Springer Geography

Courage Kamusoko

Remote Sensing Image Classification in R

Springer

Springer Geography

The Springer Geography series seeks to publish a broad portfolio of scientific books, aiming at researchers, students, and everyone interested in geographical research. The series includes peer-reviewed monographs, edited volumes, textbooks, and conference proceedings. It covers the major topics in geography and geographical sciences including, but not limited to; Economic Geography, Landscape and Urban Planning, Urban Geography, Physical Geography and Environmental Geography.

Springer Geography—now indexed in Scopus.

More information about this series at http://www.springer.com/series/10180

Courage Kamusoko

Remote Sensing Image Classification in R

Springer

Courage Kamusoko
Asia Air Survey Co., Ltd.
Kawasaki, Kanagawa, Japan

ISSN 2194-315X ISSN 2194-3168 (electronic)
Springer Geography
ISBN 978-981-13-8011-2 ISBN 978-981-13-8012-9 (eBook)
https://doi.org/10.1007/978-981-13-8012-9

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

To my girls Ai and Reina, who provide unending inspiration!!

Preface

This workbook is an introduction to remote sensing image processing and classification in R. The decision to write this workbook is based on my experience during my graduate and post-graduate studies, university teaching, and work as a consultant in remote sensing and modeling land use/cover changes. The aim of the workbook is to assist people who cannot afford commercial software and those who want to implement machine learning classifiers for remote sensing image processing and classification in R. There are many excellent introductory and advanced books on R, and remote sensing. However, there are few books that guide students, researchers, university teachers or other remote sensing practitioners for practical implementation of remote sensing in R. Therefore, this workbook is designed to be concise and practical in nature not as a complete guide to image processing and classification in R. That is, it is more of desktop reference workbook, which introduces R so that one can immediately start using the software platform and R packages for image processing and classification. Although R has an initial step learning curve, it is worth investing in R because it is free and has many packages, which might not be available in commercial software.

I also want to make it clear that while R is a good software or platform for graphics, statistical analysis, and machine learning given the number of packages available, it might not be appropriate for the analysis of massive remotely-sensed data. This is because the data being analyzed in R is held in memory. However, there many innovations in big data analytics such as parallel computing technologies in R or even cloud computing. Alternatively, other programming or scripting languages such as Python, Perl, and Ruby can be used depending on the nature of the problem and data availability.

Who should use this workbook?

The workbook is for undergraduate and graduate students in remote sensing and geographic information science or other related disciplines. While I assume no prior knowledge of R, the basic understanding of remote sensing is required. The workbook is also aimed at university teaching staff, researchers, or anyone interested in remote sensing image processing and classification. In addition, consultants or other people who are familiar with remote sensing but have limited experience in R can use this book to quickly test machine learning classifiers on small data sets.

How is this workbook organized?

This workbook is organized into five chapters. Chapter 1 introduces remote sensing digital image processing in R, which is subdivided into six sections. Section 1.1 presents a brief background on remote sensing image processing and classification, while Sect. 1.2 provides a brief overview of R and RStudio. Section 1.3 describes the data and test site, while Sect. 1.4 provides a quick guide to R. Finally, Sects. 1.5 and 1.6 provide the summary and additional exercises.

Chapter 2 covers pre-processing. Section 2.1 provides a brief background on pre-processing. Next, Sects. 2.2 and 2.3 provide tutorial exercises 1 and 2, which focus on displaying Landsat 5 Thematic Mapper (TM) imagery, and radiometric correction and reprojection. Finally, Sects. 2.4 and 2.5 provide the summary and additional exercises.

Chapter 3 focuses on image transformation. Section 3.1 provides a brief background on image transformation. Next, Sects. 3.2 and 3.3 focus on computing vegetation and texture indices. The final Sects. 3.4 and 3.5 provide the summary and additional exercises, respectively.

Chapter 4 covers image classification. Sect. 4.1 provides an overview of remote sensing image classification. Next, Sects. 4.2 and 4.3

focus on single date and multidate image classification, respectively. Finally, Sects. 4.4 and 4.5 provide the summary and additional exercises, respectively.

Last but not least, Chap. 5 focuses on improving image classification. Section 5.1 presents an overview of improving image classification, while Sect. 5.2 provides a brief background on feature selection. Next Sect. 5.3 focuses on image classification using multiple predictor variables, while Sect. 5.4 focuses on image classification using multiple predictor variables and feature selection. Finally, Sects. 5.5 and 5.6 provide the summary and additional exercises. An attempt has been made to organize the workbook in a general sequence of topics. Therefore, I encourage you to read the workbook in sequence from Chap. 1.

Conventions used in this workbook

The R commands or scripts are written in Lucida Console font size 10 in italics, while the output from the R is written in Lucida Console font size 10. Note that long output from R code is omitted from the workbook to save space. In some cases, I use small font sizes in Lucida Console to show how the R output or results would appear. This is just for illustration purposes. Readers will of course see the whole R output when they execute the commands. The hash sign (#) at the start of a line of code indicates that it is a comment. Finally, all explanations are written in New Times Roman font size 12.

Data sets, R scripts, and online resources

Information about the data sets and some sample scripts used in this workbook are available. Furthermore, I provide additional online resources on R, R packages or remote sensing in the appendix.

Kawasaki, Japan　　　　Courage Kamusoko

Acknowledgements

First and foremost, I would like to express my sincere gratitude to Professor Yuji Murayama, who has been inviting me to deliver a "Special Lecture on Geoenvironmental Sciences", with a focus on "Practical Machine Learning for Geospatial Processing and Modeling" at the Graduate School of Life and Environmental Sciences, University of Tsukuba in Japan. This workbook would not have been possible without his support. Second, I would like to thank students from the Graduate School of Life and Environmental Sciences in general, and students from the Spatial Information Science Laboratory at the University of Tsukuba in particular who have taken this lecture for years. I am truly indebted to the students because I learnt a lot about machine learning and R during the lectures.

I am also indebted to my colleague, Ms. Emiko Ariyasu from Asia Air Survey Co., Ltd. who read the first draft of the workbook. Her comments provided invaluable insights during the revision. I am also truly grateful to the Global Business Development Division management at Asia Air Survey Co., Ltd. who gave me permission to teach part-time at the University of Tsukuba. I extend my special thanks to emeritus professor Masamu Aniya, my academic advisor who provided critical support during my graduate and post-graduate days at the University of Tsukuba. At Springer (Japan), I would like to thank Yosuka Nishida, Asami Komada, and Arun Kumar, and anonymous reviewers.

Last but not least, this workbook would not have been possible without the continued support of my wife Yuko, and my girls Ai and Reina.

Contents

Abbreviations and Acronyms

ANN	Artificial Neural Network
ART	Adaptive Resonance Theory
ASM	Angular Second Moment
COST	Cosine Estimation of Atmospheric Transmittance
CRAN	Comprehensive R Archive Network
DEM	Digital Elevation Model
DOS	Dark Object Subtraction
DN	Digital Number
DSG	Department of the Surveyor-General
DT	Decision Trees
EMR	Electromagnetic Radiation
FOSS	Free and Open Source Software
GDAL	Geospatial Data Abstraction Library
GIS	Geographic Information Systems
GLCM	Grey Level Co-occurrence Matrix
GUI	Graphical User Interface
IDE	Integrated Development Environment
KNN	k-Nearest Neighbors
MIR	Mid Infrared
MODTRAN	Moderate Resolution Atmospheric Radiance and Transmittance
ML	Machine Learning
MPL	Multi-Layer Perceptron
MSAVI	Modified Soil Adjusted Vegetation Index
NDVI	Normalized Difference Vegetation Index
NIR	Near Infrared

RF	Random Forests
RFE	Recursive Feature Elimination
SAVI	Soil Adjusted Vegetation Index
SDOS	Simple Dark Object Subtraction
SOM	Self-Organizing Feature Maps
SR	Spatial Resolution
SVM	Support Vector Machines
TIR	Thermal Infrared
TM	Thematic Mapper
UAV	Unmanned Aerial Vehicles
USGS	United States Geological Survey

Chapter 1
Remote Sensing Digital Image Processing in R

Abstract Remote sensing digital image processing and classification provide critical land use/cover and land use/cover change information at multiple spatial and temporal scales. Over the past decades, a plethora of image processing and classification methods have been developed and applied. The purpose of this chapter is to introduce remote sensing digital image processing and machine learning in R. The chapter will cover remote sensing image processing and classification, a brief overview on R and RStudio, tutorial exercises, data and test site.

Keywords Remote sensing · Digital image processing · Machine learning · R · RStudio

1.1 Introduction

1.1.1 Remote Sensing Digital Image Processing

Earth observing remote sensing is the science and art of acquiring information about the Earth's surface using noncontact sensor systems. Sensors on board satellites, aircrafts or unmanned aerial vehicles (UAV) record reflected or emitted electromagnetic radiation (EMR) from features on the Earth's surface. The reflected or emitted electromagnetic radiation (EMR) from features on the Earth's surface is captured as a pixel, which is then converted to a digital number (DN) in order to produce remotely-sensed image. Computers and

C. Kamusoko, *Remote Sensing Image Classification in R*, Springer Geography,
https://doi.org/10.1007/978-981-13-8012-9_1

software are required for digital image processing since the remotely-sensed images are digital. Digital image processing procedures such as pre-processing, image transformation, and image classification will be covered in this workbook. While there many commercial, and free and open source software systems (FOSS) for remote sensing image processing and analysis, R will be used in this workbook.

Remote sensing digital image processing and analysis provide valuable information on land use/cover and its changes, agriculture, environment and other applications at multiple spatial and temporal scales. A variety of methods have been developed and used for land use/cover classification. These classification methods include: the incorporation of structural and textural information (Gong and Howarth 1990; Moller-Jensen 1990); combining satellite images with ancillary data (Harris and Ventura 1995); vegetation—impervious surface—soil models (Ridd 1995); expert systems (Stefanov et al. 2001); hybrid methods that incorporates soft and hard classifications (Lo and Choi 2004); the use of normalized difference built-up index (Xu 2007; Zha et al. 2003); neural networks and deep learning (Seto and Liu 2003; Yu et al. 2017); segmentation and object-based classifications (Guindon et al. 2004); support vector machines (Nemmour and Chibani 2006; Pal and Mather 2003) and random forests (Rodriguez-Galiano et al. 2012).

To date, many advanced classification methods (or classifiers)—in particular machine learning methods—have been used to improve remotely-sensed image classification. However, there are still major challenges such uncertainties in remotely-sensed data, lack of reliable or insufficient reference data sets, and generally a "blind" trust in advanced classification methods (Lu and Weng 2007). Although remote sensing literature is replete with successful applications (Lu and Weng 2007), land use/cover classifications based on medium resolution satellite imagery such as Landsat data (as we will see in our study area) pose many methodological challenges (Griffiths et al. 2010). This is because the urban study area is characterized by a complex and contrasting spatial and socioeconomic development patterns. For example, similar spectral responses between built-up areas on one hand, and bare vacant plots and agriculture areas on the other hand have been observed to cause classification errors.

Furthermore, forest cover classification—which is major input in forestry and climate change projects—is riddled with many uncertainties. For example, sparse forest or woodland areas exhibit high degrees of spatial heterogeneity, which is influenced by soil type, land use, and seasonal changes (Sedano et al. 2005). As a result, remotely-sensed image classification should not be limited to a mere technical approach based on "fancy" or latest machine learning methods. Rather, a more integrated approach that highlights uncertainties in data acquisition (satellite imagery and training data), pre-processing, image classification and accuracy assessment is required. Recent developments such as the use of dense satellite imagery due to the availability of more computer processing power or cloud computing, and more awareness on the need to improve accuracy assessment are positive.

In this workbook, machine learning methods or classifiers such as k-nearest neighbors (KNN), single decision trees (DT), artificial neural networks (ANN), support vector machines (SVM) and random forest (RF) will be used to classify Landsat 5 TM imagery. I have selected these machine learning classfiers since most of them have been widely used for remotely-sensed image classification over the past decades. Note that the idea is not to find the "best method" but to provide hands-on approach to remotely-sensed image processing based on remote sensing and machine learning principles. An attempt is made to focus on common problems related to pre-processing, analysis of data sets prior to image processing and classification, machine learning model tuning and performance (cross-validation), and accuracy assessment.

1.1.2 Overview of Machine Learning

During the past decades, the knowledge domain of machine learning has grown rapidly given the developments in computer technology coupled with advances in information science, statistics and data mining techniques. In addition, the availability of more data from the internet, mobile technology and social media has also contributed to the advancement of machine learning. To date, there are many machine learning classifiers, and applications in diverse fields such as banking and finance, health and medicine, image processing, and environment. While there are many definitions of machine learning, in this workbook we will simply define machine learning as computational methods that learn (or improve their performance) from data (Hastie et al. 2009; Dietterich 1999). According to Mitchell (1997), "a computer program can learn from experience *E* with respect to some class of tasks *T*, and performance measure *P*". Therefore, machine learning algorithms use experience in order to improve performance. Experience refers to data-driven tasks, which are based on statistics and probability (Mitchell 1997).

Machine learning combines statistics, computer science and data mining to solve classification, clustering, regression and other pattern recognition problems (Michie et al. 1994; Ripley 1996; Hastie et al. 2009, Cracknell and Reading 2014). It should be noted from the onset that computer-based statistical approaches are a key component of machine learning (Hastie et al. 2009). However, machine learning approaches differ from conventional statistical approaches. This is because statistical approaches first assume an appropriate data model for fitting, and then model parameters are estimated from the data. In other words, it assumed that the basic form of the model equation and error function is known and therefore, the goal is to find the coefficients of variables in a known function (Hastie et al. 2009). In this regard, the emphasis is on understanding the underlying statistical model and mechanisms as well as model assumptions (Clark 2013). In contrast, machine learning uses an algorithm to learn the relationship between the response (target or dependent) variable and its predictor (independent) variables, particularly in complex data sets (Breiman 2001). As such, the focus of machine learning is more on performance rather than understanding the underlying statistical model mechanisms and assumptions (Hastie et al. 2009; Clark 2013).

Machine learning performs inductive data inference or learning (MacKay 2003; Gahegan 2000), which simply refers to gaining information, knowledge and wisdom from the analysis of raw data (Fotheringham et al. 2002). Inductive inference or learning uses available data or observations to identify patterns. That is, generalizations are derived from specific training data (Bousquet et al. 2004). Essentially, the results obtained from the inference or learning analyses are then applied to other similar data in order to predict, interpret and inform the decision making process (Bousquet et al. 2004). This is in contrast to deductive inference, where a hypothesis (or model) regarding some natural phenomenon is formulated (Gubbins 2004). The observed data are then used to accept or reject the hypothesis.

Machine learning techniques can be classified according to many different criteria, which depend on the nature of the problem that needs to be solved (Russell and Norvig 2003). In this workbook, we will classify the machine learning techniques according to the task or learning style, and the desired output. In terms of task or learning style, machine learning techniques can be grouped into supervised, unsupervised, semi-supervised and reinforcement learning. For supervised learning, example input and output data known as training data are provided. The goal is to predict the outcome based on the input data (Hastie et al. 2009). Training is guided by the minimization of some error based on the internal structure of the learning algorithm (Bousquet et al. 2004; Hastie et al. 2009). For unsupervised learning, the algorithm finds natural groups in data without any labels (Ripley 1996; Hastie et al. 2009). Therefore, the algorithm discovers patterns in data based on clustering. Semi-supervised learning incorporates incomplete training data. That is, both labeled and unlabeled training data are used (Han et al. 2012). The training data are usually composed of a small labeled data set and a large unlabeled data set. Generally, the labeled training data are used to learn, while the unlabeled training data is used to refine the class boundaries (Han et al. 2012). In reinforcement learning, the algorithm interacts in a dynamic environment, whereby learning is based on external positive and negative feedbacks (Portugal et al. 2018). Through trial and error, the algorithm learns to reward positive feedbacks, and avoid negative feedbacks (Portugal et al. 2018).

According to the desired output, machine learning can be categorized as classification, regression, clustering, and density estimation. In classification, inputs are divided into two or more classes, and the learner must produce a model that assigns unseen inputs to one or more of these classes, while in regression the outputs are continuous rather than discrete. Both classification and regression can be supervised. Clustering is an unsupervised task where a set of unknown or unlabeled inputs are divided into groups, while density estimation finds the distribution of inputs in some space.

1.2 Overview of R

1.2.1 What Is R?

R is a free and open source integrated software platform for programming, data manipulation, computation and graphical display (R Development Core Team 2005). According to Maindonald and Braun (2003), R is a dialect of the S language (that was developed at AT&T Bell Laboratories by Rick Becker, John Chambers and Allan Wilks). This script-based software with command lines was developed by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand. R was released under the GNU GPL license in 1995, and it has since evolved. Currently, R is now being maintained by a group of developers, and is available on multiple computing platforms such as Windows, GNU/Linux, and Mac OS X. All the R scripts and commands used in this workbook were executed on the Windows platform. However, R scripts and commands can also be adapted for other platforms.

1.2.2 Installing R

In order to install R, go to the R project download link at https://www.r-project.org/ (Fig. 1.1) and click on CRAN (Comprehensive R Archive Network). This website will direct you to a list of international mirror sites from which you can download R (Fig. 1.2). Note that the CRAN mirror sites are organized by country. Therefore it is better to use the mirror site that is closer geographically in order to download R faster. For example, my mirror site is “The Institute of Statistical Mathematics” in Tokyo (https://cran.ism.ac.jp/) because I live near Tokyo. Select a mirror and click the link. A list of R downloads links for Linux, Mac (OS X) and Windows are available (Fig. 1.3). For Windows, you can select **base**, which has binaries for the base distribution if you are installing R for the first time. After that, just double-click the installer “Download R 3.4.3 for Windows (62 megabytes, 32/64 bit)” and follow the instructions. It is easy to install R on Windows, if you have administrative permission. On Windows, R is installed as a graphical user interface