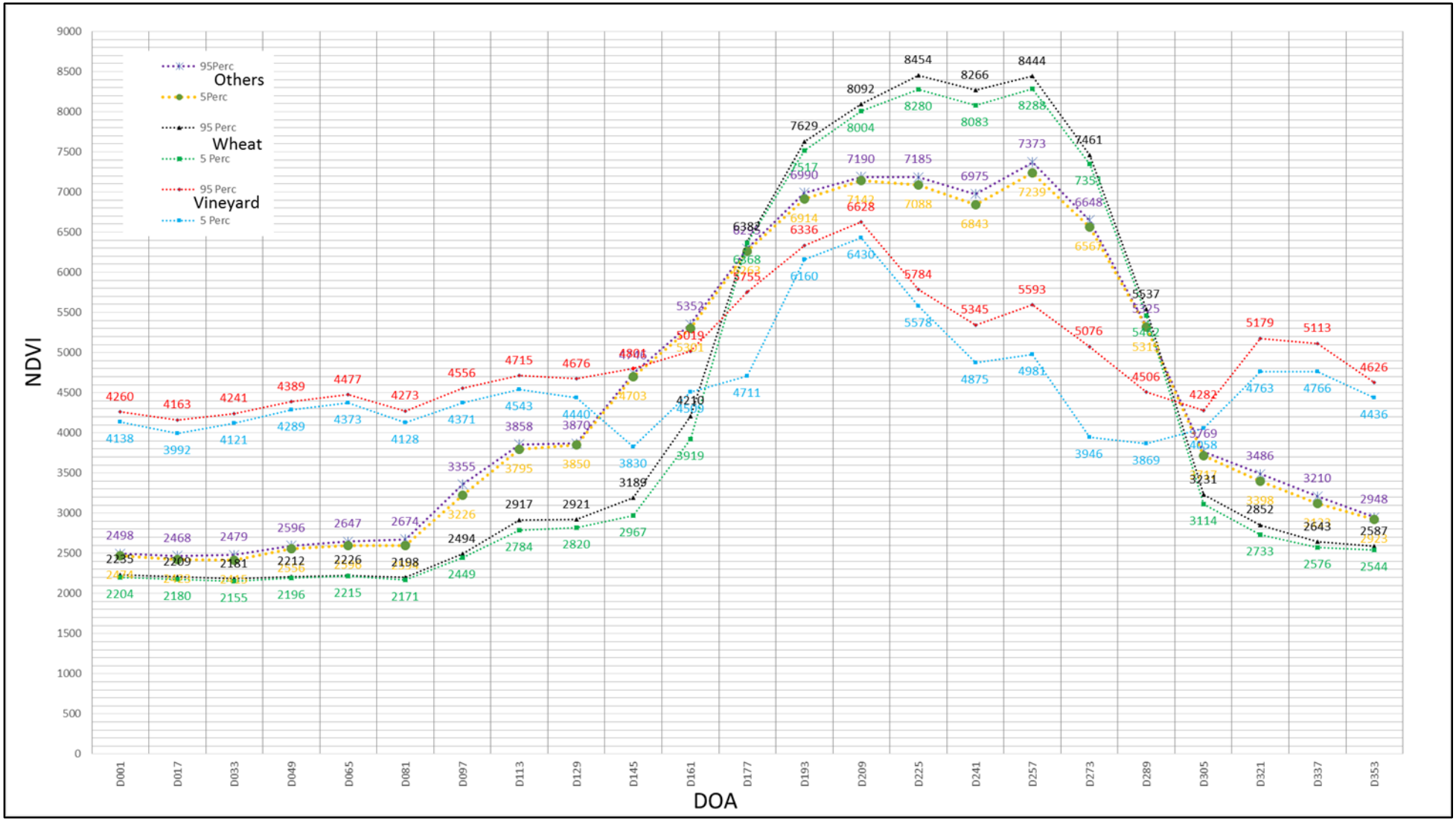


Figure 5.6 NDVI phenology curve for classes *others*, *wheat* and *vineyard*

The different software used had an influence on the output of the classification. WEKA makes use of C.45 software and the J48 DT algorithm which is believed to produce better results than the DT classification in ENVI (Jaloree, Rajput & Gour 2014). The high accuracy (71% kappa 0.5) result obtained from using the SiQ data also outlines the importance of choosing good training data. The open source Scikit-learn module coded in Python was selected as it implements the correct DT algorithms for this purpose and the outcome of this classification using MEAWAT will be presented in the following section.

5.5 PYTHON SCIKIT-LEARN TREE CLASSIFIATION RESULT

The image analysis performed in MEAWAT tests the application of different algorithms available for tree classification using Python Scikit-learn (Chapter 4). The different parameters available for the different classifiers can influence results during land cover classification. The results from the different algorithms (DT, RF, ET and Adaboost) are presented in the sub-sections below.

5.5.1 Decision tree classification

The Scikit-learn DT classification process starts with the DT classifier taking an array of X size (n-samples, n-features) which holds the training samples and an array Y of integer values, which holds the class labels for the selected classes. The classes extracted using NDVI values from the phenology curve (demonstrated in Figure 5.6) was used to fit the model. The classifier reads the training dataset in CSV format, with the dictionary assigning numeric values to the class. Subsequently each of the bands in the dataset (date of acquisition) are assigned to a row and class based on the values given by the dictionary.

Different parameters were explored to achieve an optimized and enhanced DT classification (Table 5.3) for the selected classes. MEAWAT keeps a time record for each analysis as well as cross validation information which evaluates the estimator performance. The findings emerging from the validation of the resulting model on the remaining data are regarded as the mean score with a 95% confidence interval of the score estimate of the values computed in the loop. This was done to see if the mean score could be used as a predictor for a better classification.

The Scikit-learn DT classifications resulted in a lower accuracy compared to the other classifiers (RF, ET and Adaboost) used. Different parameters were tested (Table 5.3) and it was observed with DT5A and DT5B parameters that the *criterion* parameter “entropy” produced a better result than “gini” according to the accuracies derived. Although the wheat class was better represented in “gini” than in “entropy”, both classifications did not give an accurate representation of the

vineyard and *others* classes. The *vineyard* class seemed to dominate in the same extent as the *others* class. This is incorrect considering the ratio of vineyard to other crops present in the study area, hence the low accuracy derived.

Table 5.3 Python Scikit-learn DT parameter combinations and associated accuracies

Raster	**Criterion**	**Max Depth**	**Min Split**	**Min Leaf**	**Splitter**	**% Accuracy**	**Kappa**
DT5A	Entropy	5	3	2	Best	68	0.4582
DT5B	Gini	5	3	2	Best	60.1	0.3165
DT9A	Entropy	12	2	8	Best	68.7	0.4640
DT9B	Gini	12	2	8	Best	70.4	0.4999
DT10A	Gini	12	5	8	Best	70.4	0.4999
DT10B	Entropy	12	5	8	Best	68.8	0.4661
DT12A	Gini	20	5	8	Best	70.5	0.4999
DT12B	Entropy	20	5	8	Best	68.8	0.4661
DTS1	Gini	15	3	8	Best	70.5	0.4999
DTS2	Entropy	15	3	8	Best	68.8	0.4661

The classification DT9A and DT9B output is different as the *others* class dominated the classification over *vineyard* and *wheat* classes for both *criterion*. It was noted that low class representation occurred when using *splitter* parameter “random” rather than using “best”, which is very similar to the output of DT10A and DT10B. From testing the *max depth* parameter in DT12A and DT12B, it was observed that a higher max depth produces a better output with “entropy” than “gini”, as the output of DT12B has a similar resemblance to the WEKA classification visually. However, the accuracy derived from both *criterion* proved otherwise. Lastly the DTS1 and DTS2 classification, does not give a proper representation of the three classes for the *criterions* (gini and entropy). Qualitatively, it was observed that “entropy” gave a better representation than “gini” for DT classification, but has lower accuracies observed in the raster examples tested in the Table 5.3, which presupposes that visual representation cannot be used to validate a good map or classification output. According to the tree (Figure 5.8), the

classifier splits the training sample data based on X (NDVI values) associated with date of image acquisition using *criterion* parameter “gini” or “entropy” using a specific number of samples by searching for a split in each node, then assigns a class to the child node.

5.5.2 Random forest classification

The RF classification output was better than the DT classification based on the accuracies derived from the parameters used (Table 5.4). Although the results of some of the parameters tested (RF3B and RF5A) had accuracies smaller than in the DT iterations, RF classification outputs were overall better. Classification from RF1A had better class separation for *wheat* and *others* class than RF1B, while the *vineyard* class dominated RF1B. Classification outputs for RF2A and RF2B had low class representation and mean score, with the outputs dominated by *wheat* and *vineyard*, thus undermining *wheat* as the dominant crop in the catchment. There was good class divisibility for RF3A that was presumed to have a good visual representation of the classification. It showed lesser *vineyard* and right proportions of *wheat* and *others* classes. Whereas the RF3B classification output was characterized with poor class representation, as only *vineyard* and *wheat* were prominent, hence the reason for the lower accuracy.

Table 5.4 Python Scikit-learn RF parameter combinations and associated accuracies

Raster	Criterion	N_Estmators	Max_Leaf Node	N-Jobs	Verbose	% Accuracy	Kappa
RF1A	Gini	10	None	1	0	69.7	0.4873
RF1B	Entropy	10	None	1	0	71.8	0.5251
RF2A	Gini	20	5	2	2	73.9	0.5637
RF2B	Entropy	20	5	2	2	75.5	0.5888
RF3A	Gini	10	2	1	2	73.1	0.5267
RF3B	Entropy	10	2	1	2	59.5	0.2957
RF4A	Gini	10	None	1	0	73.2	0.5445
RF4B	Entropy	10	None	1	0	73.2	0.5489
RF5A	Gini	10	2	1	5	55.6	0.2172
RF5B	Entropy	10	2	1	5	74.3	0.5707

The maximum leaf node values assigned in the parameters influenced the classification outcomes where either very low or very high values had negative effects on the outputs and the results. The

RF4A and RF4B classifications are similar to RF3A and RF3B, having poor and fair class representation respectively. With RF4B, the *others* and *vineyard* classes were the most prominent. The RF5A and RF5B classifications were different as they showed the influence of the *criterion* parameter on the output.

However, though similar parameters were used changing only the *criterion* parameter, to "gini" resulted in RF5A classification showing *wheat* dominance and slight traces of *others* class. The RF5A output was visually poor and low in accuracy compared to RF5B run with *criterion* parameter "entropy" and showed good class representation with better accuracy. Generally, it was observed that the parameter *verbose* had an influence on the classes while using the random forest classifier.

For enhanced classification results, it is however recommended to maintain the *max leaf nodes* and *verbose* parameters at two. The visual tree for the RF classification (Appendix B) similar to DT method (Figure B1 - 10) was split into 10 small trees based on the number of estimators for the classification, in contrast to DT with a single visual tree due to non-usage of the estimator parameter.

5.5.3 Extra-tree classification

The results of the ET classification had better accuracies in comparison to DT and RF. Using default settings, ET1A and ET1B (Table 5.5) had good representations for all classes, with only slight differences in the *criterion* outputs. Notably, class divisibility was observed between ET2A and ET2B classifications, with the ET2A "gini" showing presence of *wheat* in the south of the study area which should be dominated by *vineyard* and *others* classes. There was a decrease in the *wheat* representation in ET2B "entropy" compared to ET2A, although the *vineyard* class remained unchanged in both *criterions*. ET2B showed higher accuracy of 76%, kappa 0.6.

Despite lower accuracies than ET2B, the ET3A and ET3B classification outputs had good class separability with some anomalies in the *vineyard* class of ET3A. ET3B output with the lowest accuracy showed fair representation of the training class. The ET3B classification was similar to WEKA classification, even though the mean score was relatively low compared to mean scores of other outputs within the classifiers. The ET4 and ET5 classification outputs were similar, though still lower than ET2B, providing true representations of classes, with high mean scores and accuracies. *Vineyard* class remained relatively unchanged but obvious increase in the *wheat* class was observed on ET5 than ET4 for both *criterions*. Moreover, to obtain better classification results, the *max leaf nodes* and *verbose* parameters must be kept at two.

Table 5.5 Python Scikit-learn ET parameter combinations and associated accuracies

Raster	Criterion	N_Estmators	Max_Leaf Node	Min Sample leaf	Verbose	% Accuracy	Kappa
ET1A	Gini	10	None	1	0	71.7	0.5276
ET1B	Entropy	10	None	1	0	73.6	0.5553
ET2A	Gini	10	5	8	2	72	0.5290
ET2B	Entropy	10	5	8	2	76.3	0.5996
ET3A	Gini	15	2	8	2	74.2	0.5540
ET3B	Entropy	15	2	8	2	68.6	0.4465
ET4A	Gini	10	None	5	2	74.9	0.5781
ET4B	Entropy	10	None	5	2	74.8	0.5745
ET5A	Gini	10	None	3	0	72.5	0.5347
ET5B	Entropy	10	None	3	0	74.2	0.5654

5.5.4 AdaBoost tree classification for DT, RF and ET results

With AdaBoost classification, the same training dataset with classes created from the NDVI phenology curve, was also used to fit the model. As a meta-estimator, AdaBoost then improves the classification by fitting several weaker models to produce a powerful ensemble estimator. The AdaBoost classifier was therefore applied to all the different parameters explored using DT, RF and ET classifiers to achieve an optimized and enhanced AdaBoost classification and also to determine which method gives the best classification accuracy.

The AdaBoost tree classification is designed to improve the DT, RF and ET results. It can be confidently stated that the AdaBoost classifier achieved this expectation. Besides a better visual map, the accuracy of the classifications was also higher. A drastic change was observed in the DT classification with an improvement from its lowest accuracy of 60% with a kappa of 0.3 in DT5B (Table 5.3) to a 72.9% accuracy with a kappa of 0.5 (Table 5.6). This improvement was also noticed with RF classification from its lowest accuracy of 55.6% with a kappa of 0.2 in RF5A (Table 5.4) to an AdaBoost RF classification accuracy of 78.3% with a kappa of 0.63 (Table 5.6). In addition, an improvement in accuracy for the ET classification was also observed from its lowest accuracy of 68.6% with a kappa of 0.4 in ET3B (Table 5.5) to an AdaBoost ET classification having an accuracy of 78.6% with a kappa of 0.6 (Table 5.6).

Table 5.6 Python Scikit-learn AdaBoost improved accuracies on DT, RF and ET classifiers

DT	% Ada_DT Acc	Ada_DT Kappa	RF	% Ada_EF Acc	Ada_EF Kappa	ET	% Ada_ET Acc	Ada_ET Kappa
DT5A	71.4	0.5123	RF1A	75.6	0.59332	ET1A	70.8	0.51352
DT5B	72.9	0.54425	RF1B	74.7	0.57427	ET1B	72.2	0.53617
DT9A	72.9	0.54425	RF2A	75.4	0.58426	ET2A	75.3	0.58187
DT9B	72.7	0.54119	RF2B	77.8	0.62339	ET2B	76.6	0.60266
DT10A	73	0.54352	RF3A	77.8	0.62009	ET3A	78.7	0.63714
DT10B	73.2	0.54411	RF3B	78.6	0.63391	ET3B	78.6	0.6356
DT12A	73.3	0.55428	RF4A	65.8	0.43081	ET4A	68.2	0.4715
DT12B	74	0.56355	RF4B	71.1	0.5165	ET4B	71.2	0.51811
DTS1	73.3	0.55242	RF5A	78.3	0.62881	ET5A	68.6	0.5027
DTS2	73.1	0.5405	RF5B	78.7	0.63476	ET5B	62.9	0.42243

All the parameters tested had a higher accuracy when AdaBoost DT classifier was used, with an improved class representation. Although slight changes were observed between the *wheat* and *others* class, the *vineyard* class remained almost unchanged.

As with DT, both the AdaBoost RF classifier and the AdaBoost ET classifier produced better results. However, there was a low representation of the *wheat* class in AdaBoost RF4A and RF4B when compared to the other parameters tested for the algorithm. The best output result was achieved with AdaBoost RF5 with accuracies higher than 78%. In this case, the *criterions* "entropy" performed marginally better (78.7%) than the "gini" (78.3%). Also a high accuracy was obtained with the AdaBoost ET classifier ET3A (78.7%, kappa 0.64) using the "gini" criterion.

After testing the different combinations of parameters with each algorithm, it was concluded that it was best to use either the AdaBoost RF algorithm RF5B or AdaBoost ET algorithm ET3A parameters as they both achieved the highest accuracies (78% kappa 0.6). Consequently, for the purpose of this study, the AdaBoost RF5B parameter combination was used for the classification of the time-series. A suitable land cover map was created for this result. Land cover maps were generated for each year in the MODIS time-series (shown in Appendix B).

The mean score generated by the classifier based on training data only did not correlate with the classification accuracy, as some of the classification had lower mean scores but higher accuracies, especially noted for RF5 (Mean score 0.75 with accuracy of 74.3%) (Table 5.7).

Table 5.7 Python Scikit-learn mean score versus accuracies and processing time for AdaBoost DT, RF and ET

Raster	Mean Score	% Accuracy	Time (m)
DT5A	0.85	71.4	0.55
DT5B	0.83	72.9	0.49
DT9A	0.84	72.9	0.5
DT9B	0.85	72.7	0.48
DT10A	0.86	73	0.48
DT10B	0.85	73.2	0.51
DT12A	0.85	73.3	0.48
DT12B	0.85	74	0.51
DTS1	0.84	73.3	0.51
DTS2	0.83	73.1	0.48
RF1A	0.8	75.6	2.8
RF1B	0.8	74.7	3.5
RF2A	0.79	75.4	8.2
RF2B	0.78	77.8	9.2
RF3A	0.76	77.8	3.2
RF3B	0.75	78.6	3.2
RF4A	0.81	65.8	3.2
RF4B	0.81	71.1	3.2
RF5A	0.75	78.3	3.1
RF5B	0.75	78.7	3.1
ET1A	0.83	70.8	0.34
ET1B	0.79	72.2	0.21
ET2A	0.78	75.3	2.73
ET2B	0.77	76.6	3.35
ET3A	0.76	78.7	3.82
ET3B	0.76	78.6	3.83
ET4A	0.78	68.2	2.74
ET4B	0.82	71.2	2.77
ET5A	0.82	68.6	2.65
ET5B	0.82	62.9	2.65

The time taken to complete a classification was noted for all the classifiers. In all cases this was less than one minute, but with the AdaBoost classifier the longest time taken was nine minutes (this was observed with increased n_estimator parameters) (Table 5.7). For enhanced classification results, it is recommended to make use of the AdaBoost default parameters as changing the parameters has minor or no influence on the classification output and results. The

classification accuracy is therefore mainly determined by the quality of the DT, RF or ET classifier used in combination with AdaBoost. The land cover map for 2013 illustrates the classification output (Figure 5.7). The DT generated by MEAWAT for DT12A classification is shown in Figure 5.8.

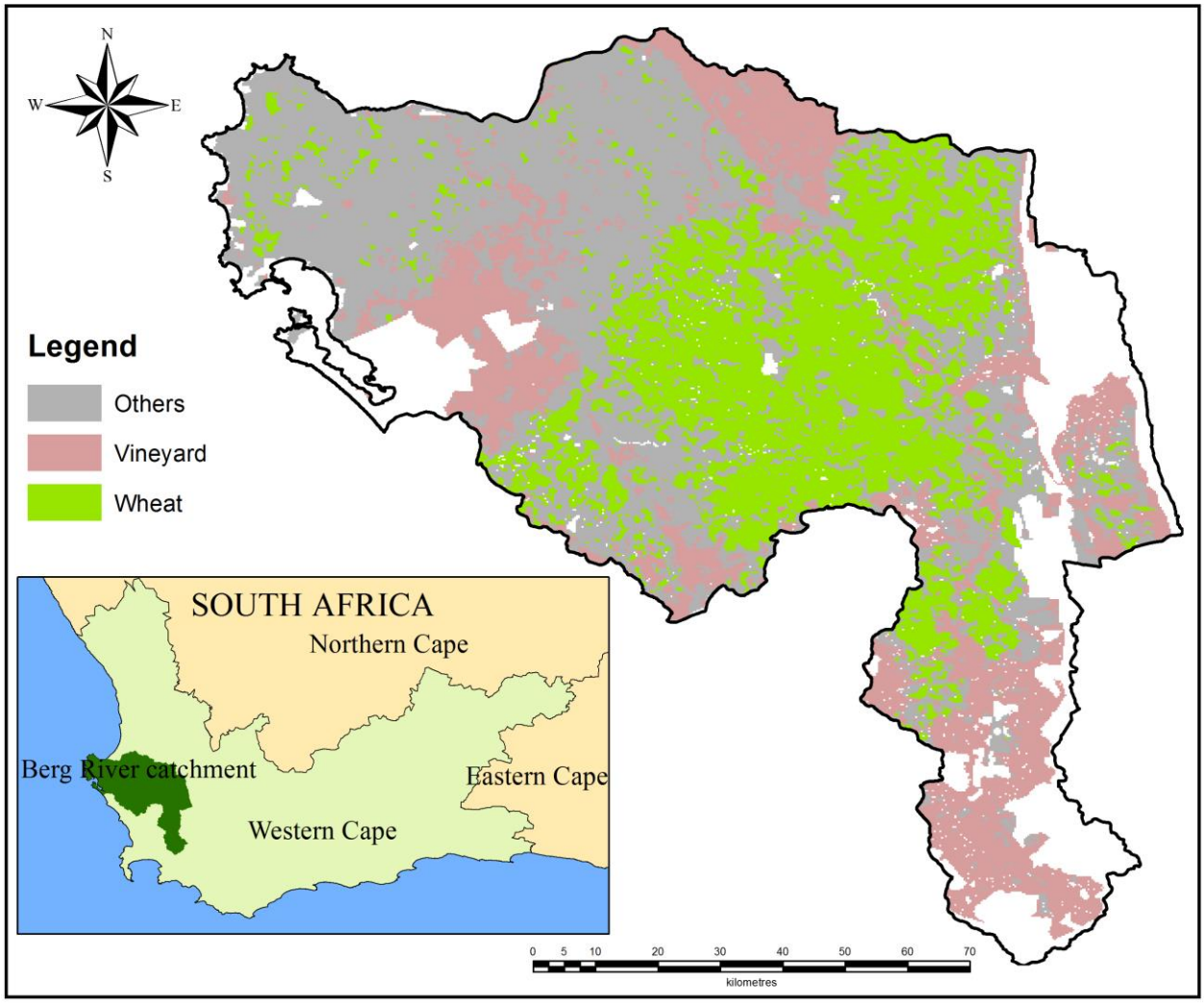


Figure 5.7 MODIS agricultural land cover from MEAWAT classification for 2013

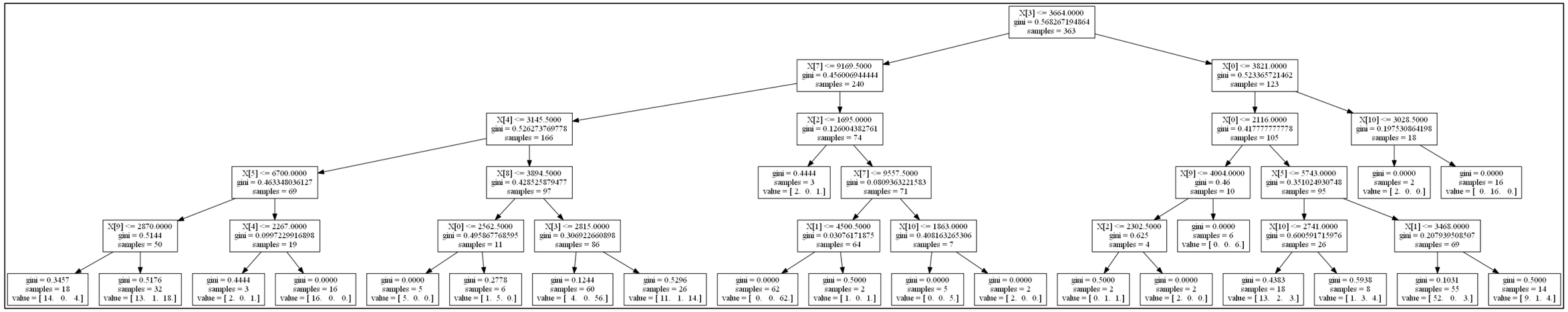


Figure 5.8 Python Scikit-learn visual tree for decision tree (DT12A) classification

5.6 ACCURACY ASSESSMENT

For the MODIS data analysis, 16000 random points were selected from pixels representing homogenous classes as described in Section 4.2. This was done to ensure that each point represented an agricultural class matching the SiQ data, i.e. the ground truth location. The land cover maps created using MEAWAT for the MODIS and Landsat data were validated using MEAWAT. The confusion matrix as generated from MEAWAT for MODIS classification AdaBoost RF5B is shown in Table 5.8.

Table 5.8 Confusion matrix for MODIS classification result using AdaBoost RF5B

		Reference Data							
Classification Result		Others	Vineyard	Wheat	**Total**	% Error of Omission	% Error of Commission	Producer Acc %	User Acc %
	Others	3259	195	952	**4406**	11.47	8.64	73.97	79.04
	Vineyard	402	599	50	**1051**	4.52	2.13	56.99	73.77
	Wheat	462	18	3840	**4320**	4.80	10.02	88.89	79.31
	Total	**4123**	**812**	**4842**	**9777**				
	Overall Accuracy		78.7%		Kappa		0.64		

Based on Table 5.8 the results are regarded as acceptable considering the low resolution of the imagery (250m). The *others* class has a high omission error compared to the *vineyard* and *wheat* classes, and this could be as a result of crops in the *others* class having a similar phenology to the *wheat* class, since only fields classified as pure wheat were selected from the SiQ data. The low producer's accuracy seen in the *vineyard* class could be attributed to misclassification due to the mixed pixel effect. Vineyards in the study area generally span smaller areas than can be effectively identified using medium resolution imagery. In addition, the *others* and *vineyard* class have a similar phenology during the growing and harvest season. This trend is a result of the NDVI value for vineyards remaining high throughout the year due to grass cover growing in the rows when they shed their leaves during winter, the rainy season. The *wheat* class was well represented based on high percentage of 88.89% and 79.31% in the producer's and consumer's accuracy respectively.

When compared to the classification performed using WEKA (80% accuracy & kappa 0.6) and ENVI (71% accuracy & kappa 0.5) software, the classification of MODIS imagery using MEAWAT can be considered successful with an accuracy of 78% and kappa of 0.6.

5.7 TRANSFERABILITY OF MEAWAT TO LANDSAT IMAGERY

The Landsat classification in MEAWAT commenced with the masking function, after which all other toolsets were executed on the Landsat image using the same parameters that were used for the MODIS classification. NDVI phenology was also generated from Landsat in order to train the classifiers to compare the similarity and therefore transferability of MEAWAT. Different sets of parameters and classifiers were tested on the Landsat imagery to determine which parameter combination will achieve a high accuracy. Based on this, the AdaBoost RF classifier was also used for the final image classification for comparison purposes with that of MODIS. The classification also resulted in the three agricultural classes: *others*, *wheat* and *vineyard*.

Since Landsat and MODIS imagery have very different properties, the MEAWAT toolbox had to be enhanced to perform the classification of Landsat data. Firstly, MODIS data products (MOD13Q1) were used, which meant that the images were already radiometrically and atmospherically corrected and NDVI calculated. In addition, cloud removal had already been done. Since Landsat imagery, after radiometric and atmospheric correction, was already in GEOTIFF format, the MEAWAT functionality of conversion from HDF format to GeoTIFF was redundant. The same training sample points were used for both MODIS and Landsat classifications, each with their own phenology. Due to cloud contamination, only 12 cloud-free Landsat images could be used for building the tree as opposed to the 23 images for the MODIS data. Landsat 8 data, which were used in the study, were also only available from mid-2013, so the entire 2014 year was analysed and compared to the 2013 MODIS data. Before classifying the Landsat imagery using MEAWAT, NDVI values were calculated from the corrected cloud-free Landsat images using the image analysis function in ArcGIS.

Various obstacles were encountered while testing the transferability of MEAWAT to different satellite imagery. These were overcome by modifications to the toolbox and input data. The first issue encountered while using MEAWAT on Landsat imagery was memory and data handling due to the improved radiometric resolution of Landsat 8 (16-bits compared to 8-bits for its predecessor). Scaling pixel values during the copy raster function has a negative effect on the NDVI values of the image. The correct conversion output is achieved by not scaling the pixel value, thus the NDVI values remain the same ranging from -10000 to 10000.

Due to the higher spatial resolution, the Landsat image size caused memory problems within the Python scripts. As a result of the memory error, an adjustment was made to the tool after considering various options to accommodate the inherent properties of Landsat image by running the classification simultaneously on multiple subsets of the image instead of the entire image at once. This functionality was implemented in the Python script. The classified agricultural land cover map created using the classification using MEAWAT is shown in Figure 5.9. It was expected that the agricultural classes would be better distinguishable due to the higher resolution of the Landsat imagery, but this was not the case as a large part of the *wheat* and *vineyard* class were classified as *others*.

Since the finer resolution of the Landsat imagery resulted in a much more heterogeneous scene than MODIS with many more distinguishable classes, a second Landsat land cover classification was created and additional sample points were generated to create training data for an additional agricultural class, namely *potatoes*. Data with no values or ambiguous NDVI values were removed from the training data. The AdaBoost RF classifier was also used for the image classification, resulting in a land cover map. With Landsat having a higher resolution than MODIS, it is believed to be able to delineate more agricultural classes provided good training data are used, hence the agricultural class *potatoes* which was substituted for the class *others*. The land cover map generated from the classification (Figure 5.10), showed better class representation as opposed to the classification carried out earlier using the SIQ training data for MODIS.

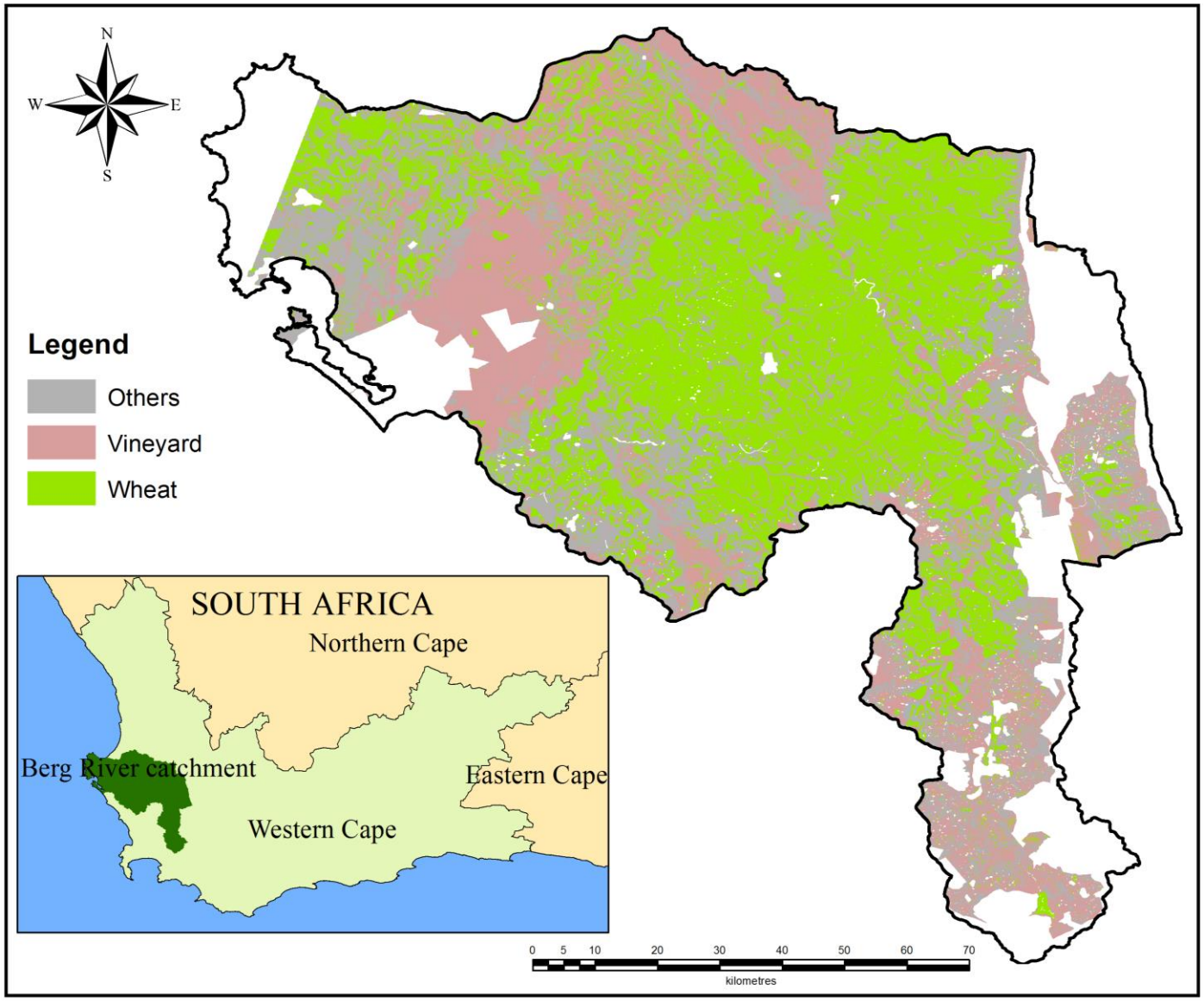


Figure 5.9 Landsat 2014 agricultural land cover classification in MEAWAT using SIQ data

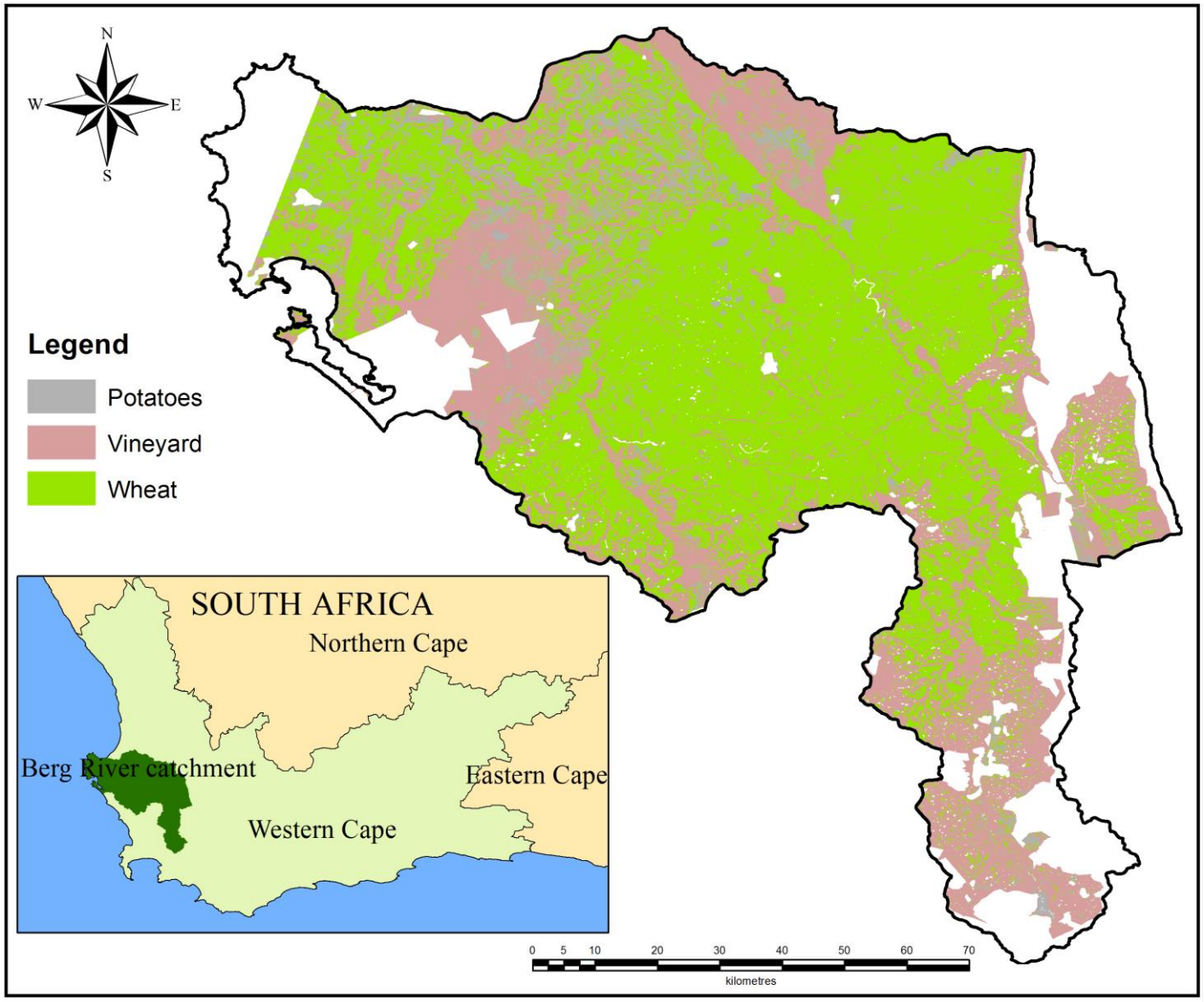


Figure 5.10 Landsat 2014 agricultural land cover classification in MEAWAT with an additional class

The Landsat land cover classification result from agricultural classes *others*, *wheat* and *vineyard* has a lower accuracy (62.3% kappa 0.4) (Table 5.9), when compared to the Landsat land cover classification result from the agricultural class *potatoes*, *wheat* and *vineyard* (89.3% kappa 0.7) (Table 5.10). The error of omission for the *vineyard* and *wheat* classes were 6.1% and 19.6% respectively which is greater when compared to that of MODIS (4.5% and 4.2% respectively). The highest producer's accuracy was observed in *wheat* class which indicated that it was a good representation of ground truth, although impaired by mixed pixel and misclassification.

Table 5.9 Confusion matrix for Landsat classification using MODIS sample points

Reference Data									
Classification Result		Others	Vineyard	Wheat	**Total**	% Error of Omission	% Error of Commission	Producer Acc %	User Acc %
	Others	1524	150	884	**2558**	10.37	24.46	59.58	33.39
	Vineyard	540	621	70	**1231**	6.10	2.08	50.45	74.91
	Wheat	1906	58	3810	**5774**	19.64	9.54	65.98	79.97
	Total	**3970**	**829**	**4764**	**9563**				
	Overall Accuracy		62.3%		Kappa		0.4		

The second Landsat classification generated based on sample points and training data different from those used with MODIS has a higher accuracy of 89% (Table 5.10) compared to accuracies derived from MODIS (78%) and the first Landsat classification of 62.3%. The Landsat classification with agricultural classes of *potatoes*, *vineyard* and *wheat* classes achieved the objective of demonstrating MEAWAT transferability with an overall accuracy of 89% and a kappa of 0.7. The omission error for all the classes was very low (<20%), and a better class representation was achieved. It was observed that the producer's accuracy of both *wheat* 97.2% and *vineyard* 78.5% were quite high compared to *potatoes* with 4.6%, which is as a result of the low representation of potatoes in the entire study.

Table 5.10 Confusion matrix for second Landsat classification with new training data

		Reference Data							
Classification Result		Potatoes	Vineyard	Wheat	**Total**	% Error of Omission	% Error of Commission	Producer Acc %	User Acc %
	Potatoes	14	23	266	**303**	2.89	2.2	4.62	38.89
	Vineyard	5	704	188	**897**	1.93	1.26	78.48	84.82
	Wheat	17	103	4305	**4425**	1.20	4.54	97.28	90.46
	Total	**36**	**830**	**4759**	**5625**				
	Overall Accuracy		89.3%		Kappa		0.7		

Based on the accuracies derived from the three classifications (MODIS in Table 5.8; Landsat in Table 5.9 and Table 5.10), the importance of good training data are once again established, as it has an influence on the output of the classification. Although issues of misclassification and mixed pixels will always occur, the classification can be improved if the classes are well represented. We can assume that the classifier is also sensitive to skewed or unbalanced training data (López et al 2013).

Based on the results achieved, MEAWAT has shown great potential for land cover classification, however for scientific purposes there is a need to compare the classification output of the different satellite imagery in order to understand MEAWAT transferability potential. This will be discussed in the next section.

5.8 COMPARISON BETWEEN MEAWAT MODIS AND LANDSAT 8 CLASSIFICATION

The land cover classification process of MODIS and Landsat images using MEAWAT were very similar, except for some extra pre-processing that was required for the Landsat image. Some functions within the resampling and pre-processing toolsets in MEAWAT used for MODIS pre-processing were no longer needed with Landsat processing as depicted in the workflow in Figure 5.11. The pre-processing steps unique to Landsat were: calculating NDVI values; scaling NDVI values to match those of MODIS; and copy raster to change the pixel depth. The Landsat process

only starts with the study area mask, cell size and layer stacking, while all the other toolsets in MEAWAT were used.

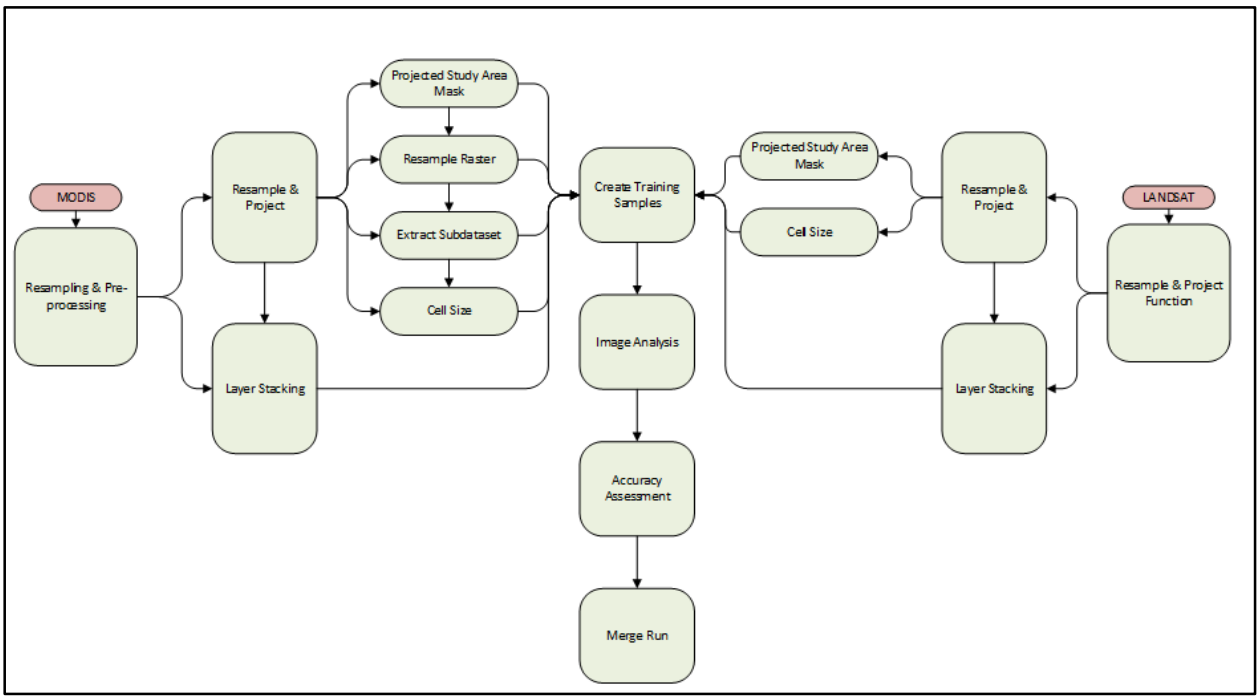


Figure 5.11 Comparison of MODIS and Landsat workflows in MEAWAT

Since MEAWAT could be used to successfully perform land cover classification of both MODIS and Landsat images the objective of demonstrating transferability was achieved. Based on both sensor characteristics, the accuracies achieved from the MODIS and Landsat classification were acceptable. The classification performed on the Landsat using MEAWAT default settings from MODIS training data, had a lower accuracy of 63% compared to MODIS having 78%. This was as a result of the training data used for the classification. For example, the class *others* was better represented in the MODIS classification than on the Landsat classification. In addition, the *wheat* and *vineyard* classes were better represented in the Landsat classification than the MODIS classification, evident from the high omission error (Table 5.8 and Table 5.9). Some of the mixed pixel challenges experienced with MODIS were reduced in the Landsat classification.

The result of the second Landsat classification using different sample points and training data, reaffirms the importance of good training data for successful image classification. With an accuracy of 89% and a kappa of 0.7, the Landsat classification in MEAWAT will be efficient for classifying smaller agricultural fields and has the potential for classifying different irrigated crops as demonstrated with class *potatoes*. The individual agricultural classes *potatoes*, *vineyard* and *wheat* were also better represented in the second Landsat classification. Similarly the *potatoes* class was well represented, even though some were misclassified as *wheat*, as a result of a similar phenology to wheat especially if planted in the winter season.

The result from the MODIS (Table 5.8) versus Landsat (Table 5.9) classifications is not an indication of superiority of one above the other. It rather shows the potential of using MEAWAT using other satellite imagery in order to contribute to the scientific community by speeding up land cover classification when working with large datasets and large area mapping.

5.9 CONCLUSION

This chapter provided the results derived from DT classification in ENVI, which could not be integrated into MEAWAT based on software and technical challenges. It is important to carefully consider creating training data for the classification. Comparison of the WEKA and ENVI classification explains the importance of machine learning algorithms over conventional image classification. The various classifiers tested provided an indication of recommended parameters to use in order to achieve a better classification result. From the study, it was evident that MEAWAT can be used successfully on MODIS and Landsat data. The next chapter comprises important conclusions drawn from the present study, and provides recommendations for future research and the tool created.

CHAPTER 6: CONCLUSION AND RECOMMENDATIONS

6.1 EVALUATION OF THE RESEARCH

Automation of land cover mapping over large areas can provide a cost effective, time management solution to manual classification and processing of many satellite imagery. Minimal human intervention during image processing will result in minimal error propagation thereby having the potential for better classification. The integration of analyses performed in various different software packages can be time consuming therefore combining these in an integrative platform such as MEAWAT, can save the user the effort of entering parameters for each single file to be processed, thereby reducing human error.

Some of the studies performed using an automated approach for image analysis were based on the conventional image analysis method (Awaad 2013; Verhegghen et al. 2009; Comber, Lawanr & Lishman 2004) which involves very little automation. The current study sought to establish whether MEAWAT can be used as an alternative way of classifying land cover using different satellite imagery with large datasets using just a single tool, instead of classifying using multiple software packages.

The aim of developing such a tool is to deliver a fast land cover classification of large datasets with less error propagated, using an automated approach with little human intervention, and applicable across a range of satellite data and study sites. In this study MEAWAT was tested with Landsat imagery and compared with results derived from MODIS imagery. The study also sought to develop a robust tool to be used on large datasets having different spatial resolutions.

Although MEAWAT was created with the aim of contributing to the scientific research community, some level of thinking and understanding of the tool is needed from the user to create a map for their purpose. Some of the benefits of using MEAWAT for automated land cover classification are highlighted in the next section.

6.2 POTENTIAL OF MEAWAT FOR AUTOMATED LAND COVER CLASSIFICATION

An automated workflow for land cover classification holds much potential as it provides a cost effective and simple way to achieve a better classification with large datasets. The most significant abilities of MEAWAT are that it is generic in nature, user-friendly and transferable to other satellite imagery. This transferability is important as it allows researchers to compare image analysis outputs from various satellite images with user specific parameters. Parameters are not hard coded, therefore the user can tweak the tool to suit their respective needs, and it can

accommodate large datasets, depending on the capability of the computer hardware used to run the process. Even users with little GIS and remote sensing background can make use of the tool, saving them time and resources to learn and master different GIS software, since classification can be performed in a single user interface using any of the classifications via Python script as opposed to other classification techniques which require additional GIS & RS software skills.

The division of the tool into categories or toolsets ensures that the tool is generic, giving users the opportunity to access any of the categories as a stand-alone toolset without necessarily using the whole tool. Accuracy assessment is imbedded into the tool, and is produced as an output alongside the classification. Since MEAWAT uses the ArcGIS framework, the user is not inconvenienced with the location of temporary files and data, because the tool allows the user to specify the relative path folder to save the data which is available in ArcGIS and also within the stand alone script. In cases where they fail to do so, data are saved to the scratch folder which is within the MEAWAT kit. The following section is a re-examination of the aim and objectives of the research in order to evaluate the success of the study.

6.3 RESEARCH OBJECTIVES REVISITED

The main aim of the research was to develop an automated technique for identifying agricultural land cover using time-series MODIS NDVI on an annual time step (one image per year) with transferability to other sensors and/or study areas. An automated customized toolset, MEAWAT, was developed for this purpose. The tool gave insight into the potential of using an automation approach when dealing with large dataset.

The first objective (Section 1.3) was to review the relevant literature and decide on which geo-processing tools and automated techniques were needed for land cover modelling. From the literature it was clear that MODIS and Landsat NDVI have successfully been used for vegetation studies. Chapter 2 described the multispectral imagery available, image analysis approaches, including machine learning, the use of ancillary and training data, and methodically reviewed how the accuracy of the map can be validated. In addition automated approaches for land cover mapping were discussed.

Chapter 3 detailed the requirement analysis and technical considerations for creating a system to perform land cover mapping using software integration through automation. This chapter also highlights the strength of geo-processing using the Python platform, how the data should be structured, which software routines are needed for integration, the significance of standards for the tool and the data to enable sharing.

Chapter 4 outlined the steps taken to achieve objectives two, three and four. The second objective of the research was, to collect and prepare applicable datasets to test the classification output. These data sets were essential to demonstrate the functionality of the tool and also highlighted the importance of creating "good" training data. Objectives three entailed the selection of the integrative platform and software tools. Python was chosen as the suitable language and platform, because it is easy to use and already incorporated into ArcMap. The tasks associated with the design and implementation of the tool (objective four), are also described in Chapter 4.

Demonstration of the use of MEAWAT (Chapter 5) satisfied the fifth objective, where the functionality of the tool was made evident by performing an agricultural land cover classification in the Berg river catchment area of Western Cape with MODIS data. Having achieved a good classification result, the tool was tested on Landsat imagery of the same study area which exhibits its transferability potential.

Unfortunately, due to lack of good training and validation data, the accuracy of land cover classification using MEAWAT on another study area was not tested. The validation results derived from classification of different sensors provided a detailed analysis for the attainment of the last objective of the research, which included evaluation of the quality of the classification, thereby highlighting the usefulness of the automated process to identify agricultural land cover classes. Despite challenges with regard to image format (12bit radiometric quantization) and associated larger size, MEAWAT could be used to derive similar classes during classification, and even identify additional classes not distinguishable within the larger MODIS pixels. Challenges and limitations of the study are discussed in the next section.

6.4 CHALLENGES AND LIMITATIONS OF THE STUDY

One of the first challenges encountered during the design phase of MEAWAT was the inability to integrate the WEKA and ENVI DT classification previously tested seamlessly into MEAWAT. The developers of WEKA software suggested the use of Jython, a rewrite of Python, to seamlessly integrate with Java, rather than Python or a Python wrapper with limited functionality for development. The use of Python functionality Scipy or Numpy in this implementation are limited and this type of integration with ArcGIS fell outside the scope of the study. Despite documented integration between ArcGIS and ENVI, the interface to directly link to the DT classification routine in ENVI, was deemed proprietary and unavailable. This challenge was overcome by using the open-source Scikit-learn Python package.

The benefit of using the open-source solution lies in the fact that MEAWAT can be available for use to anyone with only an ArcGIS license. Moreover, the need for additional licensing for ENVI software has also been removed as well as the requirement for advanced programming skills to integrate with WEKA.

Another limitation of the present study is that it was impossible to implement atmospheric and radiometric corrections in MEAWAT as a result of the specialised routines that were required. This limitation remains a challenge for future projects. Furthermore, MEAWAT was developed specifically for identifying land cover that can be identified using a unique phenological curve. Where different vegetation types have very similar phenologies, the tool may not be able to generate unique signatures and the classes may not be separable. This is the case with a crop such as vineyard in the winter rainfall region of the Western Cape, where the winter cover crop that may grow between the rows of vines create higher NDVI values than expected bare soil between the rows. However, since MEAWAT makes use of different layers of information to generate a unique signature from which classification can take place using different methods, additional information layers can help with the seperability of the classes. Though not tested, it would therefore be possible to use MEAWAT for more than just phonological curve classification from NDVI data.

In addition, the automation was developed within a particular hardware and software environment and extended testing would be required to roll the tool out for general use. The automation approach cannot be likened to an intelligent system or a robot, as it does not learn from past experiences or mistakes (PSU 2014, ESRI 2013), therefore may be affected by human or system error.

MODIS data provided good classification for large homogeneous agricultural fields achieving similar results as expected with OBIA, except that small fields could not be accurately characterized due to the spatial resolution of the pixels, but small fields could be delineated more accurately using Landsat data. Segmentation functionality was not implemented in MEAWAT, due to the complicated nature of development of such routines. The functionality was also not yet available in the ArcGIS software that was used for the development of the tool, but has subsequently been released. This falls into the recommendation following this section.

6.5 RECOMMENDATIONS

Since many accurate routines exist for Landsat atmospheric correction, it is recommended that any of such routines be integrated or embedded into the MEAWAT toolset of resampling and pre-processing, to improve or update the potential usefulness of MEAWAT as an image

processing toolset within the ArcGIS. In addition, the new segmentation functionality provided in ArcGIS 10.3 and above should be explored to widen the usability of MEAWAT.

Accuracy of any classification is based on an accurate training set and therefore the user must ensure that the training set supplied to be used for the classification is as accurate as possible (i.e. sample points and training data).

For enhanced classification results, future users of MEAWAT should make use of the suggested classifier parameters (Table 5.3, Table 5.4 and Table 5.5) and especially implement AdaBoost ensemble classifier for improved results.

Image processing, using any software (including MEAWAT), requires a computer with a large memory, preferably RAM of eight gigabyte and above to run effectively.

Further studies should investigate land cover classification of additional agricultural land cover classes, such as orchards, pastures and fruits. It is also recommended that land cover change analysis be added to the MEAWAT toolbox to expand the functionality and make it more widely applicable.

6.6 CONCLUSIONS

In conclusion, this study highlighted the potential of automation while performing land cover classification on a large volume of data. Automation was achieved through the development of a user friendly customized toolbox (MEAWAT) containing multiple toolsets for land cover classification. The study provided insight on the importance of selecting good training data, as this determines the success of the classification. The accuracies obtained from both the MODIS and Landsat classifications show a good representation of the selected agricultural classes, and changing trends in the classes over time could be seen from the resulting land cover maps, which could quantitatively be analysed.

The research has demonstrated the prospects of an automation approach for land cover modelling. The technique developed here can be used by future researchers for agricultural land cover classification. An automated tool such as MEAWAT has the potential to increase access to spatial analysis functionality as any user with little or no GIS and RS background can make use of the tool.

REFERENCES

Abburu S & Golla SB 2015. Satellite image classification methods and techniques: A review. *International Journal of Computer Applications* 119(8): 20-25.

Adesuyi A & Münch Z 2015. Using time-series NDVI to model land cover change: A case study in the Berg River catchment area, Western Cape, South Africa. *International Journal of Environmental, Chemical, Ecological, Geological and Geophysical Engineering* 9(5): 537-542.

Agone V & Bhamare SM 2012. Change detection of vegetation cover using remote sensing and GIS. *Journal of Research and Development* 2:4.

Ajay KN 2013. Advanced data structures in python [online]. Available from http://pypix.com/python/advanced-data-structures-python [Accessed 29 April 2014].

Aitkenhead MJ & Aalders IH 2011. Automating land cover mapping of Scotland using expert system and knowledge integration method. *Remote Sensing of Environment* 10: 10-16.

ArcGIS Resouce Centre 2010. Introducing the analysis and geoprocessing tool [online]. Available from http://resources.arcgis.com/en/communities/analysis.html [Accessed 2 February 2014].

Asmat A & Zamzami SZ 2012. Automated house detection and delineation using optical remote sensing technology for human settlement. *Procedia-Social & Behavioural Sciences* 36: 650-658.

Awwad WA 2003. Land cover mapping: a comparison between manual digitizing and automated classification of black and white historical aerial photography. Master's thesis. University of Florida.

Bajocco S, Dragoz E, Gitas I, Smiraglia D, Salvati L & Ricotta C 2015. Mapping forest fuels through vegetation phenology: The role of coarse-resolution satellite time-series. *PLoS ONE* 10:3.

Bartholome E & Belward AS 2005. GLC 2000: A new approach to global land cover mapping from earth observation data. *International Journal of Remote Sensing* 26(9): 1959-1977.

Bergstra J & Bengio 2012. Random search for hyper-parameter optimization. *The Journal of Machine Learning research* 13: 281-305.

Benz UC, Hofmann P, Willhauck G, Lingenfelder I, & Heynen M (2004). Multi-resolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information. *ISPRS Journal of photogrammetry and remote sensing* 58(3): 239-258.

Bhaskaran S, Paramananda S & Ramnarayan M 2010. Per-pixel and object-oriented classification methods for mapping urban features using Ikonos satellite data. *Applied Geography* 30: 650-665.

Blaschke T 2010. Object-based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing* 65: 2-16.

Boloorani AD, Erasmi S & Kappas M 2003. Urban land cover mapping using object/pixel-based data fusion and IKONOS images [online]. University of Goettingen, Institute for Geography, Department of Cartography & Remote Sensing. Goldschnidstr. 5D-37077. Available from http:///www.isprs.org/proceedings/XXXVII/part 7/pdf/129.pdf [Accessed 5 July 2014].

Brandmeyer JE & Karimi HA 2000. Coupling methodologies for environmental models. *Environmental Modelling & Software* 15(5): 479-488.

Breiman L 2001. Random Forests. *Machine Learning* 45(1): 5-32.

Brown JC, Khartens JH, Coutinho AC, Victoria DC & Bishop CR 2013. Classifying multilayer agricultural land use data from Mato Grosso using time-series MODIS vegetation index data. *Remote Sensing of Environment* 130: 39-50.

Campbell JB & Wynne RH 2011. *Introduction to remote sensing.* 5th ed. New York: The Guilford Press.

Carrao H, Goncalves P & Caetano M 2008. Contribution of multispectral and multi-temporal information from MODIS images to land cover classification. *Remote Sensing of Environment* 112(3): 986-997.

Chauhan H & Chauhan A 2014. Evaluating performance of decision tree algorithms. *International Journal of Scientific and Research Publications* 4:4.

Chen J, Johnson P, Tamura M, Gu Z, Matushita B & Eklundh L 2004. A simple method for reconstructing a high quality NDVI time-series data based on the Savitaky-Golay Filter. *Remote Sensing of Environment* 91: 332-344.

Cheng W, Zhang X, Wang K & Dai X 2009. Integrating classification and regression tree (CART) with GIS for assessment of heavy metals pollution. *Environmental Monitoring and Assessment* 158: 419-431.

Cihlar, J 2000. Land cover mapping of large areas from satellites: Status and research priorities. *International Journal of Remote Sensing* 21: 1093-1114.

Clark B & Ratcliffe G (eds) 2007. Berg River baseline monitoring programme. Final report volume 5: Synthesis. Pretoria: Department of Water Affairs and Forestry.

Cleveland CC 2009. Automating GIS tasks [online]. Available from http://www.gis-sig.org/programs/automate/automateGISTasks.pdf Presented at GIS/SIG summer programme. [Accessed 27 April 2014].

Colditz RR 2007. Time series generation and classification of MODIS data for land cover Mapping. PhD thesis. University of Wurzburg.

Comber AJ, Law ANR & Lishman JR 2004. Application of knowledge for automated land cover change monitoring. *International Journal of Remote Sensing* 25(16): 3177-3192.

Congalton RG & Green K 2009. Accessing the accuracy of remotely sensed data; Principles and Practices. 2nd ed. Florida: CRC Press.

Da Silva Brum V, Garcia LM, Kortry TS, Duque CM & Coelho dos RD 2013. Intra-urban land cover classification using IKONOS II images and data mining technique: A comparative analysis [online]. *Journal of Urban Remote Sensing Event* (JURSE). Available from http://www.ieeexplore.ieee.org/stamp/stamp.Jsp?tp=anumber=6550703 [Accessed 4 July 2014].

Dahal KR & Chow TE 2014. A GIS toolset for automated partitioning of urban lands. *Environmental Modelling & Software* 55: 222-234.

Dangermond J 2009. GIS: Design and evolving technology [online]. ArcNews, ESRI, Fall. Available from http://downloads2.esri.com/campus/uploads/library/pdfs/148907.pdf [Accessed 2 March 2016].

DeFries RS & Chan JC 2000. Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. *Remote Sensing Environment* 74: 503-515.

Desclee B, Bogaert P & Defourny P 2006. Forest change detection by statistical object-based method. *Remote Sensing of Environment* 102: 1-11.

Didan K & Huete A 2006. MODIS vegetation index product series: Collection 5 change summary. TRBS lab: University of Arizona.

Duong ND 2000. Land cover category definition by image invariants for automated classification. *International Archives of Photogrammetry and Remote Sensing* 33(7): 3.

DWA 2013. WC WSS Reconciliation strategy report. Pretoria: Department of Water Affairs, Government of South Africa.

EE Publishers 2015. Africa user conference kicks off successfully [online]. Available from http://www.ee.co.za/article/africa-user-conference-kicks-off-successfully.html [Accessed 3 March 2016].

Ehlers M, Gachter M & Jawonsky 2006. The integration of remote sensing and GIS. Sensors & systems [online]. Available from http://www.sensorandsystems.com/article/features/29580 [Accessed 5 May 2014].

ESRI 2013. ArcGIS resource centre / Automation [online]. Available from http://www.resources.arcgis.com/en/help/main/10.1 [Accessed 29 April 2014].

ESRI 2003. Spatial data standards and GIS interoperability [online]. Available from http://www.esri.com/~/media/files/pdfs/library/whitepapers/pdfs/spatial-data-standards.pdf [Accessed 2 May 2014].

eWISA 2008. The Berg River [online]. WAMTechnology, Stellenbosch. Available from: http://www.ewisa.co.za/misc/school/ebook_pdfs/thebergriver.pdf [Accessed 22 February 2013].

Exelis 2013a. Exelis Visual Information Solution [online]. Available from http://www.exelisvis.com/docs/ROIs.html [Accessed 3 October 2013].

Exelis Visual Information Solution 2013b. Integrating ENVI with ArcToolbox and Model builder [online]. Available from www.exelisvis.com/learn/whitepaperdetail/tabId/802/ArtMID/2627/ArticleID/12403/integrating.aspx [Accessed 2 February 2014].

Faour G & Kheir RB 2006. Effectiveness of using very high resolution imagery (IKONOS) for land use mapping. National Council for Scientific Research. Remote Sensing Centre, 8281 Beirut, Lebanon. Available from http://www.gisdevelopment.net/technology/ip/techip-002pf.html [Accessed on 4 July 2014].

Foody GM 2002. Status of land cover classification accuracy assessment. *Remote Sensing of Environment* 80: 185-201.

Frohlich B, Bach E, Walde I, Hese S, Schillius C & Denzler J 2013. Land cover classification of satellite images using contextual information. ISPRS annals of photogrammetry. *Remote Sensing & Spatial Information Science* 11: 3.

Frost GV, Epstein HE & Walker DV 2014. Regional and landscape – scale variability of Landsat observed vegetation dynamics in North West Siberian tundra. *Environment Research Letters* 9: 1-6.

Furlanello C, Neteler M, Merler S, Menegon S, Fontanari S, Donini A, Rizzoli A & Chemini C 2003. GIS and the random forest predictor: Integration in R for tick-borne disease risk assessment [online]. In Hornik K, Leisch F & Zeileis (eds). Proceedings of the 3rd

international workshop on Distributed Statistical Computing (DSC) held 20-22 March 2003, Vienna, Austria.

Geoscience Australia 2012. Earth observation and satellite imagery [online]. Available from http://www.ga.gov/au/earth-observation/satellites-and-sensor/modis.html [Accessed 5 May 2013].

Geurts P, Erust D & Wehenkel L 2006. Extremely randomized tree. *Machine Learning* 63: 3-42.

Gibson PJ 2000. Introductory remote sensing: Principles and concepts. London: Routledge.

Giri C & Jenkins C 2005. Land cover mapping of greater Mesomerica using MODIS data. *Canadian Journal of Remote Sensing* 31(4): 274-282.

Giri C & Long J 2014. Land cover characterization and mapping of South America for the year 2010 using Landsat 30m satellite data. *Remote Sensing* 6(10): 9494-9510.

Gong JM, Yang XM, Lu J, Lin ZJ, Su FZ, Du YY & Jiang Z 2008. Land cover classification at a scale of 1: 50000 in Sanjiangyuan study area based on SPOT5 images. *The International Archives of Photogrammetry, Remote Sensing & Spatial Information Sciences* 37(4): 1835-1840.

Goodchild MF 2006 Geographical information science: fifteen years later. In P.F. Fisher, editor, classics from IJGIS*: Twenty years of the International Journal of Geographical Information Science and Systems*. Boca Raton: CRC Press 424: 199-204.

Gopal S, Woodcock CE, Strahler AH 1999. Fuzzy neural network classification of global land cover from a 1 degree AVHRR data set. *Remote Sensing of Environment* 67(2): 230-243.

Guo Y & Zeng F 2012. Atmospheric correction comparison of SPOT-5 image based on model FLAASH and model QUAC. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 39:7.

Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH 2009. The WEKA data mining software: An update; *SIGKDD Explorations* 11:1.

Haiden S, Midgley SJE & New M. 2014. The food energy water land biodiversity (FEWLB) nexus in the Berg River catchment, Western Cape. A synthesis of stakeholder perspectives. African climate and development initiative. University of Cape Town.

Hansen M, Dubayah R & DeFries R 1996. Classification trees: An alternative to traditional land-cover classifiers. *International Journal of Remote Sensing* 17(5): 1075-1081.

Hastie T, Tibahirani R & Friedman J 2009. Elements of statistical learning. 2nd ed. New York: Springer-Verlay.

Her Y & Heatwole C 2007. Land use classification in Zambia using Quickbird & LANDSAT imagery. Paper presentation at ASABE annual international meeting held 17-20 June 2007, Minneapolis, Minnesota.

Huang X & Jensen JR 1997. A Machine-Learning approach to automated knowledge-based building for remote sensing image analysis with GIS data. *Photogrammetric Engineering & Remote Sensing* 63(10): 1185-1194.

Hu Q, Wu W, Xia T, Yu Q, Yang P, Li Z & Song Q 2013. Exploring the use of Google Earth imagery and object-based methods in land use / cover mapping. *International Journal of Remote Sensing* 5: 6026-6042.

Huete A, Justice C & Van leeuwen W (1999). MODIS vegetation index (MOD13) algorithm theoretical basis document (ATBD). EOS project office. Greenbelt: NASA Goddard Space Flight Center.

Huth J, Kuenzer C, Wehrmann T, Gebhardt S, Tuan VQ & Dech S 2012. Land cover and land use classification with TWOPAC: Towards automated processing for pixel and object-based image classification. *International Journal of Remote Sensing* 4(9): 2530-2553.

Ioannis M & Meliadis M 2011. Multi-temporal LANDSAT image classification and change analysis of LULC in the prefecture of Thessaloniki Greece. Proceedings of the International Academy of Ecology & Environmental Sciences 1: 15-25.

Irons JR, Dwyer JL & Barsi JA 2012. The next Landsat satellite. The Landsat data continuity mission. *Remote Sensing of Environment* 122: 11-21.

Ismail MH & Jusoff K 2008. Satellite data classification accuracy assessment based from reference dataset. *International Journal of Environmental, Ecological, Geological and Geophysical Engineering* 2(3): 23-29.

Jackson TJ, Chen D, Cosh M, Li F, Anderson M, Walthall C, Doriaswamy P & Hunt ER 2004. Vegetation water content mapping using Landsat data derived normalized difference water index for corn & soybeans. *Remote Sensing of Environment* 92: 457-482.

Jaloree S, Rajput A & Gour S 2014. Decision tree approach to build a model for water quality. *Binary Journal of Data Mining & Networking* 4: 25-28.

Jiang D, Huang Y, Zhuang D, Zhu Y, Xu X, & Pen H 2012. A simple semi-automatic approach for land cover classification from multi-spectral remote sensing imagery. *PLOSONE* 7:9.

Jonsson P & Eklundh L 2002. Seasonality extraction by function fitting to time series of satellite sensor data. *IEEE Transaction on Geoscience and Remote Sensing* 40:8.

Justice CO & Townshend JRG 2002. Special issue on Moderate Resolution Imaging Spectroradiometer (MODIS): A new generation of land surface monitoring. *Remote Sensing of Environment* 83: 1-2.

Kaunda-Bukenya N, Tadesse W, Fu Y, Tsegaye T & Wagaw M 2012. Spatial decision support system (SDSS) for stormwater management and water quality assessment, water quality monitoring and assessment [online]. Available from http://www.intechopen.com/books/water-quality-monitoring-and-assessment/spatial-decision-support-systemfor-urban-stormwater-management-and-water-quality-assessment [Accessed 31 December 2014].

Keith DA, Orscheg C, Simpson CC, Clarke PJ, Hughes L, Kennelly SJ, Major RE, Soderquist TR, Wilson AL & Bedward M 2009. A new approach and case study for estimating extent and rates of habitat loss for ecological communities. *Biological Conservation* 142: 1469-1479.

Keuchel J, Naumann S, Heiler M & Siegmund A 2003. Automatic land cover analysis for Tenerife by supervised classification using remotely sensed data. *Remote Sensing of Environment* 86(4): 530-541.

Kim HO & Yeom JM 2012. Multi-temporal spectral analysis of rice fileds in South Korea using MODIS and RapidEye satellite imagery. *Journal of Astronomy and Space Sciences* 29(4): 407-411.

Kim HO, Yeom JM & Kim YS 2011. Agricultural land cover classification using Rapid-eye satellite imagery in South Korea. *Remote Sensing for Agriculture, Ecosystems and Hydrology* 13: 817424.

Kindu M, Schneider T, Teketay D & Knoke T 2013. Land use/land cover change analysis using object based classification approach in Munessa-Shashenene landscape of Ethopian Highlands. *Remote Sensing Journal* 5: 2411-2435.

Kleynhans W, Olivier JC, Wersels KJ, Salmon BP, Bergh F & Steenkamp K 2011. Detecting land cover change using an Extended Kaman filter on MODIS NDVI time-series. *IEEE Geoscience and Remote Sensing letters* 8(3): 507-511.

Knight JF, Lunetta RL, Ediriwickrema J & Khorram S 2006. Regional scale land cover characterisation using MODIS NDVI 250m multi-temporal imagery. A phenology based approach. *GIScience and Remote Sensing* 43(1): 1-23.

Knorn J, Rabe A, Radeloff VC, Kuemmerle T, Kozak J & Hostert P 2009. Land cover mapping of large areas using chain classification of neighbouring Landsat satellite images. *Remote Sensing. Environment* 113: 957-964.

Kohl M, Magnussen SS & Marchetti M 2006. Sampling methods, remote sensing and GIS multiresource forest inventory. Berlin Heidelberg: Springer- Verlag.

Kokalj Z & Ostir K 2007. Land cover mapping using Landsat satellite image classification in the classical Karst – Kras region. *AITA CARSOLOGICA* 36(3): 433-440.

Kong C, Kai X & Wu C 2006. Classification and extraction of urban land-use information from high-resolution image based on object multi-features. *Journal of China University of Geosciences* 17: 151-157.

Kraft P, Multsch S, Vaché KB, Frede HG & Breuer L 2010. Using Python as a coupling platform for integrated catchment models. *Advances in Geosciences* 27(27): 51-56.

Kux HJH, Novack T, Ferreira R & Oliveira DA 2011. Urban land cover classification using optical VHR data and the knowledge-based system inter-image. *The International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* 38(4):7.

Larocque GR, Bhatti J & Arsenault A 2014. Integrated modelling software platform development for effective use of ecosystem models. *Ecological Modelling* 288: 195-202.

Lazar A & Shellito BA 2009. Classification in GIS Using Support Vector Machines [online]. Available from http://www.irma-international.org/viewtitle/20393/ [Accessed 15 February 2016].

Lewinski ST & Bochenek Z 2008. Rule based classification of SPOT imagery using object-oriented approach for detailed land cover mapping. Proceedings of the 28th EARSel Symposium, "Remote sensing for a changing Europe" held 2-5 June 2008, Istanbul, Turkey.

Li P, Jiang L & Feng Z 2014. Cross-comparison of vegetation indices derived from Landsat 7 enhanced thematic mapper plus (ETM+) and Landsat 8 Operational land imager (OLI) sensor. *Remote Sensing* 6: 310-329.

Liang S 2001. Land cover classification methods for multi-year AVHRR data. *International Journal of Remote Sensing* 22(8): 1479-1493.

Lillesand TM, Kiefer RW & Chipman JW 2004. Remote sensing and image interpretation. 5th ed. New York: John Wiley and Sons.

Lim HS, Matjafri MZ & Abdullah K 2009. Land cover classification over Penanag Island, Malaysia using SPOT data. *International Journal of the Computer, the Internet & Management* 17:1.

Liu JY, Zhuang DF, Luo D & Xiao X 2003. Land cover classification of China: integrated analysis of AVHRR imagery and geophysical data. *International Journal of Remote Sensing* 24(12): 2485-2500.

López V, Fernández A, García S, Palade V & Herrera F 2013. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences* 250: 113-141.

Louppe G & Geurts P 2012. Ensembles on random patches. *Machine learning and knowledge discovery in databases* 1(7543): 346-361.

Loveland TR, Reed BC, Brown JF, Ohlen DO, Zhu Z & Yang L 2000. Development of a global land cover characteristic database and IGBP DISCover from 1 km AVHRR data. *International Journal of Remote Sensing* 21(6&7): 1303-1330.

Lowry J, Ramsey RD, Thomas K, Schrupp D, Sajwaj T, Kirby J, Waller E, Schrader S, Falzarano S, Langs L & Manis G 2007. Mapping moderate-scale land-cover over very large geographic areas within a collaborative framework: a case study of the Southwest Regional Gap Analysis Project (SWReGAP). *Remote Sensing of Environment* 108(1): 59-73.

Lu L, Kuenzer C, Guo H, Li Q, Long T & Li X 2014. A novel land cover classification map based on a MODIS time-series in Xinjiang China. *International Journal of Remote Sensing* 6: 3387-3408.

Luccio M 2013. The integration of remote sensing and GIS. Sensors and Systems [online]. Available from http://www.sensorandsystems.com/article/features/29580 [Accessed 5 May 2014].

Lunneta RS, Knight JF, Ediriwickrema J, Lyon JG & Worthy LD 2006. Land cover change detection using multi-temporal MODIS NDVI data. *Remote Sensing of Environment* 105: 142-154.

Maguire DJ, Batty M & Goodchild MF 2005. GIS, spatial analysis, and modeling. Redlands, CA: ESRI Press.

Mallinis G, Pleniou M & Koutsias N 2010. Object-based vs pixel-based mapping of fire scars using multi-scale satellite data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38(4): 7.

Manandhar R, Odeh IOA & Ancev T 2009. Improving the accuracy of land use and land cover classification of Landsat data using post-classification enhancement. *Remote Sensing* 1: 330-344.

Maree R, Geurts P, Visimberga G, Piater J & Wehenkel L 2003. A comparison of generic machine learning algorithms for image classification [online]. Available from http://www.montefiore.ulg.ac.be/services/stochastic/pubs/2003/MGVPW03/maree-ai2003-comparison.pdf [Accessed 11 November 2014].

Maskora Z, Zemek F & Kvet J 2008. Normalized difference vegetation index (NDVI) in the management of mountain madows. *Boreal Environment Research* 13: 417-432.

Mather PM & Koch M 2011. Computer processing of remotely sensed images: An introduction. John Wiley & Sons.

Mathieu R, Aryal J & Chong AK 2007. Object-based classification of IKONOS imagery for mapping large-scale vegetation communities in urban areas. *Sensors* 7: 2860-2880.

Medingegneria 2009. Iraq Marshlands: Multi-temporal monitoring of land cover [online]. Available from http://www.medingegneria.it/files/download/MODIS_MED_web.pdf [Assessed 22 February 2013].

Mekru M, Moulin B & Bergeron N 2012. Automated generation of informed virtual geographic environment using GIS data. Proceedings of global geospatial conference held 14-17 May 2012, Quebec City, Canada.

Mitchell S 2003. An extensive examination of data structure [online]. Available from https://msdn.microsoft.com/en-us/library/aa289148(v=vs.71).aspx [Accessed 8 February 2016].

Morton DC, Defries RS & Shimabukwo YE 2013. LBA-ECO LC-22 Land cover from MODIS vegetation indices. Mato Grosso Brazil. Available from http://ftp.daac.ornl.gov/data/lba [Accessed 29 April 2014].

Myint SW, Gober P, Anthony B, Grossman-clarke S & Weng Q 2011. Per-pixel vs object based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment* 115: 1145-1161.

NASA (National Space and Aeronautics Agency) 2011. The Landsat program [Online]. Available from: http://Landsat.gsfc.nasa.gov [Accessed: 20 September 2011].

Navulur K 2007. Multispectral image analysis using the object oriented paradigm. Boca Raton: CRC Press.

OGC 2013. OGC Standards [online]. Open Geospatial Consortium Inc. Available from hhtp://www.opengeospatial.org/standards/is [Accessed 2 May 2014].

Otukei JR & Blaschke T 2010. Land cover change assessment using decision trees, support vector machines, and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation* 12: 27-31.

Orhan O, Ekercin S & Dadaser-Celik F 2014. Use of Landsat land surface temperature & vegetation indices for monitoring drought in the salt lake basin area, Turkey. *The Scientific World Journal* 142938: 1-11.

Ozdogan M, Yang Y, Allez G & Cewantes C 2010. Remote sensing of irrigated agriculture: Opportunities and challenges. *International Journal of Remote Sensing* 2(9): 2274-2304.

Ozdogan M & Gutman G 2008. A new methodology to map irrigated areas using multi-temporal MODIS ancillary data: An application example in the continental US. *Remote Sensing of Environment* 112: 3520-3537.

Pal M & Mather PM 2001. Decision tree based classification of remotely sensed data. Proceeding of 22nd Asian conference on remote sensing held 5-9 November 2001, Singapore.

Pal M & Mather PM 2003. An assessment of the effectiveness of decision tree methods for land-cover classification. *Remote Sensing of Environment* 86: 554−565.

Pandya R & Pandya J 2015. C5.0 Algorithm to improved decision tree with feature selection and reduced error pruning. *International Journal of Computer Applications* 117:16.

Panju DR & Trisasongko BH 2012. Seasonal pattern of vegetation cover from NDVI time-series tropical forest [online]. Available from http://www.intechopen.com/books/tropical-forests/seasonal-pattern-of-vegetative-cover-from-ndvi-time-series [Accessed 4 November 2014].

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M & Duchesnay E 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12: 2825-2830.

Pensylvania State University (PSU) 2014. GIS programming and automation [online]. Geography 485. Available from http://www.e-education.psu.edu/geog485/node/30 [Accessed 27 April 2014].

Pericles T 2007. Comparison of AVHRR, MODIS and vegetation for land cover mapping and drought monitoring at 1 km spatial resolution. PhD thesis. Ceres: Cranfield University.

Pijanowski BC, Brown DG, Shellito BA & Manik GA 2002. Using neural networks and GIS to forecast land use changes: a land transformation model. *Computers, environment and urban systems* 26(6): 553-575.

Pimpler E 2013. Programming ArcGIS 10.1 with Python cookbook. Brimingham: PACKT Publishing.

Pradhan R, Ghose MK & Jeyaram A 2010. Land cover classification of remotely sensed satellite data using bayesian and hybrid classifier. *International Journal of Computer Applications* 7(11): 975-8887.

Punia M, Joshi PK & Porwal MC 2011. Decision tree classification of land use land cover for Delhi, India using IRS-P6 AWiFS data. *Expert Systems with Applications* 38(5): 5577-5583.

Pu R, Landry S & Yus Q 2011. Object based urban detailed land cover classification with high spatial resolution IKONOS imagery. *International Journal of Remote Sensing* 32(12): 3285-3308.

Python 2014. The python tutorial [online]: Data structure. Available from http://www.docs.python.org/2/tutorial/datastructures.html [Accessed 2 May 2014].

Quinlan JR (1993). C4.5: programs for machine learning. San Mateo, California: Morgan Kauffmann Publishers.

RHP (River Health Programme) 2004. State-of-rivers report: Berg River system. Pretoria: Department of Water Affairs and Forestry.

Ridgeway G 2007. Generalized boosting models: A guide to the GBM package. *Computing Science and Statistics* 31: 172-181.

Rogan J, Franklin J, Stow D, Miller J, Woodcock C & Roberts D 2008. Mapping land cover modifications over large areas: A comparison of machine learning algorithms. *Remote Sensing of Environment* 112: 2272-2283.

Rogan J & Miller J 2006. Integrating GIS and remotely sensed data for mapping forest http://wwwclarku.edu/department/geography/pdf/miller%20book%20chapt6.pdf [Accessed 5 May 2014].

Rogan J, Miller J, Stow D, Franklin J, Levien L & Fischer C 2003. Land-cover change monitoring with classification trees using Landsat TM and ancillary data. *Photogrammetric Engineering & Remote Sensing* 69(7): 793-804.

Rokos D & Argislas D 1995. Study of forest vegetation regeneration based upon Landsat TM images analysis: Preliminary results. *Earsel Advances in Remote Sensing* 4: 3-13.

Sah S, Van Aardt JAN, Mckeown DM & Messinger DW 2012. A multi-temporal analysis approach for land cover mapping in support of nuclear incident response. Proceeding of SPIE. Algorithms and technology for multi-spectral, hyper-spectral & ultra-spectral imagery 18:8.

Sanner MF 1999. Python: a programming language for software integration and development. *Journal of Molecular Graphics and Modelling* 17(1): 57-61.

Santos T, Tenedorio JA & Encarnacao S 2007. Comparing pixel vs object based classifiers for land cover mapping with ENVISAT-MERIS data. Netherland: Mill Press Rotterdam.

Schulthess U, Weichet H, Gunder S, Steine C, Schelly K, Reigber S, Hober E, Gruner V & Jug-Rothenhausler F 2008. Multi-temporal land cover classification approach with new Rapid-eye image data. Proceedings of 2nd workshop of the EARSel SIG on land use & Land cover held 28-30 September 2006.

Sharma R, Ghosh A & Joshi PK 2013. Decision tree approach for classification of remotely sensed satellite data using open source support. *Journal of Earth System Science* 122, 5: 1237-1247.

Spruce JP, Sader S, Ryan RE, Smoot J, Kuper P, Ross K, Prados D, Russel J, Gasser G, Mckellip R & Hargrove W 2011. Assessment of MODIS NDVI time-series data products for detecting forest defoliation by gypsy moth outbreaks. *Remote Sensing of Environment* 115: 427-437.

Stuckenberg T, Münch Z & van Niekerk A. 2013. Multi-temporal remote sensing land-cover change detection for biodiversity assessment in the Berg River Catchment. *South African Journal of Geinformatics* 2(3): 189-205.

Sucic S & Capuder T 2016. Automation of flexible distributed multi-generation systems by utilizing optimized middleware platform. *Applied Energy* 169: 542-554.

Sugumaran R & Degroote J 2010. Spatial decision support systems: Principles and practices. Boca Raton, Florida: CRC Press.

Tapsall B, Milenov P & Tasdemir K 2010. Analysis of Rapid-eye imagery for annual land cover mapping as an aid to European Union (EU) common agricultural policy. Proceedings of ISPRS TC VII symposium: 100 Years ISPRS held 5-7 July 2010. Vienna, Austria 38(7): 568-573.

Tateishi R & Mukouyama Y 2008. Land cover classification using SPOT data. Remote sensing & Image research centre, Chiba: Chiba University.

Tiede D, Luthje F & Baraldi A 2014. Automatic post classification land cover change detection in LANDSAT images: Analysis of changes in agricultural areas during the Syrian crisis.

Proceeding from Gemeinsame Tagung Conference held 26-28 March 2014. Hamburg, Germany.

Toms S 2015. *ArcPy and ArcGIS – Geospatial analysis with Python*. Birmingham: Pakt Publishing.

Torma M 2013. Land cover classification of FINNISH LAPLAND using decision tree classification algorithm. *The photogrammetric Journal of Finland* 23: 2.

Tyrallora L & Gonschorek J 2012. Spatio-Explorative analysis and its benefits for a GIS integrated automated feature identification. *Computational science and its applications* 7334: 220-233.

Turner AA 2012 (Ed.). Western Cape Province state of biodiversity 2012. CapeNature scientific services, Stellenbosch.

Van Niekerk A 2008. Clues: A web-based land use expert system for the Western Cape. PhD thesis. Stellenbosch: Stellenbosch University.

Van Wyngaarden R & Waters N 2007. An unfinished revolution – gaining perspective on the future of GIS. *GeoWorld* September: s.p.

Verhegghen A, Ernst C, Defourny P, Beuche R 2009. Automated land cover mapping and independent change detection in tropical forst using multi-temporal high-resolution dataset. *The International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* 38(4): 7.

Vermeulen D 2011. Evaluation of atmospheric correction methods for SPOT-5 imagery. Honours assignment. Stellenbosch: Stellenbosch University, Department of Geography and Environmental Studies.

Wang J, Sammis TW, Gutschick VP, Gebremichael M, Dennis SO & Harisson RE 2010. Review of satellite remote sensing use in forest health studies. *The Open Geography Journal* 3: 28-42.

Wang W, Hu HX & Hu J 2009. Land cover change detection based on MODIS 250m vegetation index time-series data. Proceeding of 17th International Conference on Geoinformatics. Fairfax VA USA 16: 1-6.

Wardlow BD & Egebert SL 2010. A comparison of MODIS 250m EVI and NDVI data for crop mapping: Case study for southwest Kansas. *International Journal of Remote Sensing* 31(3): 805-830.

Wardlow BD, Egebert SL & Kastens JH 2007. Analysis of time-series MODIS 250m vegetation index data for crop classification in the US Central Plains. *Remote Sensing of Environment* 108(3): 290-310.

Wardlow BD & Egebert SL 2008. Large area crop mapping using time-series MODIS 250m NDVI data: An assessment for the US Central Great Plains. *Remote Sensing of Environment* 112: 1096-1116.

Wehrmann T, Dech S & Glaser R 2005. An automated object based classification approach for updating conine land cover data. Webling: DLR-DFD German Remote Sensing Data Centre.

Weng Q 2011. Advances in environmental remote sensing, sensors, algorithms & applications. Florida: CRC Press.

Wessels KJ, Defries RS, Dempewolf J, Anderson LO, Hansen AJ, Powell SL & Moran EF 2004. Mapping regional land cover with MODIS data for biological conservation: examples from the greater Yellowstone ecosystem. USA and Brazil. *Remote Sensing of Environment* 92: 67-83.

Witten IH & Frank EIBE 2000. Data mining: Practical machine learning tools and techniques with java implementation. San Francisco: Morgan Kanfmann Publishers.

Wolfe RE, Nishihama M, Fleiga AJ, Kuypera JA, Roy DP, Storey JC & Patt FS 2002. Achieving sub-pixel geolocation accuracy in support of MODIS land science. *Remote Sensing of Environment* 83: 31-49.

Wunderlich AL 2012. Automation in ArcGIS 10: Understanding changes in methods of customization & options for migration of legacy code [online]. Proceeding of Digital Mapping Technique workshop held 16-19 May 2010, Carlifornia USA. Available from http://www.pubs.usgs.gov/of 2012/1171 [Accessed 28 March 2014].

Yang F, Yang J, Wang J & Zhu Y 2015. Assessment and validation of MODIS and GEOV1 LAI with ground-measured data and an analysis of the effect of residential area in mixed pixel. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8: 2.

Yang J, Chen F, Xi J, Xie P & Li C 2014. A multitarget land use change simulation model based on cellular automata and its application. *Abstract and Applied Analysis* 14: 1-11.

Yacouba D, Guangduo H & Xingping W 2009. Assessment of land cover / land use changes using NDVI and DEM in Puer & Sima counties Yuman Province China. *World Rural Observations* 1(2): 1-11.

Yu Q, Gong P, Clinton N, Biging G, Kelly M & Schirokauer D 2006. Object based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. *Photogrammetric Engineering and Remote Sensing* 72: 799-811.

Zandbergen P 2013. Python scripting for ArcGIS. Carlifonia: Esri Press.

Zhan X, Sohlberg RA, Townshend JRG, Dimiceli C, Carroll ML, Eastman JC, Hansen MC & Defries RS 2002. Detection of land cover changes using MODIS 250m data. *Remote Sensing of Environment* 83: 336-350.

Zhang X, Friedl MA, Schaaf CB, Strahler AH, Hodges JCF, Gao F, Reed BC & Heute A 2003. Monitoring vegetation phenology using MODIS. *Remote Sensing of Environment* 84: 471-475.

Zhao D, Huang L, Li J & Qi J 2007. A comparative analysis of broadband and narrowband derived vegetation indices in predicting LAI and CCD of a cotton canopy. *ISPRS Journal of Photogrammetry & Remote Sensing* 62: 25-33.

Zhao X, Stein A, Wang TJ, Chen XL & Tian LQ 2012. Accuracy assessment of extensional uncertainty modelled by random sets. London: Taylor & Francis Group.

Zhou J, Jia L & Menenti M 2015. Reconstruction of global MODIS NDVI time series: Performance of Harmonic Analysis of Time Series (HANTS). *Remote Sensing of Environment* 10: 1-12.

APPENDICES

APPENDIX A: MODIS TIME-SERIES LAND COVER MAPS

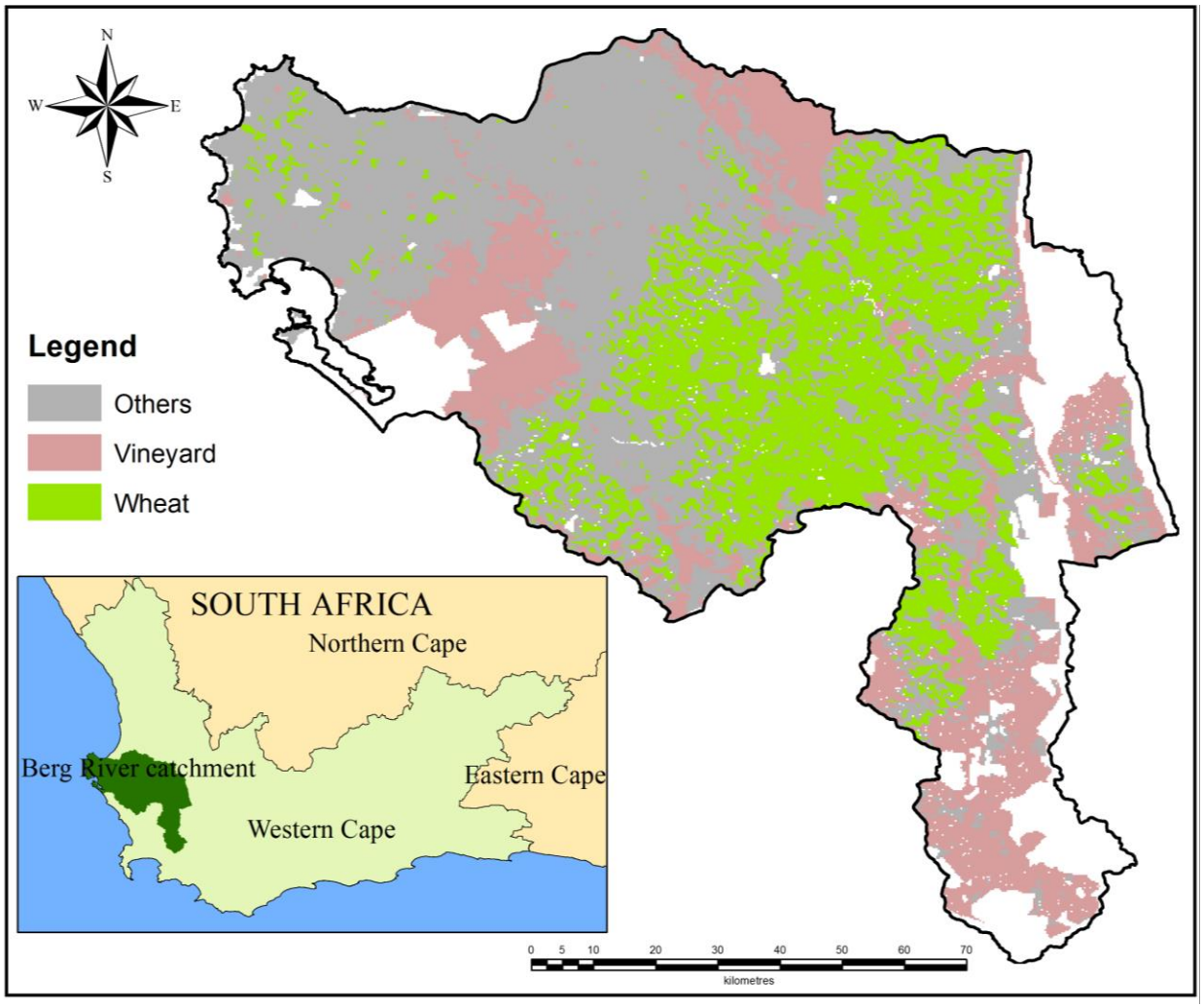


Figure A.1 MODIS land cover map of 2007

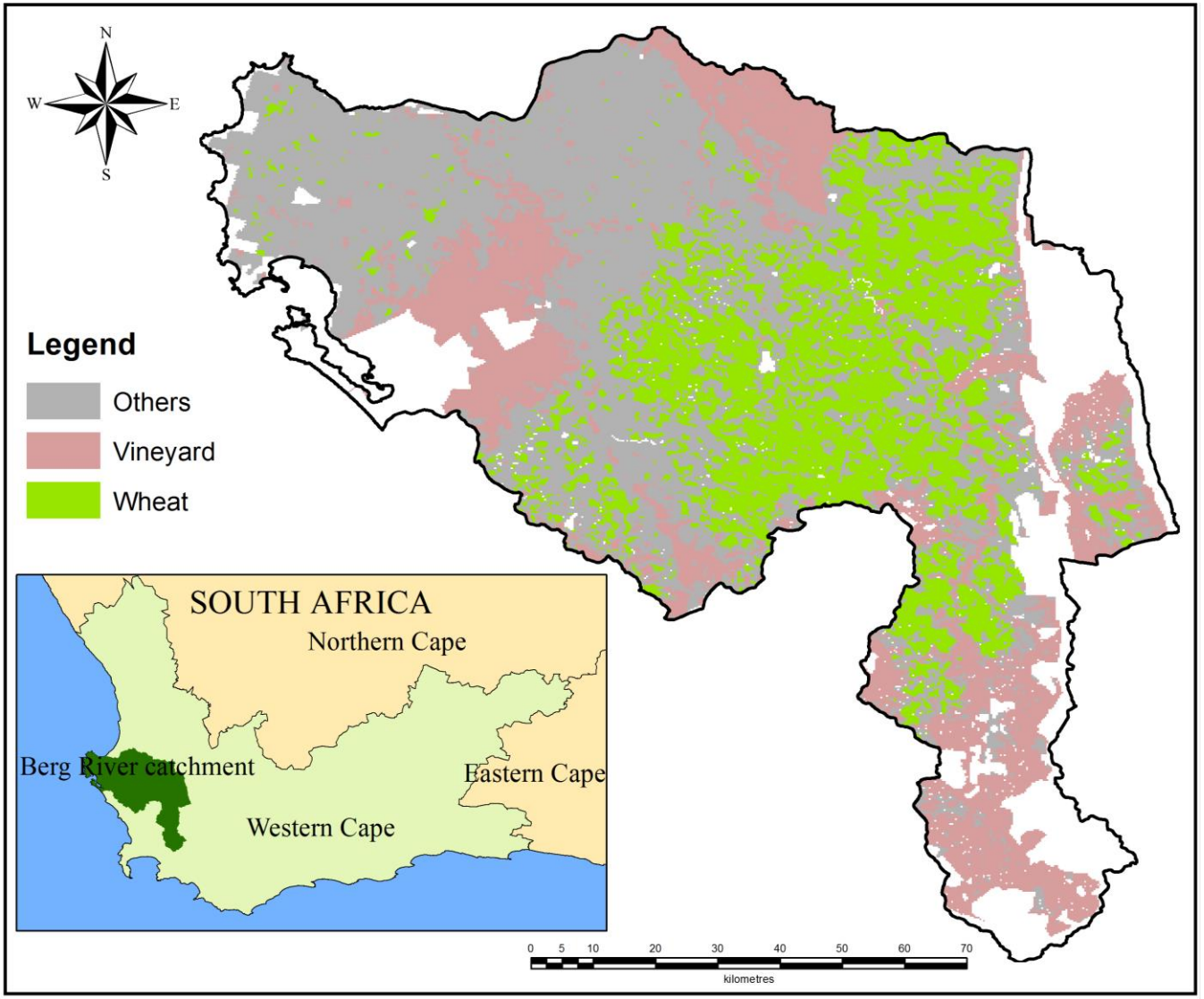


Figure A.2 MODIS land cover map for 2008

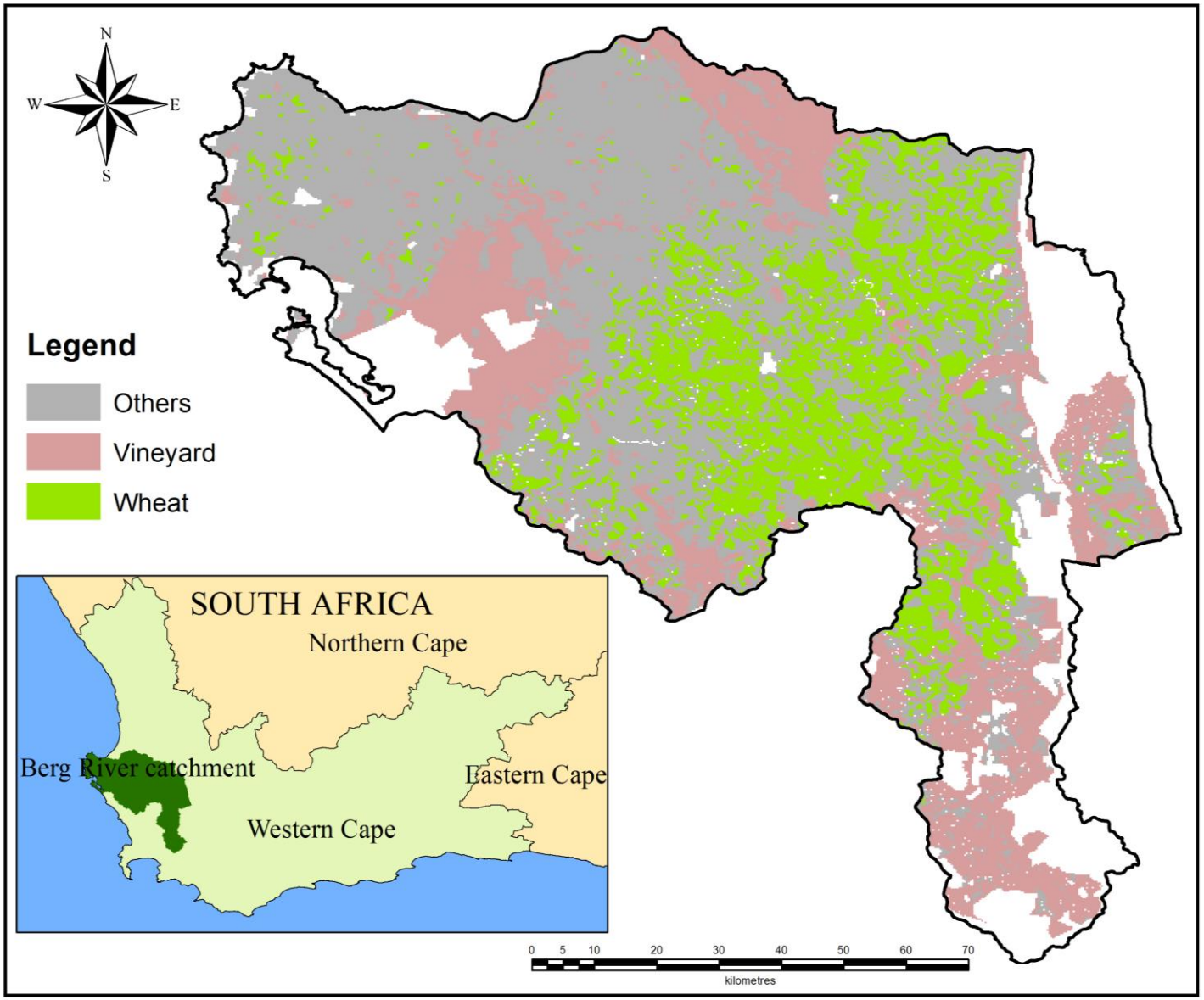


Figure A.3 MODIS land cover map for 2009

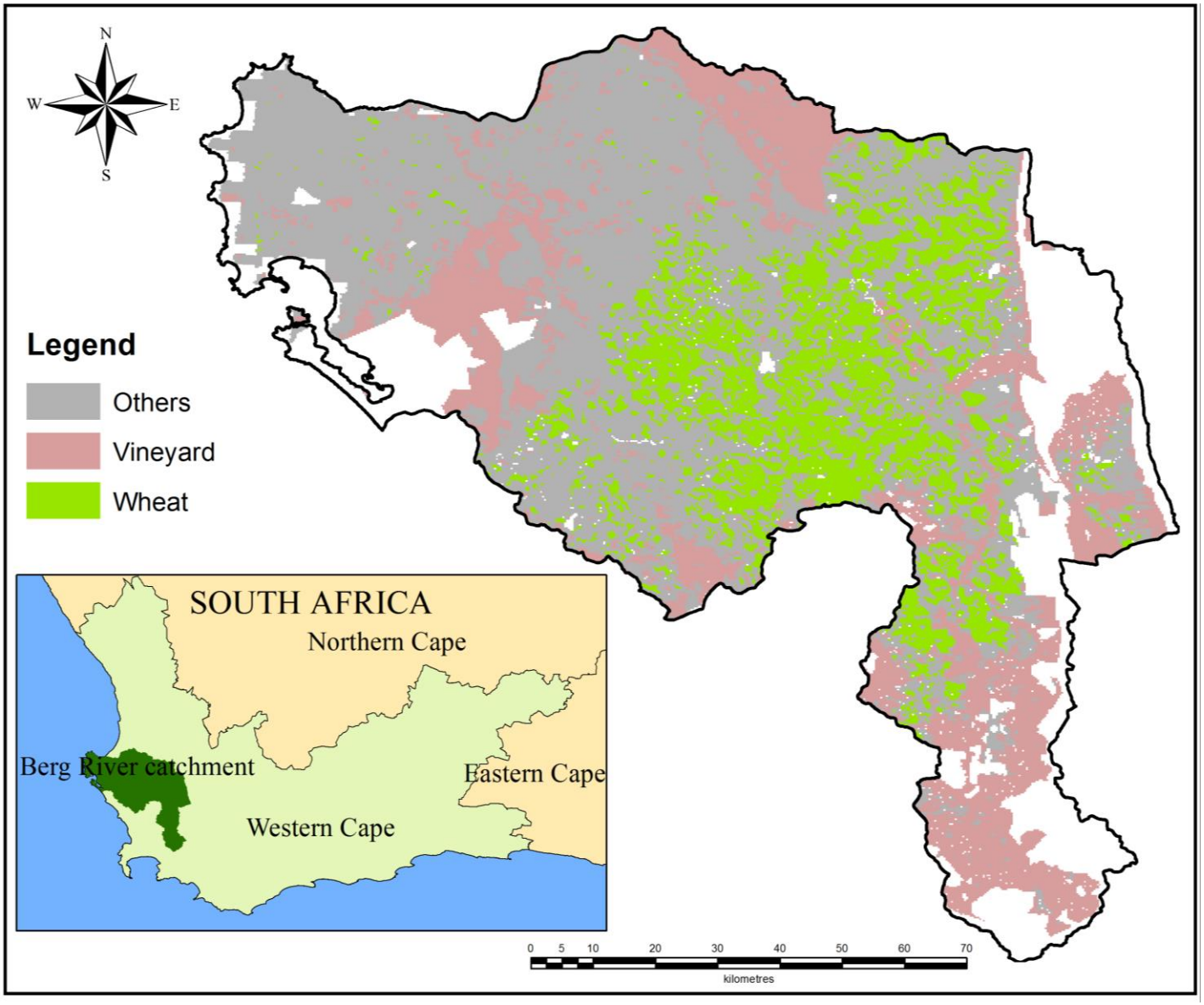


Figure A.4 MODIS land cover map for 2010

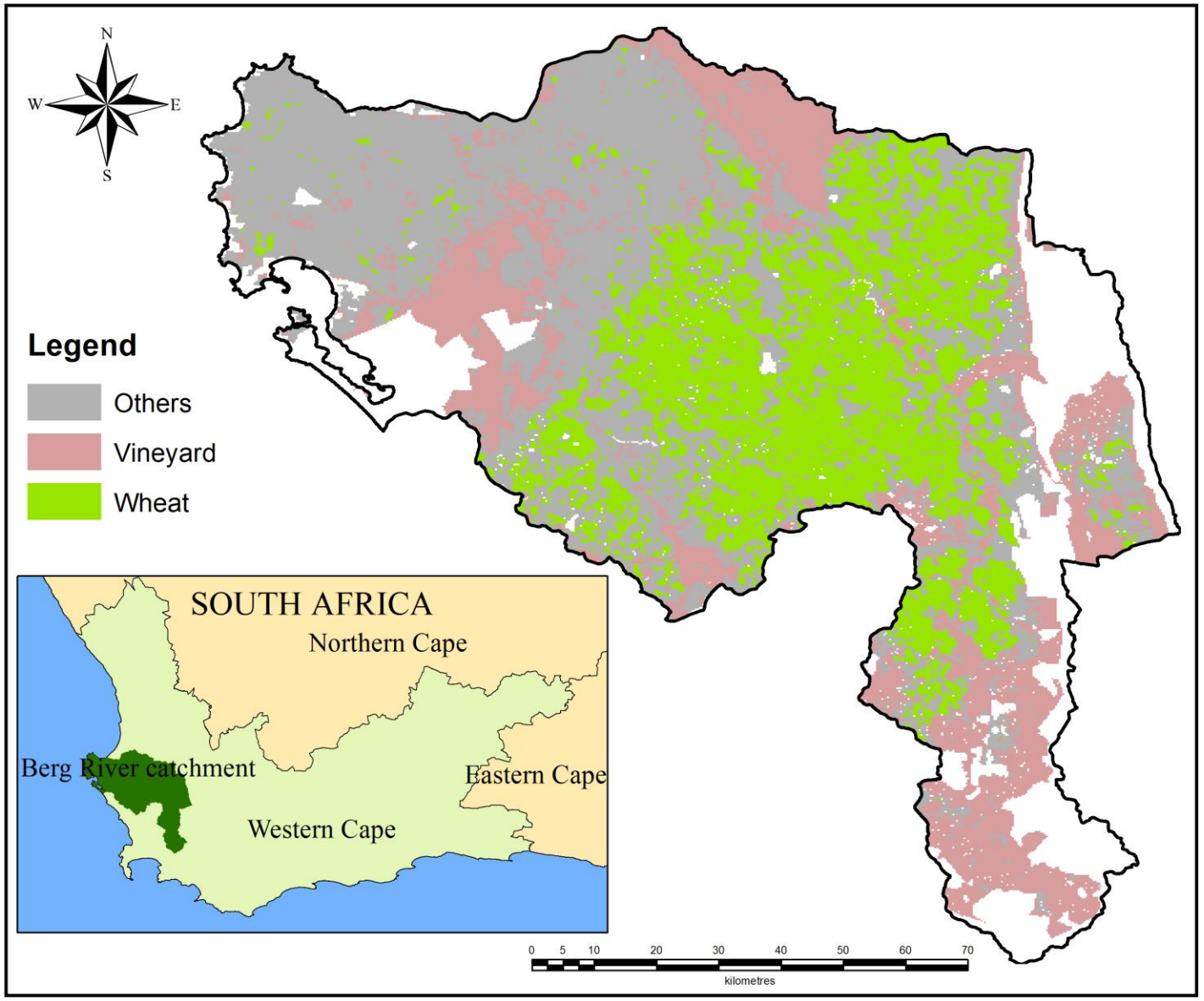


Figure A.5 MODIS land cover map for 2011

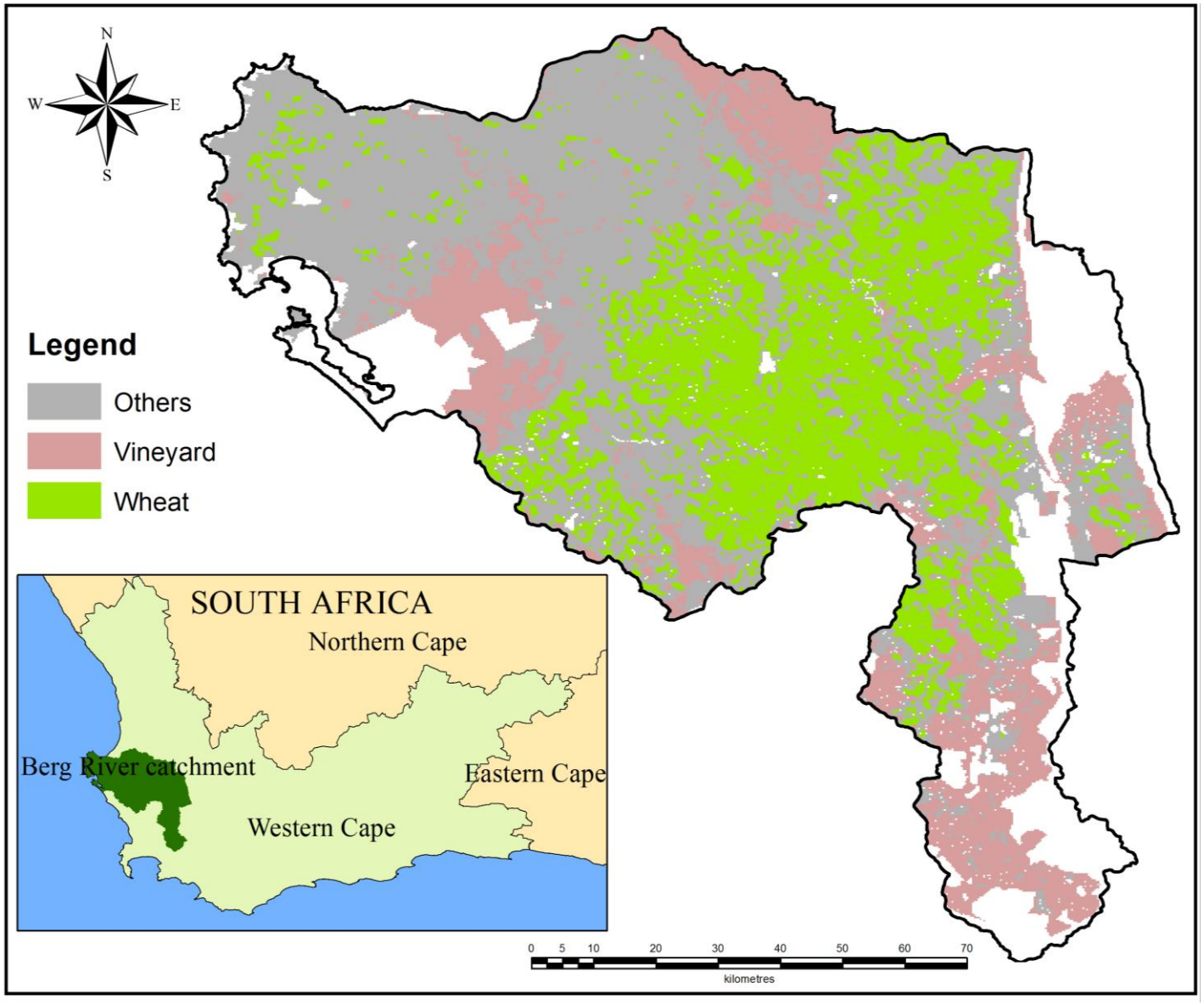


Figure A.6 MODIS land cover map for 2012

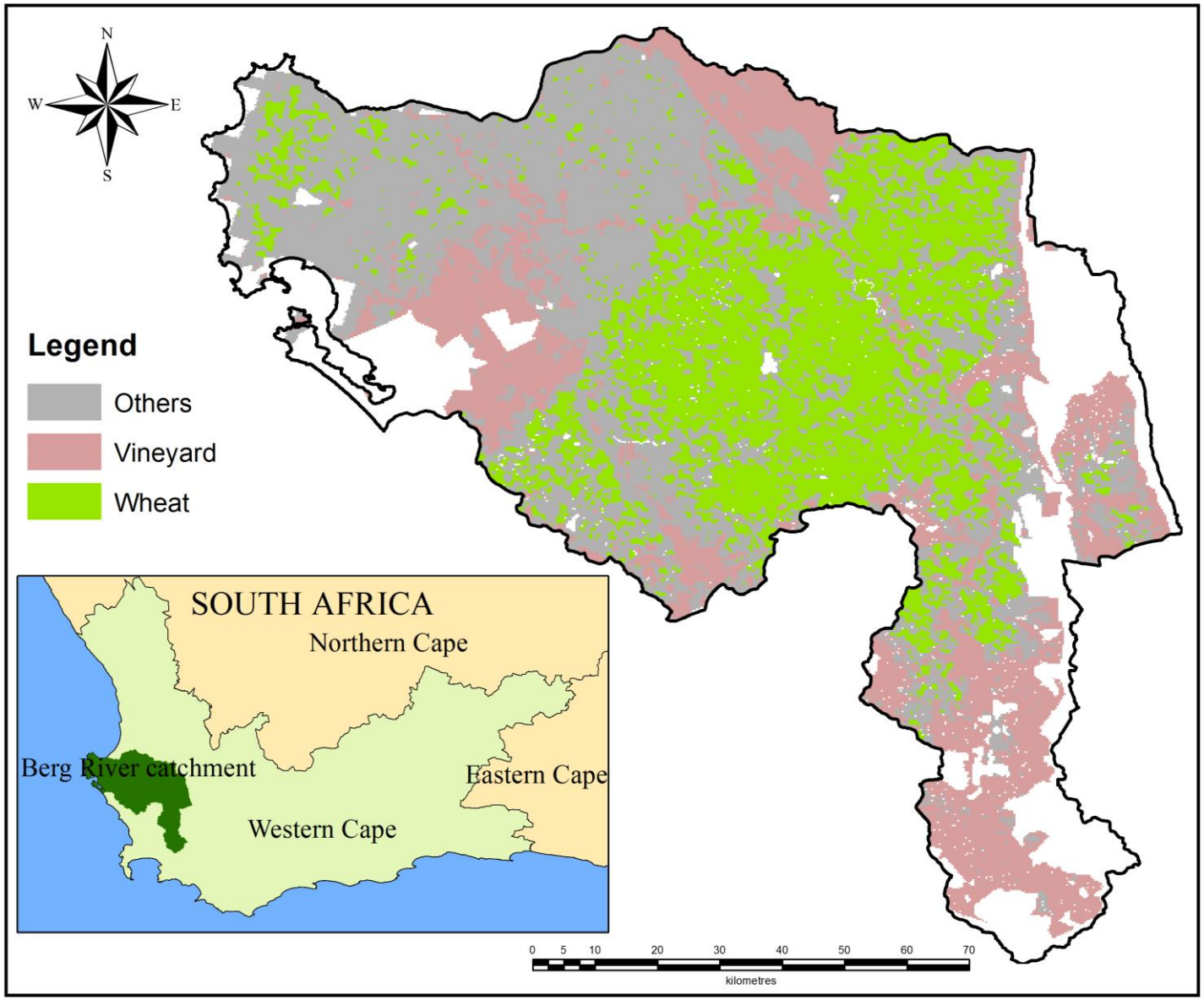


Figure A.7 MODIS land cover map for 2014

APPENDIX B: RANDOM FOREST CLASSIFICATION VISUAL TREES

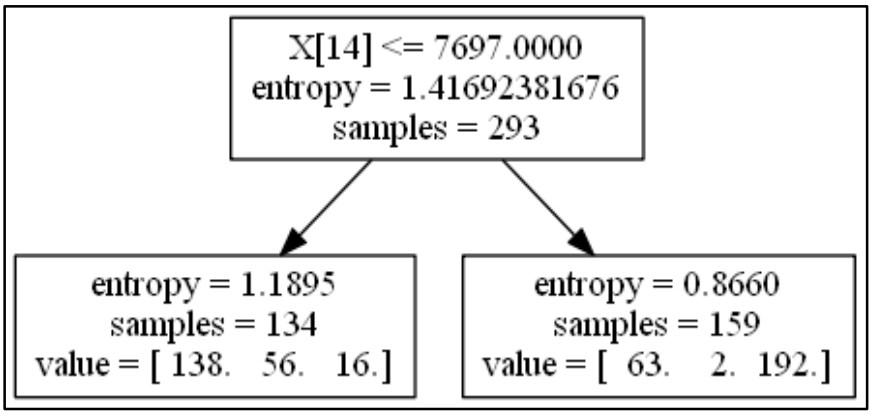


Figure B.1 Random Forest tree one

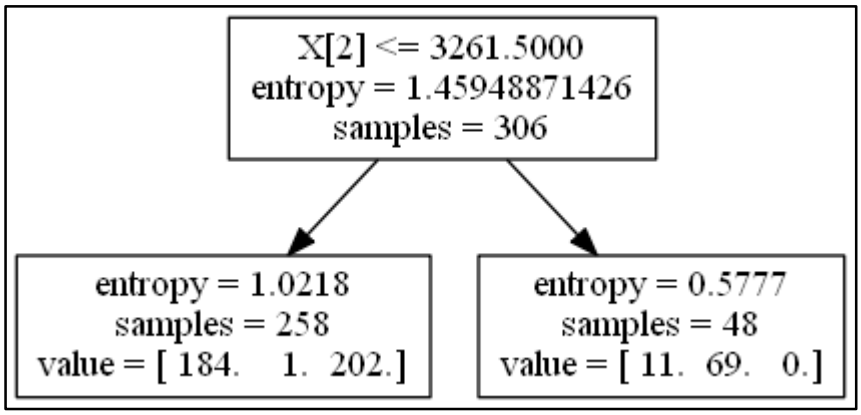


Figure B.2 Random Forest tree two

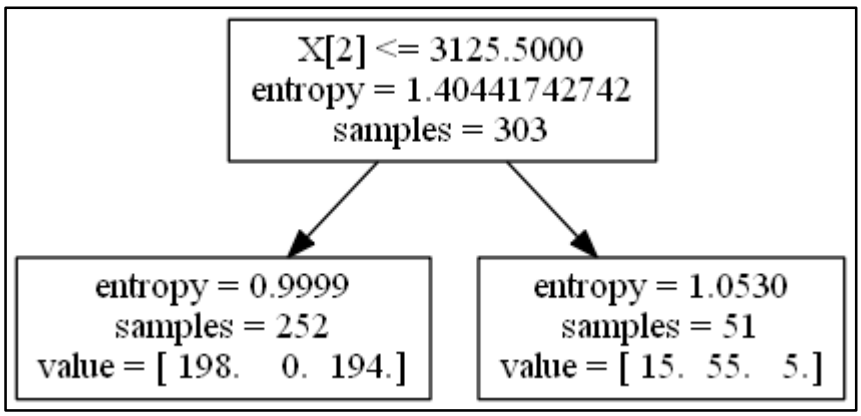


Figure B.3 Random Forest tree three

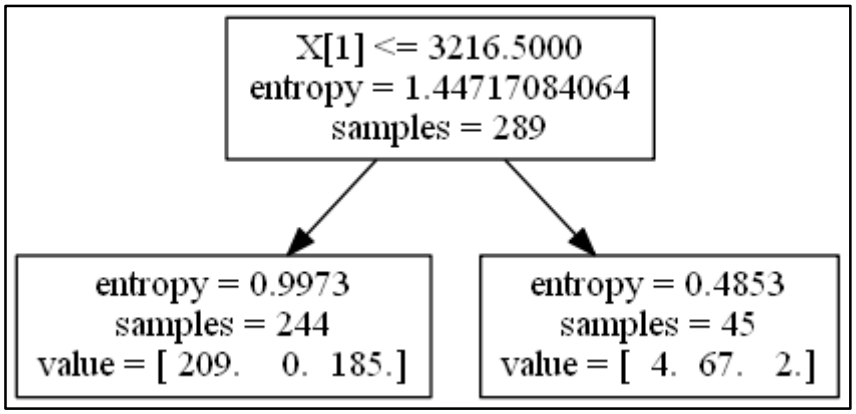


Figure B.4 Random Forest tree four

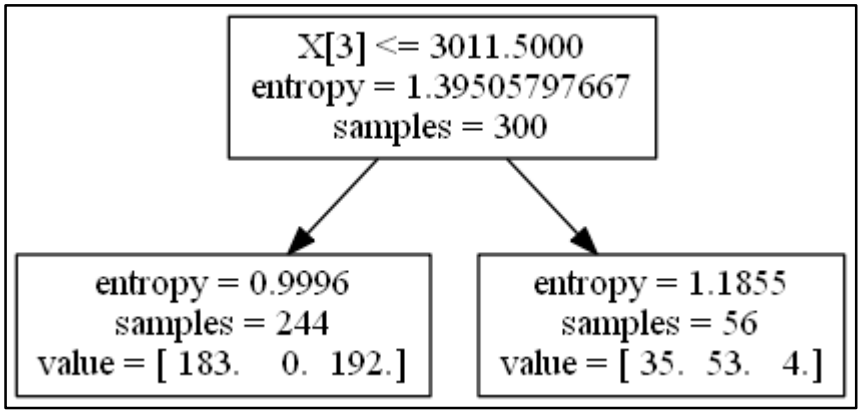


Figure B.5 Random Forest tree five

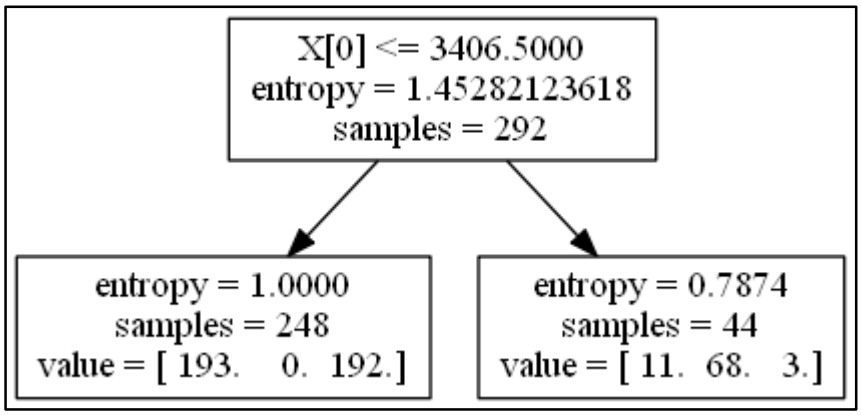


Figure B.6 Random Forest tree six

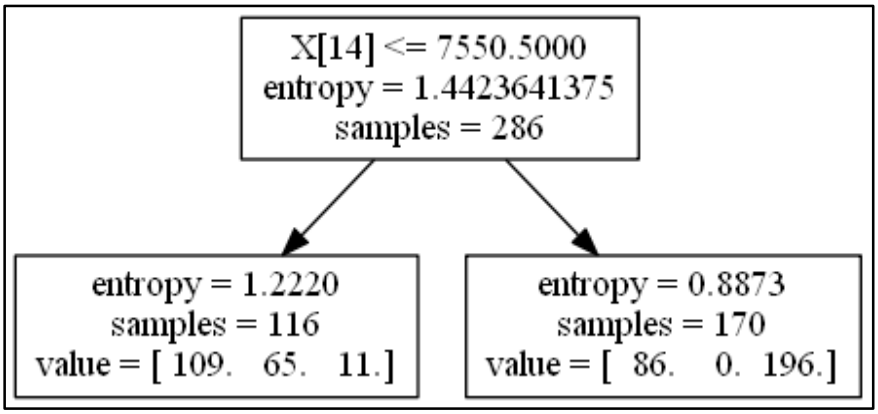


Figure B.7 Random Forest tree seven

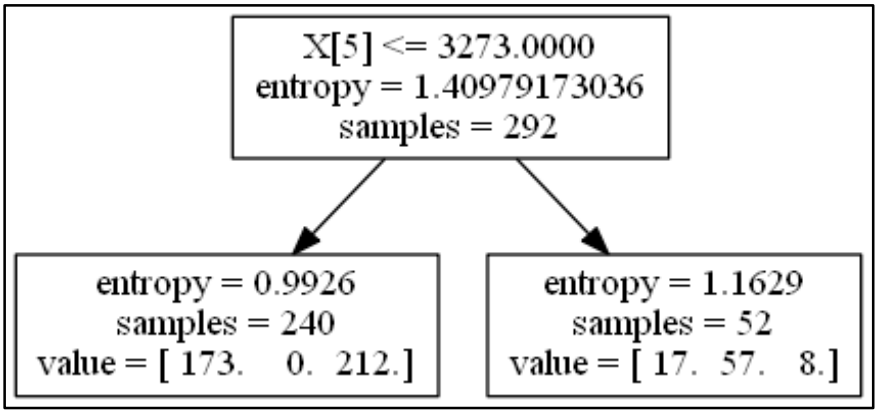


Figure B.8 Random Forest tree eight

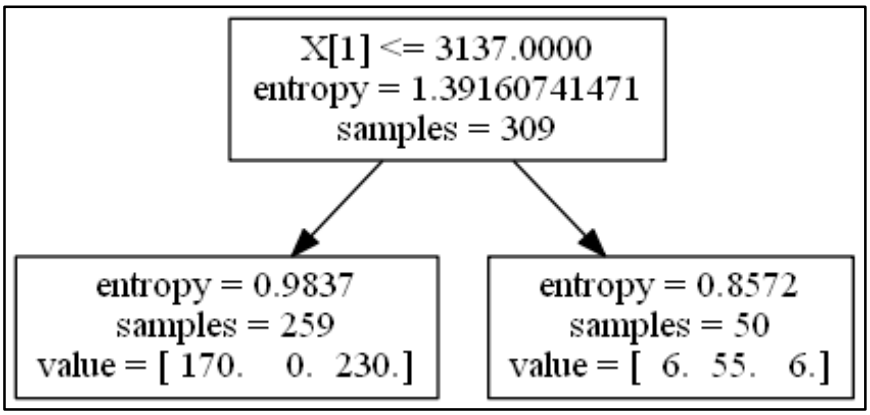


Figure B.9 Random Forest tree nine

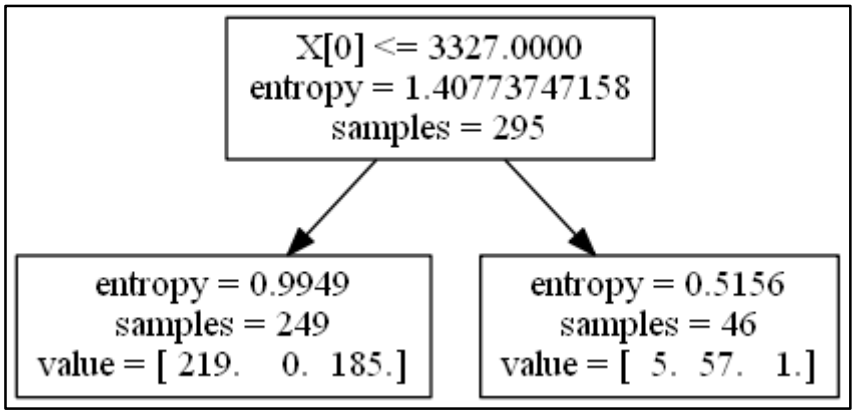


Figure B.10 Random Forest tree ten