

Using a Coupled Radiative Transfer Model and Artificial Neural Networks to Retrieve Forest Biochemical and Biophysical Maps from Remote Sensing Data



 **Trier University**

Author:

Zavud Baghirov (1399020)

Supervisors:

Prof. Dr. Thomas Udelhoven
Dr. Martin Schlerf

Department of Environmental Remote Sensing and Geoinformatics

University of Trier

Submitted in partial fulfillment of the requirements for the degree of
Master of Science in

Environmental Science

November 2021

Abstract

Over the last decades, forest ecosystems have been increasingly disturbed by a variety of disturbances. One of the major disturbances to the forests have been reported to be insect infestations and drought. As a result, retrieving forest biophysical and biochemical maps from remote sensing data could aid in the study of forest disturbances. In this study, the radiative transfer models PROSPECT5, 4SAIL and FLIM were coupled in order to simulate forest canopy reflectance for the National Park Hunsruck-Hochwald. The simulated data was used to train Artificial Neural Networks (ANN), and the final trained model was used to predict forest biophysical and biochemical parameters using the PRISMA hyperspectral remote sensing image of the study area. Two ANN models were trained, one with the dimensionality reduction technique, PCA and one with simulated reflectance for PRISMA data bands. In general, both trained ANN models performed well. However, it was determined that the ANN that utilized all the available bands was superior in terms of loss and prediction errors for each plant parameter. Specifically, when the dimensionality reduction technique was not used, prediction of the parameter LAI_s resulted in noticeably higher accuracy. The optimal neural network model was chosen and used for the final predictions after being trained with the 231 simulated PRISMA bands. On the testing data set, this model had an MSE of 0.077 and an MAE of 0.13. The plant parameter C_{ab} was predicted with an R^2 of 0.89 and a root mean square error (RMSE) of 4.064 ($\frac{\mu g}{cm^2}$). C_w was predicted with an R^2 of 0.94 and an RMSE of 0.003 ($\frac{g}{cm^2}$). R^2 and RMSE values for C_m were 0.95 and 0.002 ($\frac{g}{cm^2}$) respectively. Finally, R^2 and RMSE values for LAI_s were 0.95 and 0.44 ($\frac{m^2}{m^2}$), respectively.

Contents

List of Figures	vi
List of Tables	vii
List of Abbreviations	viii
1 Introduction	1
2 Forest Disturbances by Bark Beetle	4
3 Radiative Transfer Models (RTM)	7
3.1 Leaf Radiative Transfer Model - PROSPECT	8
3.2 Scattering by Arbitrary Inclined Leaves (SAIL)	9
3.3 Forest Light Interaction Model (FLIM)	11
3.4 Invertable Forest Reflectance Model (INFORM)	12
3.5 Challenges of RTMs	14
4 Machine Learning Regression Algorithms (MLRA)	15
4.1 Decision Trees (DT)	15
4.2 Kernel-Based Machine Learning Regression Methods (KBMLRM) .	16
4.3 Artificial Neural Networks (ANN)	18
4.4 Challenges of Machine Learning Regression Algorithms (MLRA) . .	20
5 Hybrid Machine Learning Method	22
6 Satellite Data	24
7 Study area	25
8 Methods	27
8.1 Local sensitivity analysis	27
8.2 RTM simulation (INFORM)	29
8.3 Spectral resampling	31
8.4 Statistics of simulated data and PRISMA image	31

8.5	Gaussian noise	31
8.6	Defining training, validation and testing sets	32
8.7	Data processing	32
8.8	Principal Component Analysis (PCA)	33
8.9	Artificial Neural Networks (ANN)	33
8.9.1	Cost function	35
8.9.2	Optimizer algorithm	35
8.9.3	Mini batch size	37
8.9.4	Regularization	37
8.9.5	Early stopping	38
8.9.6	Activation function	39
8.9.7	Weight initialization	39
8.9.8	Processing of input and target variables	40
8.9.9	Number of epochs	40
8.9.10	Neural Network Architecture design	41
8.10	Final prediction and map retrieval	43
8.11	GitHub repository of the study	43
9	Results	44
9.1	Local sensitivity analysis	44
9.2	RTM simulation (INFORM)	46
9.3	Spectral resampling	46
9.4	Statistics of simulated data and PRISMA image	47
9.5	Gaussian noise	49
9.6	Principal Component Analysis (PCA)	50
9.7	Artificial Neural Networks (ANN)	51
9.7.1	Training	51
9.7.2	Evaluation of the NN models on the testing set	53
9.8	Final prediction and map retrieval	56
9.8.1	Retrieval of C_{ab} map	56
9.8.2	Retrieval of C_w map	58
9.8.3	Retrieval of C_m map	59
9.8.4	Retrieval of LAI_s map	60

Contents

10 Discussion	61
10.1 Artificial Neural Networks for predicting plant traits	61
10.2 Limitations	62
10.3 Future perspectives	63
11 Conclusion	65
References	66

List of Figures

3.1	Diagram of PROSAIL as a combination of PROSPECT5 and 4SAIL in the forward mode	11
3.2	Visualization of the INFORM model as a combination of PROSPECT5, 4SAIL and FLIM in the forward mode	13
7.1	The study area on Germany's map.	25
7.2	The study area.	26
8.1	Architecture of the Neural Network with Principal Components as inputs	41
8.2	Architecture of the Neural Network with simulated PRISMA image bands as inputs	42
9.1	Effects of varying the chosen parameters on the simulated spectra .	45
9.2	Mean and mean \pm standard deviation in the a) LUT and b) PRISMA image	47
9.3	Difference between averaged LUT and PRISMA image reflectance .	48
9.4	Effect of adding 3% Gaussian noise to the simulated spectra. The randomly chosen pixel from the PRISMA data was plotted to illustrate the noise typically found in the image	49
9.5	Principal Component Analysis: a) Screeplot, b) Cumulative variance explained by the first 5 PCs	50
9.6	Training history of ANN with PCs	51
9.7	Training history of ANN with simulated PRISMA bands	52
9.8	Predicted versus modelled RTM parameters for the PCA based NN (1st column) and NN with simulated PRISMA bands (2nd column)	54
9.9	Predicted map of the parameter C_{ab}	56
9.10	Predicted map of the parameter C_w	58
9.11	Predicted map of the parameter C_m	59
9.12	Predicted map of the parameter LAI_s	60

List of Tables

3.1	Input parameters of PROSPECT5	8
3.2	Input parameters of 4SAIL	10
3.3	Input parameters of FLIM	11
8.1	INFORM parameters varied in local sensitivity analysis (each parameter were varied 15 times)	27
8.2	INFORM Parameters that were kept constant while one parameter was varied at a time	28
8.3	Range of full input parameters that were used to create a LUT size of 316800	30
9.1	Final results of the trained NNs	53
9.2	Differences between the performances of the trained 2 NN models on the testing set parameters	55

List of Abbreviations

3D	Three-dimensional
AL	Active Learning
ANN	Artificial Neural Networks
DNN	Deep Neural Networks
DR	Dimensionality Reduction
DT	Decision Trees
FLIM	Forest Light Interaction Model
FOV	Field of View
GP	Gaussian Process
GPR	Gaussian Process Regression
GSD	Ground Sampling Distance
INFORM	Invertable Forest Reflectance Model
KBMLRM	Kernel-Based Machine Learning Regression Methods
LAI	Leaf Area Index
LUT	Look-Up-Table
MAE	Mean Absolute Error
ML	Machine Learning
MLRA	Machine Learning Regression Algorithms
MSE	Mean Squared Error
NIR	Near Infrared
NN	Neural Networks
PC	Principal Component
PCA	Principal Component Analysis
PRISMA	PRecursore IperSpettrale della Missione Applicativa
PROSAIL	The combination of PROSPECT and SAIL models
ReLU	Rectified Linear Activation Unit
RF	Random Forest

List of Abbreviations

RFR	Random Forest Regression
RTM	Radiative Transfer Model
SAIL	Scattering by Arbitrary Inclined Leaves
SGD	Stochastic Gradient Descent
SVM	Support Vector Machines
SVR	Support Vector Regression
SWIR	Short Wave Infrared
VI	Vegetation Index
VNIR	Visible and Near Infrared
WT	Wavelet Tranform

1 Introduction

Forests cover approximately 30% of the earth's surface and play a critical role in the water and carbon cycles (Laurent et al., 2011). As forest environments are being threatened more and more, there is an increased concern for forest conservation (Grammatikopoulou and Vačkářová, 2021). The effects of climate change on many different ecosystems have been studied and shown. Forest ecosystems are particularly more sensitive to climate change compared to other ecosystems (Esquivel-Muelbert et al., 2019; Lindner et al., 2010). According to Lindner et al. (2010), the primary reason forest ecosystems are particularly susceptible to extraordinary climate changes is because, in comparison to other organisms, trees typically live longer and have fewer adaptive abilities to cope with rapid environmental changes.

Another threat to the forest that has been increasingly reported during the last century is insect disturbances. Insect disturbances have increased the concern about forest species diversity, biogeochemical cycle and the economic value of forest ecosystems (Senf et al., 2017). The main forest insects that are responsible for many outbreaks can be divided into bark beetles and defoliators. Many bark beetle species can reproduce their offspring within both dead and living tree organisms' phloem. These bark beetles can easily destroy the structure of translocation processes (e.g. water and nutrient translocations) by introducing various pathogens (e.g. fungal) into healthy trees (Senf et al., 2017).

A study by Kautz et al. (2017) has reported that there is only a limited amount of consistent data for studying forest disturbance caused by insects (Senf et al., 2017). Remote sensing images provide very interesting insights into studying environmental disturbances and changes within forest areas. According to the study by Lechner et al. (2020), remote sensing techniques have recently been actively used for forest studies. For example, remote sensing images are offering important opportunities to

map ecosystem distribution, to study forest plant productivity and to get insights about the 3D structural properties of forest plants (Lechner et al., 2020).

One of the most important insights remote sensing provides is the fact that the image of the earth’s surface can be acquired over time. This gives a unique opportunity to be able to study environmental changes in a specific area of interest over the time period the images are acquired. For example, there have been studies where remote sensing techniques were used to map the distribution of invasive species and to monitor land changes (e.g. forest habitat fragmentation) in forests (Lechner et al., 2020).

Another interesting opportunity that remote sensing data provides is to estimate vegetation biophysical and biochemical properties from remotely sensed images. Essentially, there are various ways to retrieve plant biophysical parameters from remote sensing data. Each method has its own advantages and disadvantages. The radiative transfer model (RTM) has been accepted to be the superior method to all the available techniques (Darvishzadeh et al., 2019b; Wang et al., 2018b). Various RTMs have been proposed at different scales, such as leaf and canopy scales. Different RTMs such as leaf and canopy can be used in a coupled mode in order to simulate canopy reflectance based on vegetation biophysical and biochemical properties (Morcillo-Pallarés et al., 2019).

There are mainly three inversion methods to estimate plant traits from remote sensing data. These methods are numerical optimization, look-up-table and hybrid techniques. Numerical optimization is considered to be a classical method. However, the major drawbacks of this method are its computational inefficiency and the high risk of poor convergence (Zhang et al., 2021). The LUT method refers to a technique where databases to store simulated data are created by RTMs. And then, plant biophysical variables of the simulated reflectance that represent the measured data most closely are chosen as retrievals. Although the LUT method is more efficient than the classic numerical optimization, it is still a computationally inefficient process given that inversion needs to be done for each pixel in the remote sensing product (Danner et al., 2021). The hybrid method can be used to overcome

1. Introduction

the challenges of these less efficient techniques. In a typical hybrid method, a regression model is combined with RTMs (Zhang et al., 2021).

Invertable Forest Reflectance Model (INFORM) is a combination of three RTMs (leaf and canopy) that can be used to simulate forest canopy reflectance in terms of canopy biophysical and biochemical parameters (Atzberger, 2000; Schlerf and Atzberger, 2006). INFORM and machine learning models can be combined using a hybrid method to retrieve plant traits.

The main objectives of this study are to:

1. simulate forest canopy reflectance (for the national park Hunsrück-Hochwald) using the more recent submodels (PROSPECT5, 4SAIL and FLIM) of the coupled chain RTM INFORM
2. train Artificial Neural Networks (ANN) using the simulated canopy reflectance data in order to assess whether the relationship between plant traits and spectral data can be efficiently learned
3. assess whether the trained ANN can deal with collinearity occurring in hyperspectral remote sensing data
4. retrieve forest biophysical and biochemical maps for the study area Hunsrück-Hochwald

2 Forest Disturbances by Bark Beetle

Insect disturbance is just one of the numerous natural disturbances that forests face. Insect outbreaks can have a significant impact on the functioning of an ecosystem (Raffa et al., 2009). However, there have been concerns regarding these forest disturbances as severity and frequency of these disturbances have increased (Millar and Stephenson, 2015).

One of the major insect contributors to European forest disturbances is called the European spruce bark beetle (*Ips typographus*). These bark beetles have been increasingly destroying European forests. These pest organisms specifically affects Norway spruce forests, as *Ips typographus* are major pests of Norway spruce trees (Öhrn et al., 2014). Temperature has a significant effect on the successful swarm of *Ips typographus*. The temperature of 16.5° is considered to be the minimum temperature (lower threshold) for the bark beetle to swarm and attack the trees. The length of warm days is another factor increasing the chance of successful attack by the bark beetle. The European spruce trees have defense mechanisms to fight against the mass beetle outbreak. However, when there are at least three warm days in a row with a temperature well-above the swarming threshold of *Ips typographus*, then the likelihood of these pest organisms successfully overcoming the defense mechanisms of the trees and infesting them is higher (Wermelinger, 2004).

Once a tree is infested, there are typically three stages that the tree will go through. These stages are called the green attack, red attack and gray attack stages (Spritsin et al., 2011). The green attack stage is the first stage and it refers to a stage where usually the leaves of the infested trees are still green. At this stage, it is difficult to distinguish the infested trees at the canopy level (Wulder et al., 2006). During the green attack stage, the infested tree may still be alive, but once this stage has started, the death of the infested tree is inevitable. Moreover, during this stage, the freshly hatched generation of the bark beetle uses the inner bark

2. Forest Disturbances by Bark Beetle

of the tree to grow and develop (Niemann et al., 2005). The next stage is called the red attack stage. During this phase, the leaves of the infested trees slowly turn yellow and eventually become reddish (Fernandez-Carrillo et al., 2020). The red attack stage occurs within the next year (one year) after the bark beetles perform a mass attack (Niemann et al., 2005). The final stage is the gray attack stage, which refers to a stage where the trees have already shed their leaves (Wulder et al., 2009). At this point, the infested trees are completely dead and abandoned by the new generation of the bark beetle (Niemann et al., 2005). During the final stage, typically only the gray bark of the tree is left. Hence, this stage is called the gray stage (Abdullah et al., 2019a).

Bark beetles are known to have a mutual (symbiotic) relationship with many different fungi species (e.g. *Ceratocystis spp*). These fungi species can easily infect trees. Sapwood and phloem of the trees are specifically affected by these species (Niemann et al., 2005; Six et al., 2003). Once the trees are infected by the fungi carried by bark beetles, these fungi will start to disrupt the translocation of water and important nutrients in the infected tree (Paine et al., 1997; Wermelinger, 2004). These are potentially causing changes in both biophysical and biochemical properties at the canopy and leaf level (Abdullah et al., 2018).

There are several factors that help *Ips typographus* to attack trees successfully. One of the major contributors to this process is thought to be climate change. Typically, when bark beetles exist at natural and low densities, they cannot overcome the defense mechanisms of the trees. Climate change, on the other hand, is facilitating these beetles' successful attack in numerous ways. Increased length and number of warm and very dry days, for example, significantly increase the likelihood of a mass bark beetle outbreak. Specifically, these factors trigger beetle attacks in two ways: directly (e.g. increased insect population) and indirectly (e.g. weakened tree defence mechanisms against bark beetle) (Bale et al., 2002; Bentz and Jönsson, 2015; Marini et al., 2017).

Effective forest management and early intervention is essential in order to deal with bark beetle outbreaks. Detecting the infested trees while they are still going

2. Forest Disturbances by Bark Beetle

through the green attack stage is essential, because at this point, the beetles have not yet left the tree to look for new host trees. One of the traditional ways of identifying and assessing infested trees is through manual field surveys. Typically, these surveys are performed by foresters who try to look for sawdust within the bark of the trees. However, this method is extremely difficult, time-consuming and require a lot of labor to study large forest areas. The field of remote sensing, however, offers methods to more effectively detect and map forest infestations (Abdullah et al., 2019b; Immitzer and Atzberger, 2014).

Immitzer and Atzberger (2014) used World-View-2 data in order to study and detect bark beetle infestations during the green attack stage. In this study, a forest infestation map was produced using a machine learning method. Study by Ortiz et al. (2013) used RapidEye and TerraSAR-X data for early detection of bark beetle. This study demonstrated the use of statistical and machine learning methods for bark beetle detection. Lausch et al. (2013) used hyperspectral airborne data HyMap to study early detection of bark beetle infestations with a limited amount of success. In the study by Lausch et al. (2013), single and multi-date Landsat data were successfully utilized to study forest disturbance by the bark beetle.

There have only been limited studies assessing the effect of the bark beetle green attack stage on plant biophysical and biochemical properties. A study by Abdullah et al. (2018) showed that there is a significant difference in leaf chlorophyll and nitrogen content between healthy and infested trees during the green attack stage. This study also concluded that spectral responses (using in-situ data) between healthy and infested trees were significantly different. Work by Abdullah et al. (2019b) concluded that the leaf chlorophyll and water content of healthy plants were significantly higher than infested trees at the green attack stage. They found dry matter content to be significantly higher for infested trees than healthy trees. A recent study by Ali et al. (2021) for the first time demonstrated the use of canopy chlorophyll content time series data retrieved from multispectral remote sensing data using the INFORM RTM for bark beetle infestation detection.

3 Radiative Transfer Models (RTM)

Radiative transfer models take advantage of physical laws that determine “causal and effect” relationships. There have been various radiative transfer models developed (leaf, canopy etc.) during the last decades (Verrelst et al., 2019).

Combatal et al. (2003) describe that RTM inversion process involves modifying the vegetation biophysical parameters (variables) $V = \{V_1, V_2, \dots, V_{n_{var}}\}$ in such a way that the simulated canopy reflectance simulated by an RTM M is most similar to the canopy reflectance R measured by a remote sensing sensor. The n_{var} number of canopy variables and geometric parameters C are required by the RTM M . It is also important to mention that the inversion is achieved with an error or uncertainty, namely ε :

$$R = M(V, C) + \varepsilon \quad (3.1)$$

The error term ε in the Equation (3.1) refers to the uncertainties in both measured and modeled canopy reflectance (Combatal et al., 2003).

Radiative transfer models can be used to retrieve forest canopy biophysical and biochemical variables from remote sensing satellite data. This chapter mainly focuses on a coupled-chain radiative transfer model called the Invertable Forest Reflectance Model (INFORM) and its sub-models. INFORM combines leaf and canopy radiative transfer models (Atzberger, 2000), namely PROSPECT, SAIL (Scattering by Arbitrary Inclined Leaves) and FLIM (Forest Light Interaction Model) models (Schlerf and Atzberger, 2006). There are different versions of some of these sub-models (PROSPECT4, PROSPECT5 etc.). This chapter specifically reviews the PROSPECT5, 4SAIL and FLIM models.

3.1 Leaf Radiative Transfer Model - PROSPECT

In essence, PROSPECT is a leaf Radiative Transfer Model (RTM) that takes as an input leaf plant variables and outputs leaf reflectance and transmittance for the wavelength range of 400nm-2500nm (Jacquemoud and Baret, 1990). According to Yang et al. (2021), several versions of this leaf model have been developed during recent years. For example, the PROSPECT4 and PROSPECT5 models are two of the recent and improved versions of the previously proposed models. The main advantage of PROSPECT5 over PROSPECT4 is that the PROSPECT5 version takes carotenoid content of the leaf into account apart from chlorophyll (a+b) content separately (Yang et al., 2021).

Reflectance and transmittance of plant leaves are known to be driven by leaf biochemical parameters. The PROSPECT models have been successfully used both to simulate leaf optical properties (transmittance and reflectance) and vegetation leaf biochemical variables. One of the main strengths of this model is that it can easily be applied to many different plant species living under various environmental conditions (Yang et al., 2021). In the PROSPECT5 model, leaf reflectance and transmittance are treated as a function of 6 plant leaf parameters (Table 3.1) (Jacquemoud and Ustin, 2019).

Table 3.1: Input parameters of PROSPECT5

Parameter	Symbol	Unit
Leaf structure parameter	N	-
Chlorophyll content	Cab	$\frac{\mu g}{cm^2}$
Leaf carotenoid content	Car	$\frac{\mu g}{cm^2}$
Brown Pigment Content	Cbrown	-
Equivalent water thickness	Cw	$\frac{g}{cm^2}$
Leaf dry matter content	Cm	$\frac{g}{cm^2}$

These leaf parameters (Table 3.1) can give very important insights about the plant's health status. For example, the chlorophyll (a+b) content of a plant leaf is a useful indicator of plant photosynthetic functioning (Darvishzadeh et al., 2019a,

3. Radiative Transfer Models (RTM)

Feret et al. (2008)). Chlorophyll is one of the most important plant traits that can also be used to understand plant infestations and diseases (Darvishzadeh et al., 2019a). Ecological and environmental stresses can impact the carotenoids content of plants. Furthermore, there is a strong relationship between nitrogen and chlorophyll concentration (Feret et al., 2008). It is also important to note that, these parameters (e.g. chlorophyll content) can also be used to study the whole ecosystem functioning (e.g. ecosystem productivity) apart from being very useful for understanding individual plant health status (Darvishzadeh et al., 2019a).

Some of these parameters can be analyzed in laboratories. For instance, the chlorophyll content of leaves can be analyzed in a laboratory, and laboratory techniques typically provide the most accurate results. However, these in-situ techniques are very expensive and invasive (e.g. destruction of the leaves is necessary). Here, remote sensing data are very advantageous as they offer a much more efficient methodology for retrieving these variables. Additionally, remote sensing data can cover a large spatial area, making it much more efficient to retrieve plant leaf parameters (Table 3.1) for this specific area (Darvishzadeh et al., 2019a).

3.2 Scattering by Arbitrary Inclined Leaves (SAIL)

SAIL is a canopy radiative transfer model that is used in order to simulate bi-directional canopy reflectance (Verhoef, 1984). This model (Verhoef, 1984) was extended from the study by Suits (1971). Furthermore, the 4SAIL model was proposed in the study by Verhoef et al. (2007), which is numerically much more optimized and computationally more efficient than the previously proposed versions.

The SAIL model can be considered to be a simple model and takes only a few variables as its inputs. SAIL requires leaf reflectance and transmittance as well as background reflectance. Leaf transmittance and reflectance can be simulated using the PROSPECT model and directly fed into the SAIL model (Berger et al., 2018). Table 3.2 shows the input variables that the 4SAIL model requires.

3.2. Scattering by Arbitrary Inclined Leaves (SAIL)

Table 3.2: Input parameters of 4SAIL

Parameter	Symbol	Unit
Average leaf inclination angle	ALIA	$^{\circ}$
Leaf area index	LAI	$\frac{m^2}{m^2}$
Hot spot parameter	Hot	$\frac{m}{m}$
Solar zenith angle	tts	$^{\circ}$
Observer zenith angle	tto	$^{\circ}$
Sun-sensor azimuth angle	psi	$^{\circ}$
Soil brightness	α_{soil}	-
Leaf reflectance	ρ	nm
Leaf transmittance	τ	nm
Backroung reflectance	bgr	nm

The parameters observer zenith angle, sun zenith angle and sun-sensor azimuth angle (Table 3.2) are usually to be provided to the SAIL model explicitly. The parameter *Hotspot* refers to the ratio between leaf size and canopy height (Berger et al., 2018). It is important to note that SAIL is mainly designed to work better with relatively homogeneous forests. However, some researchers have also shown that this model can yield reasonable results when used on heterogeneous canopies (Jay et al., 2017).

The PROSPECT and SAIL models can be combined together to simulate canopy reflectance. The combination of PROSPECT and SAIL models is usually called PROSAIL in the literature. PROSAIL is one of the most famous models that is used for simulation of canopy reflectance and estimation of plant biophysical parameters (Berger et al., 2018; Jay et al., 2017).

The PROSAIL model (Figure 3.1) can be used in a forward mode to get the simulated canopy reflectance based on the input parameters of PROSPECT (Table 3.1) and SAIL (Table 3.2) (Berger et al., 2018).

3. Radiative Transfer Models (RTM)

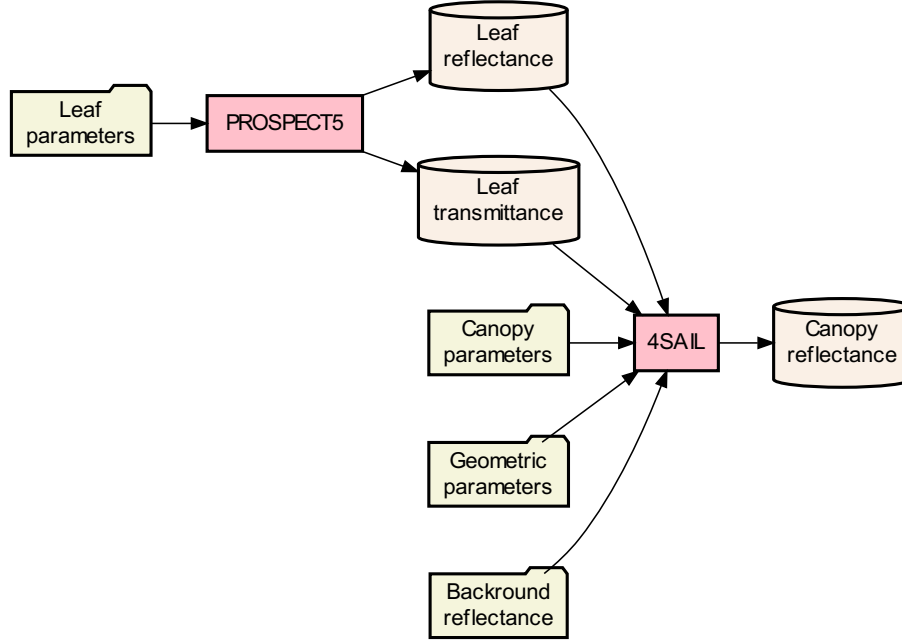


Figure 3.1: Diagram of PROSAIL as a combination of PROSPECT5 and 4SAIL in the forward mode

3.3 Forest Light Interaction Model (FLIM)

The FLIM model was first developed by Rosema et al. (1992), and it has stochastic properties. The FLIM model assumes that forests consist of discontinues mixture of gaps and crowns. In this model, canopy reflectance is considered to be a probability of seeing a gap (ground) or crown (Rosema et al., 1992). Table 3.3 shows the input parameters of the FLIM model.

Table 3.3: Input parameters of FLIM

Parameter	Symbol	Unit
Canopy reflectance at infinite depth	R_c	nm
Background reflectance	R_g	nm
Transmission in viewing direction	T_o	nm
Transmission in sun direction	T_s	nm
Stand density	SD	ha^{-1}
Crown diameter	CD	m
Mean crown height	H	m
Solar zenith angle	tts	$^{\circ}$
Observer zenith angle	tto	$^{\circ}$
Sun-sensor azimuth angle	psi	$^{\circ}$

3.4. Invertable Forest Reflectance Model (INFORM)

The FLIM model assumes that the background reflectance (R_g) is known (Rosema et al., 1992) and it is the reflectance coming from the forest floor (Atzberger, 2000). Transmission in viewing (T_o) and sun (T_s) direction parameters (Table 3.3) can be estimated using the SAIL model (Atzberger, 2000; Schlerf and Atzberger, 2006).

3.4 Invertable Forest Reflectance Model (INFORM)

The INFORM model was first developed by Atzberger (2000), and it consists of three sub-models. INFORM innovatively combines the three RTMs PROSPECT, SAIL and FLIM (Atzberger, 2000; Schlerf and Atzberger, 2006). The INFORM model can be used to efficiently simulate forest canopy bi-directional reflectance within the wavelength range of 400nm-2500nm (Schlerf and Atzberger, 2006) and, compared to other 3D radiative transfer models, INFORM requires fewer input variables (Atzberger, 2000; Ali et al., 2020).

The SAIL model can be used to calculate the parameters transmittance in the viewing (T_o) and sun (T_s) directions (Table 3.3) (Verhoef, 1984). However, in the SAIL model, the fact that the crown transmittance can be impacted (reduced) by woody parts of trees is ignored (Atzberger, 2000). Therefore, in the study by Atzberger (2000), it is shown that the INFORM model modifies the parameters transmittance in the viewing (T_o) and sun (T_s) direction in such a way that considers the shadow on the ground cast by woody parts of trees. However, in the study by Schlerf and Atzberger (2006), this technique was not applied because of the difficulties with parameterisation. Furthermore, INFORM represents leaf area index (LAI) as LAI of single trees (LAI_s) by dividing LAI by the canopy closure (Ali et al., 2016, 2020).

As the INFORM model is a combination of the three sub-models, its input parameters are simply the parameters of the sub-models. Therefore, the whole input parameters of the INFORM model can be found in Table 3.1 (PROSPECT5),

3. Radiative Transfer Models (RTM)

Table 3.2 (4SAIL) and Table 3.3 (FLIM). Figure 3.2 shows how INFORM can be used in the forward mode as a combination of PROSPECT5, 4SAIL and FLIM.

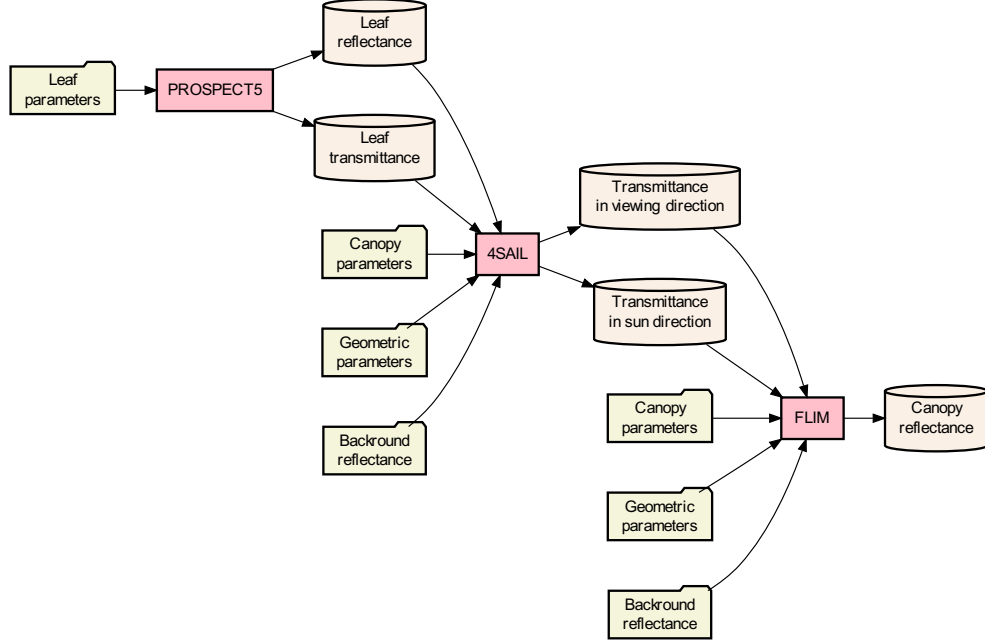


Figure 3.2: Visualization of the INFORM model as a combination of PROSPECT5, 4SAIL and FLIM in the forward mode

Successful applications of the INFORM model to retrieve forest biophysical parameters from hyperspectral remote sensing satellite data have been confirmed by various studies. For example, in the studies by Ali et al. (2016) and Wang et al. (2017), the INFORM model was successfully used to retrieve forest biophysical maps from the hyperspectral airborne remote sensing data HySpex. INFORM's suitability to be applied for multispectral remote sensing data was also proved in different studies. For example, Darvishzadeh et al. (2019a) used the INFORM RTM in order to retrieve forest leaf chlorophyll content maps from the multispectral remote sensing satellite data Sentinel-2 and RapidEye. Ali et al. (2020) also utilized the INFORM model to retrieve forest canopy plant traits from Sentinel-2 multispectral remote sensing data.

3.5 Challenges of RTMs

One of the main challenges of the RTMs is the so-called ill-posed problem (Combal et al., 2003; Zhu et al., 2019). RTMs are mainly used for inversion problems. And inversion problems are usually ill-posed problems as opposed to well-posed problems. The ill-posed problem refers to a problem where multiple solutions can lead to similar or the same output. It makes the radiative transfer modelling process a very challenging process as multiple sets of solutions (e.g. leaf/canopy variables) can yield very similar simulated reflectance to the measured reflectance (Combal et al., 2003; Darvishzadeh et al., 2008; Zhu et al., 2019).

Fortunately, the problem of ill-posedness can be efficiently addressed using prior information (Combal et al., 2003; Darvishzadeh et al., 2008; Zhu et al., 2019). Combal et al. (2003) divides this prior information into three groups. The first category of prior information is to use data coming from the same spatial area but collected using another sensor (e.g. radar). The second type of prior information is the knowledge about the architecture of the canopy that is being studied. This knowledge could help decide what type of RTM to select (turbid medium, geometric etc.). Finally, the last type of prior information to be considered is the knowledge of the usual distribution range of biophysical and biochemical variables of canopies. This information can be gained with the help of either specialist/expert knowledge or by combining experimental data (Combal et al., 2003).

4 Machine Learning Regression Algorithms (MLRA)

Machine Learning Regression Algorithms (MLRA) are non-linear and non-parametric techniques that can be used to retrieve forest biophysical and biochemical parameters. Apart from non-parametric methods, there are also parametric techniques that can be used for the same purpose. However, the main advantage of the non-parametric methods over the parametric ones is the fact that they often do not assume any underlying relationships between the response (e.g. plant traits) and predictor (e.g. canopy reflectance) variables. These models do not rely on any linearity, which makes them very compelling techniques that can be applied in various research fields (Sinha et al., 2020; Rivera et al., 2015; Verrelst et al., 2019).

According to the extensive review study by Verrelst et al. (2019), Decision Trees (DT), Kernel-Based Machine Learning Regression Methods (KBMLRM) and Artificial Neural Networks (ANN) are the three most widely used Machine Learning (ML) families for retrieval of plant biophysical and biochemical variables in remote sensing. This chapter focuses on the most commonly used ML techniques and briefly reviews them.

4.1 Decision Trees (DT)

Decision Tree (DT) methods rely on trees and branches to illustrate the outcome of each decision (Verrelst et al., 2019). The Random Forest (RF) is a non-parametric learning model and it is also considered to be an ensemble model (Breiman, 2001). Essentially, the Random Forest Regression (RFR) algorithm builds multiple small regression trees where each tree has a vote on the prediction. Based on the votes of each tree, a final prediction is made (Breiman, 2001; Powell et al., 2010). The major advantage of the Random Forest model is that it is not sensitive to overfitting (Breiman, 2001; Powell et al., 2010; Verrelst et al., 2019), which is a well-known phenomenon challenging many ML algorithms' performance. Apart from that,

4.2. Kernel-Based Machine Learning Regression Methods (KBMLRM)

the RF algorithm can successfully deal with large amounts of training data as well as outliers and noise in the training data. These properties are the reason why the RF model is an attractive method for plant trait mapping applications in remote sensing (Verrelst et al., 2019).

Essentially, there are two main parameters of the RF algorithm. These are the number of trees (ntree) and number of variables (mtry) that will be randomly sampled at every split (Wang et al., 2018a). Breiman (2001) suggests that a tree count of 500 (ntree = 500) may be sufficient in many cases.

The applicability of the RFR algorithm for retrieval of plant traits has been demonstrated in many studies. For example, Ali et al. (2020) indicates that, when compared to the traditional Look Up Table (LUT) method with merit function, the RFR’s retrieval accuracy was higher. In the studies by Han et al. (2016) and Pullanagari et al. (2016), RFR was compared to the other widely used ML algorithms (e.g. SVM). In these studies, no method was found to be significantly superior to one another, indicating that all the widely used ML methods are highly competitive (Verrelst et al., 2019).

4.2 Kernel-Based Machine Learning Regression Methods (KBMLRM)

Kernel-based learning algorithms rely on turning the current dimension of the data into a higher dimension and solving non-linear problems using a kernel function. Kernel methods offer very flexible implementation and they can be efficiently utilized as long as the linear problem can be solved in terms of dot products (linear algebra) (Verrelst et al., 2019). According to the review study by Gómez-Chova et al. (2011), kernel-based methods play a very important role in the remote sensing field. Kernel-based models can efficiently deal with small number of training samples (training data), very high-dimensional and noisy data sets. Unlike kernel-based algorithms, other machine learning methods, such as neural networks (NN), are known to be sensitive to the noise and high dimensions of the data, and importantly, they tend

4. Machine Learning Regression Algorithms (MLRA)

to perform very poorly when large training data sets are not available. All of these properties make kernel-based algorithms very attractive alternatives for remote sensing researchers as the mentioned issues that kernel-based algorithms handle are well-known for remote sensing data (Gómez-Chova et al., 2011).

Support Vector Regression (SVR) (Drucker et al., 1997) is one of the most widely used KBMLRM (Verrelst et al., 2019). One of the advantages of Support Vector Machines (SVM) (Boser et al. (1992)) over other ML methods such as ANN is that it mainly relies on minimizing the risk function, as opposed to trying to minimize the error in the training data set. Artificial neural networks, however, tries to minimize the error function, which makes them very susceptible to overfitting to the training data (Karimi et al., 2008). The applicability of SVM to estimate plant biophysical traits has been demonstrated in various studies. For example, Karimi et al. (2008) and Yang et al. (2011) used SVM to estimate plant biophysical parameters from hyperspectral remote sensing data. A study by Tuia et al. (2011) demonstrated the successful use of multioutput SVR in order to retrieve the estimated biophysical parameters. Various studies compared the performance of SVR to the other ML methods. For example, Pullanagari et al. (2016) used different methods to map macro and micro nutrients from hyperspectral data. This study concluded that although SVR performed better for a certain parameter retrieval, this algorithm was not superior to the other ML methods in general.

Another widely used kernel-based ML method is Gaussian Process Regression (GPR). GPR is a Bayesian, non-parametric, and probabilistic ML method that provides important insights for retrieval of plant traits from remote sensing data (Camps-Valls et al., 2016, 2019). The major advantage of GPR over the other widely used ML algorithms is that GP models provide confidence intervals for the estimations (Berger et al., 2020). In other words, besides providing very good results, GPR can also give information about how much uncertainty (e.g. error bars) exists in the predictions or estimations of the retrieved parameters. GPR can easily deal with different types of data and it can be implemented in a way that efficiently handles noise in the data. These issues commonly occur in remote sensing

4.3. Artificial Neural Networks (ANN)

data, which makes the GPR algorithm very interesting for the remote sensing community (Camps-Valls et al., 2016, 2019). Successful applications of GPR to retrieve plant biophysical and biochemical variables have been demonstrated in the literature. A relatively recent study by Berger et al. (2020) used two versions of GPR (homoscedastic and heteroscedastic) for retrieval of plant nitrogen content. This study showed that the both versions of GPR performed well for the retrieval of plant nitrogen content. Upreti et al. (2019) used different ML methods to retrieve plant traits from the multispectral remote sensing satellite data Sentinel-2. They found GPR to be the best performing algorithm within the cross-validation stage, but not in general (Upreti et al., 2019). A study by Caicedo et al. (2014) compared different ML models in terms of accuracy and concluded that GPR yielded the most accurate results. However, it is important to note that GPR does not come without drawbacks. For example, one of the main challenges of GPR is that it is computationally costly despite advances (Camps-Valls et al., 2019).

4.3 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) is a powerful ML method that consists of neurons and layers (Simon, 1999). Usually, the ANN has three main components: an input layer, (a) hidden layer(s) and an output layer (Jensen et al., 1999; Quan et al., 2017). A typical mechanism for training an ANN is that first a training data set is given to the model as an input layer and then, the model trains and predicts output. Once the predicted output is available, the current predicted output is compared to the true output so that the weight parameters can be adjusted in a way that the predicted and true output are closely similar (Ingram et al., 2005; Quan et al., 2017). This learning process can be repeated multiple times until similarity between the predicted and true outputs is at a certain threshold (e.g. convergence). This threshold can be defined by the user depending on their preferences (Jensen et al., 1999; Quan et al., 2017).

4. Machine Learning Regression Algorithms (MLRA)

Main advantages of ANN are their simplicity, computational efficiency and their ability to learn from data where linearity is not assumed (Schlerf and Atzberger, 2006; Walczak, 2019). Furthermore, ANNs do not require any prior information about the distribution that the data come from unlike the traditional statistical methods (Walczak, 2019).

ANNs have been utilized in the remote sensing domain, especially for plant property mapping, since the 1990s (Verrelst et al., 2019). There are many studies in the literature that show the superiority of the ANN methods to more traditional statistical methods. For example, studies by Malenovsky et al. (2013) and Kalacska et al. (2015) showed that compared to the traditional statistical and vegetation index (VI) methods, ANN yielded more accurate results for retrieval of plant parameters from hyperspectral remote sensing data. Neinavaz et al. (2016) compared ANN's performance to other methods (e.g. linear parametric) and found that ANN was superior to the compared methods for retrieval of an important plant variable, LAI. Moreover, a recent study by Danner et al. (2021) used four widely used ML methods to retrieve plant traits and compared the results of each method. In general, ANN was found to be the best performing ML method among the others, considering the models' efficiency, robustness, accuracy, and computational time.

Despite being a powerful ML method, ANN comes with drawbacks. There are many descriptors in a typical ANN that may be correlated to each other. Because of that, there is a risk of the model being stuck in the local minima. Another well-known challenge of ANN is that it is susceptible to overfitting. Fortunately, there have been many methods developed to deal with these problems (e.g. regularization) (Ghasemi et al., 2018). A study by Schlerf and Atzberger (2006) points out another potential drawback, which is the fact that the ANN model may behave unpredictably when the model does not represent the spectral properties of the target variable well.

4.4 Challenges of Machine Learning Regression Algorithms (MLRA)

There are common issues that almost all MLRAs are challenged with. Hyperspectral remote sensing data come with many bands that may be highly correlated and noisy. This problem is a well-known problem and it is also called the “Hughes phenomenon” (Hughes, 1968) or the “curse of dimensionality” (Danner et al., 2021). One of the main problems is that MLRAs’ computational cost becomes higher with an increased amount of training data. Most MLRAs require high computational power for very costly training phases. Apart from that, many highly correlated bands may indicate highly redundant data which cause difficulties for statistical and ML methods (Rivera-Caicedo et al., 2017).

There are some methods that can be used to efficiently deal with the “curse of dimensionality” problem. Some of the most commonly used methods for dealing with this issue involve reducing the space (dimension) of the original data. Rivera-Caicedo et al. (2017) indicate that this issue can be solved in two different domains. The first method involves choosing the samples that provide most of the information compared to the other samples. The second domain involves the use of feature selection or dimensionality reduction (DR) techniques. In the first technique, the amount of samples is minimized but accuracy is still a priority (Rivera-Caicedo et al., 2017).

The second set of techniques (DR) tries to reduce the original data space while keeping most of the information available. In other words, DR techniques convert a large number of features with a high amount of redundancy into a much smaller feature set with no or as little redundancy as possible (Lee and Verleysen, 2007). Two of the commonly used DR techniques to compress the spectral reflectance data in remote sensing are so-called wavelet transform (WT) and principal component analysis (PCA) (Ke et al., 2016).

After reducing the dimension of the original data, new and low-dimensional data can be used to train ML models. MLRAs can be combined with PCA to reduce the computational cost and avoid the “curse of dimensionality” problem. There

4. Machine Learning Regression Algorithms (MLRA)

have been studies comparing the performance of ML methods when combined with DR techniques to feature selection or VI based ML applications. For example, Liu and Pan (2017) compared the performance of PCA based ANN with VI based ANN and concluded that using PCA before training is superior to using VI in the training phase. However, it is also important to note that PCA is not the only DR technique, although it is the most commonly used DR technique in plant retrieval studies by remote sensing researchers. A study by Rivera-Caicedo et al. (2017) reviews and compares different DR techniques and concludes that PCA is not always the best performing DR technique and the use of other techniques may yield better results in some cases.

In general, most ML algorithms need a large amount of training data to learn from the training data and generalize well to the data that were not used in the training phase (e.g. test data). This leads to another challenging property of ML for retrieval of plant biophysical traits. This is because of the fact that it is extremely difficult to collect enough training data for plant traits in the field for a ML method to perform well. It requires various *in-situ* field techniques and these techniques need to be used in a limited time-frame. It is still not guaranteed that training data collected in the field can generalize well to other fields (Danner et al., 2021). One way to overcome this problem is to use the so-called hybrid-machine learning method.

5 Hybrid Machine Learning Method

The hybrid machine learning method is a combination of physically based RTMs and MLRAs. The main idea behind this technique is to make use of the advantages of the two different methods at the same time. RTMs can provide simulated data based on well-known and approved physical laws. Therefore, the data provided by RTMs can be applicable universally. Machine learning methods provide opportunities to efficiently deal with non-linear data (without making assumptions) in a reasonable time frame (Abdelbaki et al., 2021; De Grave et al., 2020; Berger et al., 2021; Ke et al., 2016). The study by Fernández-Guisuraga et al. (2021) mentions three main advantages of using the hybrid method over other techniques: better generalization of the model, better estimation of the biophysical parameters, and finally, more efficient computation.

A typical procedure for using hybrid machine learning methods to retrieve plant biophysical variables involves simulating coupled leaf and canopy models (e.g. PROSAIL) and storing the simulated data in the so-called look-up-table (LUT). The LUT typically stores simulated reflectance data and a wide range of plant parameters. Once the LUT is available, a ML model can be trained using all the information available in the LUT as a training data set (Verrelst et al., 2019).

There have been a large number of studies utilizing hybrid machine learning methods for retrieval of plant traits. For example, a recent study by César de Sá et al. (2021) combines PROSAIL and widely used MLRAs with the hybrid approach in order to retrieve plant parameters from Sentinel-2 satellite data. In this study, the performance of each MLRA combined with PROSAIL was assessed when noise was added. In the study by Wei et al. (2017), plant LAI was successfully estimated using random forest regression and PROSAIL in a hybrid work flow. Verrelst et al. (2016) used hybrid RTM and kernel-based machine learning techniques to

5. Hybrid Machine Learning Method

estimate plant leaf area index and chlorophyll content. This study included the active learning (AL) technique for an efficient retrieval strategy.

6 Satellite Data

In this study, the hyperspectral satellite data PRISMA (PRecurso IperSpettrale della Missione Applicativa) was utilized. The PRISMA sensor has Field of View (FOV) of 2.45° and swath of 30 km. The sensor takes images within the spectral range of 400-2500nm. The PRISMA image has 66 bands in the visible and near infrared (VNIR) region (400-1010nm) and 173 bands within the short-wave infrared (SWIR) region (920-2505nm) of the spectrum. The spectral resolution of the data is somewhere between 6nm and 12nm. The hyperspectral data has a ground sampling distance (GSD) of 30 m (Candela et al., 2016; Giardino et al., 2020; Verrelst et al., 2021). The scientific fields that are expected to largely benefit from the PRISMA data are mainly environmental, climate change and forest research areas, among others (Giardino et al., 2020).

In this study, L2 product of PRISMA data was retrieved and processed. First, the image was cropped to the study area using a polygon shape file covering the study area. Only 231 bands of the PRISMA image were available and all the available bands were utilized for further analysis. The PRISMA image contained clouds and shadow within the national park area and therefore, pre-processing of the image was necessary. The software package ENVI was used in order to accurately mask out the shadows and clouds available within the study area.

7 Study area

Hunsrück-Hochwald is a German national park that was founded in 2015. The national park covers areas in both Saarland and Rhineland-Palatinate German federal states (Figure 7.1).

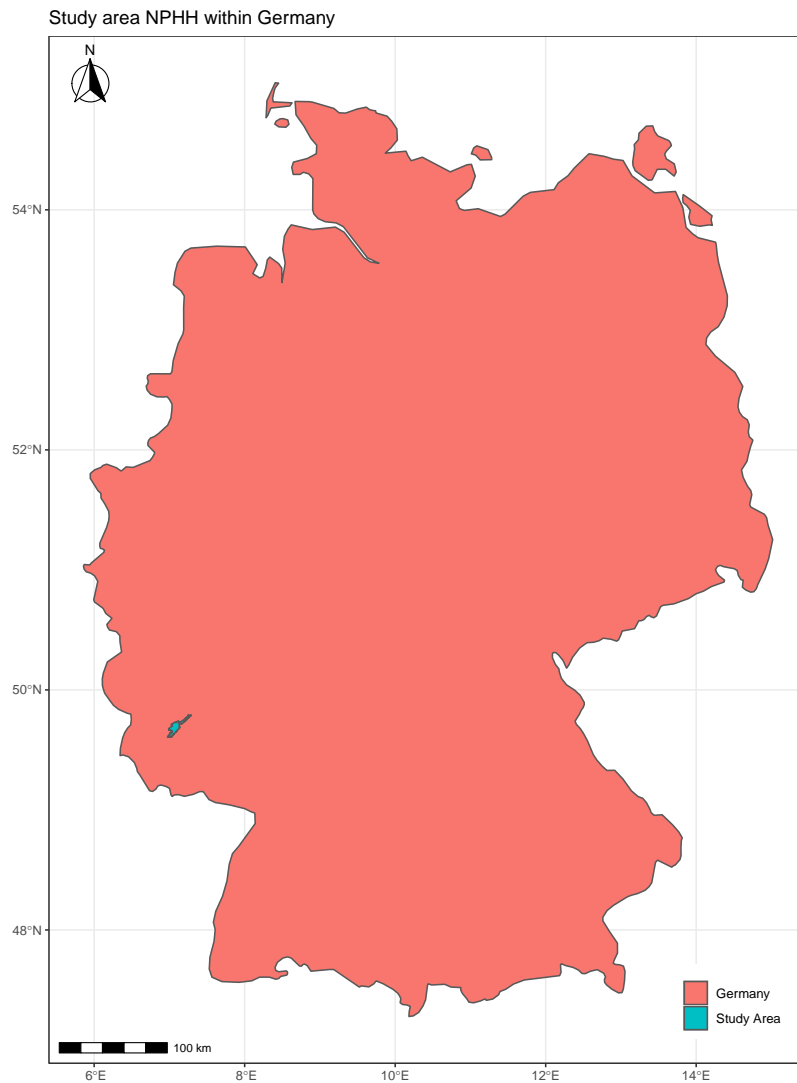


Figure 7.1: The study area on Germany's map.

The area of the national park is about 10,000 hectare (Fischer-Stabel and Mattern, 2016). There are various forestry activities (selective cutting etc.) occurring in the national park area. The forest is mainly dominated by the plant species

7. Study area

Norway spruce (*Picea abies*), oak (*Quercus petraea*) and beech (*Fagus sylvatica*) (Buddenbaum et al., 2015).

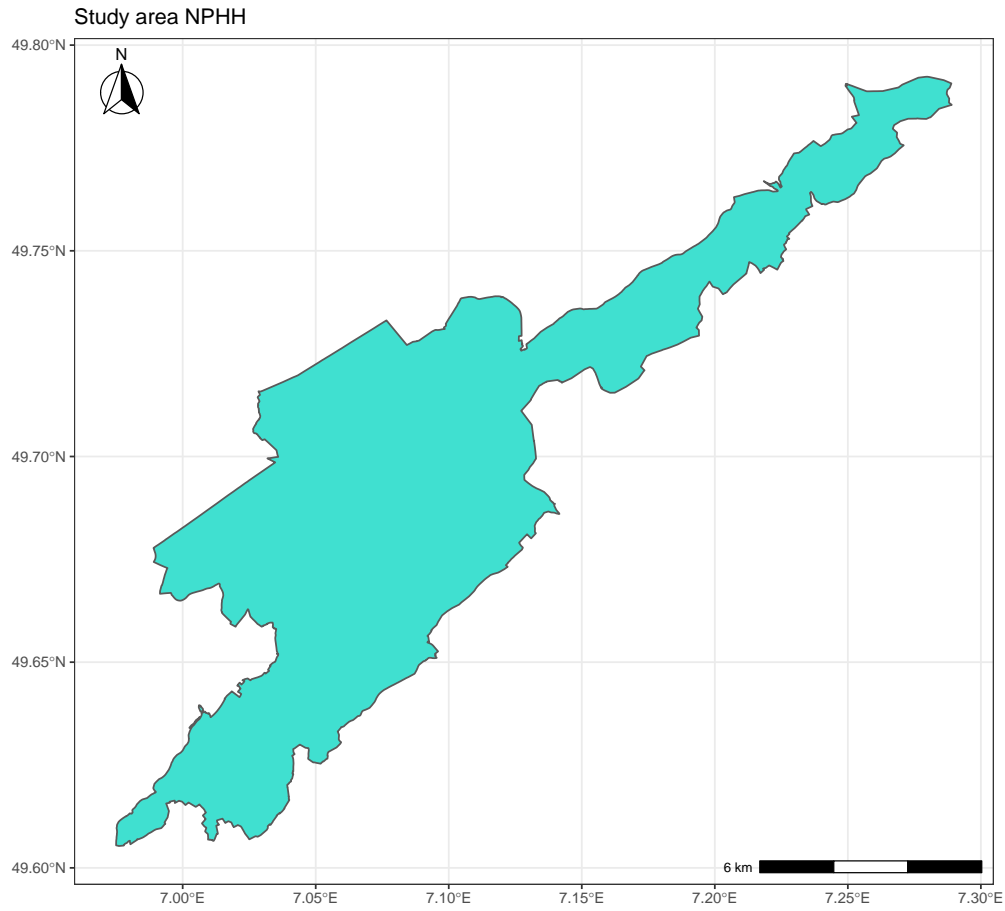


Figure 7.2: The study area.

8 Methods

This section explains the methods used in this research.

8.1 Local sensitivity analysis

Local sensitivity analysis was performed to assess the effect of each of the main 6 plant biochemical and biophysical variables on the PRISMA image bands. In the local sensitivity analysis, simulation is performed by keeping all the variables constant at their determined or default values except the parameter of interest. This way the effect of a specific parameter on the simulated spectra can be assessed. In this research the plant parameters C_{ab} , C_w , C_m , LAI_s , CD and SD were varied each 15 times (Table 8.1), while keeping the rest of the variables at their default values (Table 8.2). The default and varied values were chosen based on the available studies in the literature (e.g. Darvishzadeh et al. (2019a); Laurent et al. (2011); Schlerf and Atzberger (2012)) where similar RTM method was used to simulate reflectance for Spruce trees.

Table 8.1 shows the 6 parameters that were varied, their units, minimum and maximum values. Each parameter was varied 15 times, meaning 15 different spectra were simulated for each variable.

Table 8.1: INFORM parameters varied in local sensitivity analysis (each parameter were varied 15 times)

Parameter	Abbrev.	Unit	Min	Max
Chlorophyll content	C_{ab}	$\frac{\mu g}{cm^2}$	20	60
Equivalent water thickness	C_w	$\frac{g}{cm^2}$	0.0035	0.035
Leaf dry matter content	C_m	$\frac{g}{cm^2}$	0.008	0.03
Leaf area index (single)	LAI_s	$\frac{m^2}{m^2}$	0	7
Stem density	SD	ha^{-1}	200	5000
Crown diameter	CD	m	1.5	8.5

8.1. Local sensitivity analysis

Table 8.2 shows the determined default values for each INFORM parameter that were kept during the sensitivity simulation while one of the parameter was varied (Table 8.1).

Table 8.2: INFORM Parameters that were kept constant while one parameter was varied at a time

Parameter	Abbr	Unit	Value
Leaf structure parameter	N	—	3
Chlorophyll content	C_{ab}	$\frac{\mu g}{cm^2}$	40
Leaf carotenoid content	C_{ar}	$\frac{\mu g}{cm^2}$	8
Brown Pigment Content	C_{brown}	—	0.001
Equivalent water thickness	C_w	$\frac{g}{cm^2}$	0.0117
Leaf dry matter content	C_m	$\frac{g}{cm^2}$	0.03
Average leaf inclination angle	$ALIA$	$^\circ$	65
Leaf area index (single)	LAI_s	$\frac{m^2}{m^2}$	6
Leaf area index (understorey)	LAI_u	$\frac{m^2}{m^2}$	0.5
Hot spot parameter	Hot	$\frac{m}{m}$	0.02
Solar zenith angle	tts	$^\circ$	45.43
Observer zenith angle	tto	$^\circ$	0
Sun-sensor azimuth angle	psi	$^\circ$	181.41
Soil brightness	α_{soil}	—	0.5
Stem density	SD	ha^{-1}	700
Crown diameter	CD	m	5
Mean Height	H	m	20
Fraction of diffuse incoming	$skyl$	—	0.1
Soil reflectance spectrum	B_g	—	default

Solar zenith angle and *Sun-sensor azimuth angle* were calculated based on the PRISMA image acquisition parameters (date, lat/long etc.) using the *solar position calculator* at <https://www.esrl.noaa.gov/gmd/grad/solcalc/azel.html>.

RTM models PROSPECT5, 4SAIL and FLIM were coupled (INFORM) in order to simulate canopy reflectance. Simulations were carried out using the *ccrtm* package (Visser, 2021) in *R* (R Core Team, 2021). The default soil spectra provided by the *ccrtm* package (Visser, 2021) was used for the simulations. Spectral resampling was performed in order to resample the INFORM output spectra (1nm

8. Methods

resolution between 400nm and 2500nm) into PRISMA image bands. For spectral resampling the *R* package *hdsar* (Lehnert et al., 2019) was utilized.

8.2 RTM simulation (INFORM)

PROSPECT5, 4SAIL and FLIM RTM models were coupled (INFORM) to simulate forest canopy reflectance based on different values of plant biophysical and biochemical parameters. The 6 parameters that were mentioned in the previous chapter were varied and spectra was simulated based on each combination of these variables. The number of combinations increase exponentially, which in turn requires increased computational power. Therefore, the trade-off must be taken into account between computational power or time and accurate simulation.

Different authors suggest different number of LUT size for RTM simulation. For example, Danner et al. (2021) mention that LUT size of minimum 50,000 is recommended. Ali et al. (2020) and Darvishzadeh et al. (2019a) created a LUT size of 100,000 and 500,000 respectively.

In this research, LUT size of 316,800 was created based on each combination of different plant biophysical and biochemical parameters. The range of the varied parameters and parameters that were kept constant were determined based on the suggestions of the studies that were mentioned in the previous chapter. These studies used similar methods to simulate canopy reflectance for mainly Spruce forests/trees.

Table 8.3 shows the variables that were used to simulate forest canopy parameters. Table 8.3 also contains information about the range of the values and how many times each parameter was varied.

8.2. RTM simulation (INFORM)

Table 8.3: Range of full input parameters that were used to create a LUT size of 316800

Parameter	Abbr	Unit	Min	Max	Steps
Leaf structure parameter	N	—	3	3	—
Chlorophyll content	C_{ab}	$\frac{\mu g}{cm^2}$	20	60	15
Leaf cartenoid content	C_{ar}	$\frac{\mu g}{cm^2}$	8	8	—
Brown Pigment Content	C_{brown}	—	0.001	0.001	—
Equivalent water thickness	C_w	$\frac{g}{cm^2}$	0.0035	0.035	10
Leaf dry matter content	C_m	$\frac{g}{cm^2}$	0.008	0.03	11
Average leaf inclination angle	$ALIA$	$^\circ$	65	65	—
Leaf area index (single)	LAI_s	$\frac{m^2}{m^2}$	0	6.5	16
Leaf area index (understorey)	LAI_u	$\frac{m^2}{m^2}$	0.5	0.5	—
Hot spot parameter	Hot	$\frac{m}{m}$	0.02	0.02	—
Solar zenith angle	tts	$^\circ$	45.43	45.43	—
Observer zenith angle	tto	$^\circ$	0	0	—
Sun-sensor azimuth angle	psi	$^\circ$	181.41	181.41	—
Soil brightness	α_{soil}	—	0.5	0.5	—
Stem density	SD	ha^{-1}	200	5000	4
Crown diameter	CD	m	1.5	8.5	3
Mean Height	H	m	20	20	—
Fraction of diffuse radiation	$skyl$	—	0.1	0.1	—
Soil reflectance spectrum	B_g	—	default	default	—

All simulations were performed using the library *ccrtm* (Visser, 2021) in *R* programming language (R Core Team, 2021) using the most recent version 4.1.0. Generating a LUT size of 316,800 is an expensive process from a computational standpoint (depending on how much computer resources and time are available this might change). Also, all simulations are independent of each other, meaning simulation of one spectra has no effect on the other, as every simulated spectra is simulated based on a different combination of parameters. These two factors make the generation of such a large LUT good candidate for parallel computation. Therefore, the software packages *doParallel* (Corporation and Weston, 2020) and *foreach* (Microsoft and Weston, 2020) were utilized for parallel computation (using all the available cores) in *R* programming language (R Core Team, 2021). This significantly reduced the computational time. All of the simulations were computed on a Lenovo Thinkpad E480 running under Windows 10 operating system with a

8. Methods

processor Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 2001 Mhz, 4 Core(s), 8 logical processor(s).

8.3 Spectral resampling

The output of INFORM simulations have 1nm spectral resolution within the range of 400nm-2500nm and needs to be spectrally resampled to PRISMA image bands. In this research, the spectral response function of the PRISMA image was used. Band center wavelengths and full width half maximum values were extracted from the PRISMA image metadata and used for spectral resampling. For spectral resampling, the *R* package *hdsar* (Lehnert et al., 2019) was utilized.

8.4 Statistics of simulated data and PRISMA image

Statistical information such as standard deviation and mean were calculated for the simulated (and resampled to PRISMA bands) data and all the pixels of the PRISMA image within the study area. Pixels that are out of the study area boundary were masked out. Then, average spectra in the LUT (synthetic database) and PRISMA image (only study area) were compared to each other. LUT contains 316,800 simulated spectra, the number of pixels within the study area in the PRISMA image is only 95,517. Statistical information were extracted using the libraries in the *tidyverse* package (Wickham et al., 2019) and the plots for visualization were produced using *ggplot2* (Wickham, 2016).

8.5 Gaussian noise

Simulated reflectance data usually do not contain any noise. This is, however, not the case with remote sensing data as they are commonly found to contain various types of noise (Rivera-Caicedo et al., 2017). In this study, 3% Gaussian noise was added to each simulated spectra in the LUT in order to make the simulated data

8.6. Defining training, validation and testing sets

more similar to the real remote sensing data. In order to assess the effect of adding 3% Gaussian noise to the simulated data, one spectra from the LUT and one pixel from the PRISMA image were randomly picked and plotted.

8.6 Defining training, validation and testing sets

The data in the LUT was divided into training, validation and testing sets. Model building and training will be done using only the training set. Validation set will be used to validate the model (e.g. assessing the impact of different hyper-parameters) and the performance of the final model will be tested using the testing set. This step is important because it will allow us to monitor whether the model can generalize to the data (e.g. testing set) it was not trained on.

First, the full data set was shuffled and about 20% of the data was randomly sampled and assigned to validation and testing sets (10% validation, 10% testing sets). Random sampling ensures that there is no any pattern contained in any of the divided data sets.

8.7 Data processing

First, the simulated canopy reflectance in the training data set was normalized and standardized using the Equation (8.1):

$$Band_{n_{scaled}} = \frac{Band_n - \mu_{Band_n}}{\sigma_{Band_n}} \quad (8.1)$$

Here $Band_n$ refers to the reflectance values in the n th simulated band and μ_{Band_n} and σ_{Band_n} are mean and standard deviation of the reflectance values in the n th simulated band. $Band_{n_{scaled}}$ is a transformed version of $Band_n$ that has a mean of 0 and standard deviation of 1. This step ensures that all simulated bands have the same mean and standard deviation.

Data normalization and standardization were only performed using training data set. Mean and standard deviation of the training set were then used to transform the validation and testing data sets.

8.8 Principal Component Analysis (PCA)

Hyperspectral remote sensing data can contain many highly correlated bands. Dimensionality reduction techniques can be efficiently used to reduce the dimensions of hyperspectral remote sensing data. Benefits of reducing the dimensions of simulated data in plant biophysical variable retrieval studies have been demonstrated (Danner et al., 2021; Rivera-Caicedo et al., 2017). In this study, one of the most commonly used DR technique Principal Component Analysis (PCA) was performed. In general, PCA tries to capture as much variation as possible with smaller number variables compared to the original data. PCA produces new variables called Principal Components and each Principal Component (PC) contains certain amount of variation available in the original data. Typically first PC contains the most variation, the second PC contains the second most variation and so on (Bro and Smilde, 2014).

Like in the processing step, PCA was only applied to the training data and the PCA result in the training data was used to transform the validation and testing sets. Cumulative sum of the variations the PCs contain was calculated in order to assess the proportion of the variation that can be explained with fewer variables than the original data (LUT). PCA and data processing performed using the package *recipes* (Kuhn and Wickham, 2021) in *tidymodels* (Kuhn and Wickham, 2020).

8.9 Artificial Neural Networks (ANN)

Considering the fact that we have relatively large number of training examples, non-linear relationship between the input variables (image bands or PCs) and target variables (biophysical and biochemical variables), many different modern and optimization algorithms developed that can learn various types of non-linear relationships, and finally availability of the optimized open source software packages and support, Artificial Neural Network was chosen as a training model in this study.

Deep Neural Networks (DNN) is a specialized name for Artificial Neural Networks. DNN are also called Deep Feedforward Networks. Feedforward network algorithm refers to an algorithm that tries to use the input example data x to learn a function

8.9. Artificial Neural Networks (ANN)

$f()$ that approximates the output variable y by adjusting the weights variable θ (Goodfellow et al., 2016):

$$y = f(x; \theta) \quad (8.2)$$

These algorithms are called networks due to the fact that they usually consist of multiple functions connected to each other with networks. Neural Networks are typically composed of multiple layers, such as input, hidden and output layers. Input layer is typically the training examples and output layer is the target variable (e.g. variable to predict). Hidden layers are in between input and output layers and different functions can be applied to different hidden layers (Goodfellow et al., 2016). For example, the Equation (8.3) shows a function $f(x)$ that is formed by three hidden layers:

$$f(x) = f^{(3)}\left(f^{(2)}\left(f^{(1)}(x)\right)\right) \quad (8.3)$$

In the Equation (8.3), $f^{(1)}$, $f^{(2)}$ and $f^{(3)}$ are the first, second and third layers, respectively. During the learning process, the neural network model is shown the output variables y of corresponding input variables x and the job of the hidden layers is to figure out how to match the target variable y as closely as possible (Goodfellow et al., 2016).

In this research, two neural network models were built. The first model was trained using only 5 PCs as input variables, and the second neural network was trained using simulated 231 PRISMA bands. The aim of building two neural network models is to compare the performance of a neural network using only 5 PCs to a neural network that uses the original 231 simulated PRISMA bands to predict the output variables. This could help assess whether ANN can deal with the multi-collinearity that is available in hyperspectral remote sensing data.

8.9.1 Cost function

Choosing an appropriate cost function is an important part of building neural networks model. Essentially, cost function is what the neural network tries to minimize after each iteration. The cost function mean squared error (MSE) is the most widely used cost function for neural network models when the target variable is continuous (Allaire, 2018).

In this study, MSE was chosen to be the cost function of the neural network models. The Equation (8.4) shows how the cost function MSE is defined:

$$MSE = \frac{1}{N} \sum_{i=1}^n (y_i - f(x_i; \theta))^2 \quad (8.4)$$

In the Equation (8.4), N is the number of examples, y_i is the i th true value and $f(x_i; \theta)$ is the predicted i th value. x_i refers to the input parameter of i th example, and θ typically refers to weight term w and a bias term b .

Apart from MSE, mean absolute error (MAE) was calculated as a metric and monitored during the training. MAE is defined as shown in the Equation (8.5):

$$MAE = \frac{1}{N} \sum_{i=1}^n |y_i - f(x_i; \theta)| \quad (8.5)$$

8.9.2 Optimizer algorithm

In this research, Adam optimizer (Kingma and Ba, 2014) was applied to the neural network model. Before explaining what Adam does, it is important to visit the Stochastic gradient descent (SGD) algorithm.

Stochastic gradient descent is a very important algorithm used to build neural network models. SGD is used to update the learned weight and bias parameters θ . Let's assume that we have a model that tries to minimize the cost function $J(\theta)$

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m L(x^i, y^i, \theta) \quad (8.6)$$

8.9. Artificial Neural Networks (ANN)

where L is the amount of loss for the i th example. Then, we compute the gradient as shown in the Equation (8.7):

$$\nabla_{\theta} J(\theta) = \frac{1}{m} \sum_{i=1}^m \nabla_{\theta} L(x^i, y^i, \theta) \quad (8.7)$$

Finally, we can update the weight and bias parameters within the term θ :

$$\theta \leftarrow \theta - \epsilon \nabla_{\theta} J(\theta) \quad (8.8)$$

In the Equation (8.8) ϵ is called learning rate and needs to be tuned. The learning rate parameter ϵ is a very important hyperparameter for neural network and it is considered to be the most difficult hyperparameter to tune (Goodfellow et al., 2016). Batch gradient descent uses the same idea, but unlike SGD, batch gradient descent makes an update for the whole training set after each iteration (Ruder, 2016).

Usually, some of the directions in the parameter space can have a significant effect on the cost and some of them may not have any effect at all. Adaptive learning algorithms can efficiently solve this problem. One of the most commonly used adaptive learning algorithm is Adam (Kingma and Ba, 2014). Essentially, Adam can be thought as a combination of RMSProp (Hinton, 2012) and Momentum (Polyak, 1964) algorithms with minor differences. In the RMSProp algorithm, exponentially decaying averages are used to mitigate the problem of extreme updates. This helps the model converge more rapidly and less sensitive to extreme cases. The Momentum algorithm uses additional parameter called velocity v apart from learning rate ϵ . And during the gradient descent update, the velocity term v is additionally taken into consideration. The Adam algorithm implements first and second order moment terms. Unlike RMSProp, Adam also implements correction for the bias of first and second order moments (Goodfellow et al., 2016). Goodfellow et al. (2016) can be referred to for more detailed explanation of how the algorithms RMSProp, Momentum and Adam are implemented.

The main drawback of Adam is that now there are more hyperparameters to tune compared to simpler algorithms such as SGD. The Adam optimizer requires

8. Methods

the hyperparameters step size ϵ , exponential decay rates $p1$ and $p2$ and a constant δ . In this study, the suggested default value of 0.001 was used for the step size term ϵ . $p1$ and $p2$ were set to 0.9 and 0.999 respectively (suggested values). The constant term δ is 10^{-8} . Different most commonly used learning rate values (lr) were tested. The optimum learning rate value (specific to this study) was found to be 0.0001.

8.9.3 Mini batch size

Performing gradient update for the whole training set after each iteration can cause the computational cost to increase rapidly when the training data size m is large. Mini batch gradient descent can efficiently overcome this problem (Goodfellow et al., 2016). Mini batch gradient descent makes updates for the mini batch size of n , as opposed to updating gradients after training on the whole training data, with a size of m (Ruder, 2016).

Considering the fact that our training data is relatively large ($m = 250, 120$), we implemented Adam with mini batch size of $n = 512$. This means that, during the neural network training, after each iteration 512 examples from the full training data set ($m = 250, 120$) is randomly sampled. Based on this 512 samples, we then calculate the loss and make gradient updates. When all the examples are sampled from the full training data, this means one epoch of the training phase is complete and the next epoch can start.

8.9.4 Regularization

Regularization is a technique that prevents the problem of overfitting to the training data by adding a penalty term to the cost function. It is an important technique because it can help the trained model to generalize better to the data that it has never seen. In this study, training and validation loss were monitored during the training phase. After some iteration, the model started to fit to the training data too well and the performance of the model on the validation was poorer. Therefore, L^2 norm regularization was applied in order to overcome the problem of overfitting.

8.9. Artificial Neural Networks (ANN)

L^2 regularization is a penalty term that is directly added to the defined cost function. In general, given a cost function J , the L^2 penalty is implemented as shown in the Equation (8.9).

$$\tilde{J}(\theta) = J(\theta) + \lambda w^T w \quad (8.9)$$

In the Equation (8.9) \tilde{J} is called regularized cost function and it tries to minimize the cost J and the regularization term added together. The term λ is L^2 regularization factor and it controls the amount of penalty that is added. For example, when $\lambda = 0$ there is no any constraint added and $\tilde{J} = J$. The larger the λ , the more penalty we add, and therefore the flexibility of the weights to fit to the training data is more limited. The L^2 regularization does not affect bias term b and only regularizes weight term w . λ is another hyperparamter that needs to be tuned. In this study, most commonly used values (0.1, 0.01, 0.001 etc.) were tested and $\lambda = 0.0001$ was found to work well in terms of improving the generalizability of the model to the validation set.

The cost function in this study was updated from the Equation (8.4) to the Equation (8.10):

$$M\tilde{S}E = \frac{1}{N} \sum_{i=1}^n (y_i - f(x_i; \theta))^2 + 0.0001 w^T w \quad (8.10)$$

Here, $M\tilde{S}E$ is the regularized cost function that our model tries to minimize.

8.9.5 Early stopping

Early stopping is a simple technique that monitors the loss and it halts the training when the model starts to overfit to the training data or when it does not improve its accuracy on the validation set over epochs. In this study, validation loss was monitored during the training after each epoch. The parameter *patience* controls when to stop the training and it was set to 50 in this research. This means that if there is no improvement on the validation loss during the last 50 epochs of the training, the training process must stop. This helps with the overfitting

8. Methods

problem as well as it halts the unnecessary training steps (if there is no any improvement after each epoch).

8.9.6 Activation function

In neural networks, the units of the layers are activated by using activation functions. In this study, rectified linear unit activation (ReLU) function was utilized. The units in the hidden layers receive a vector x from the previous layer, and transforms it using a weight term W and a bias term b as follows:

$$z = W^T x + b \quad (8.11)$$

And then, we can use a non-linear activation function ReLU to activate z :

$$g(z) = \max\{0, z\} \quad (8.12)$$

As we can see, ReLU gives a value of 0 when z is smaller than 0, and it returns z otherwise. ReLU is undefined at $z = 0$.

8.9.7 Weight initialization

Neural network training is an iterative process and initial weights need to be given. Goodfellow et al. (2016) indicate that, the initial weight values can sometimes have a large impact on the neural network performance and depending on how the initial weights are defined the neural network model may not converge at all. In this research the He initialization (He et al., 2015) method was applied. This initialization technique was designed to work with NN that make use of the ReLU activation functions. He initialization technique draws initial weights randomly from normal distribution, where the mean $\mu = 0$, and standard deviation $\sigma = \sqrt{\frac{2}{n_l}}$. Here, n_l refers to the number of nodes in the previous layer $l - 1$, that is connected to the nodes in the current layer l .

8.9.8 Processing of input and target variables

In this study, input variables are the 5 PCs for the first neural network model and 231 simulated PRISMA image bands for the second neural network model. Simulated PRISMA image bands were normalized and standardized so that all the bands have the same mean ($\mu = 0$) and standard deviation ($\sigma = 1$) (see Equation (8.1)).

Our target variables are INFORM parameters we want to predict. These are C_{ab} , C_{cw} , C_{cm} and LAI_s . The ranges of these variables are different. For example, minimum and maximum values for C_{ab} are 20 and 60 and for C_{cw} these values are 0.0035 and 0.035. The cost function with these output variables would be largely driven by variables that have higher magnitudes and interpreting the loss after each epoch would be difficult. Therefore, the output variables were normalized using the Equation (8.13).

$$Y_{scaled} = \frac{Y - \mu_Y}{\sigma_Y} \quad (8.13)$$

Here, Y is the output variable, μ_Y is the mean and σ_Y is the standard deviation of the output variable Y .

8.9.9 Number of epochs

Number of epochs is another hyperparameter and needs to be specified by the user. In this study, number of epochs was set to a relatively large number, 5000. However, as the early stopping technique was also used, setting the number of epochs to a large value does not guarantee that the training phase will continue for 5,000 epochs. Setting number of epochs to a high number ensures that the model can learn properly as long as the validation loss decreases. If the validation loss does not decrease for the last 50 epochs (specific to this study), the training will stop at any number of epoch.

8.9.10 Neural Network Architecture design

Designing NN architecture is an important step of building neural network models. Here, number of hidden layers and number of units in each layers are hyperparameters and they need to be tuned. In this research, several neural network architectures were tested. We started with the simplest architecture (1 input layer, 1 hidden layer with 1 hidden unit and output layer) and gradually increased the depth (1, 2, 3) and width (2^1 , 2^2 , 2^3 , 2^4 etc.) of the NN to assess the impact of the design on the performance of the model. In general, it was observed that the more complicated the neural network architecture is, the better it performs. Increasing the number of hidden layers significantly increased the generalizability of the model to the testing data when the number of hidden units were kept constant at each layer. Optimal NN architecture was chosen based on the time it took for the model to converge, training, validation and testing accuracy.

The Figure 8.1 shows the chosen neural network architecture for the first model (where PCs were used as inputs):

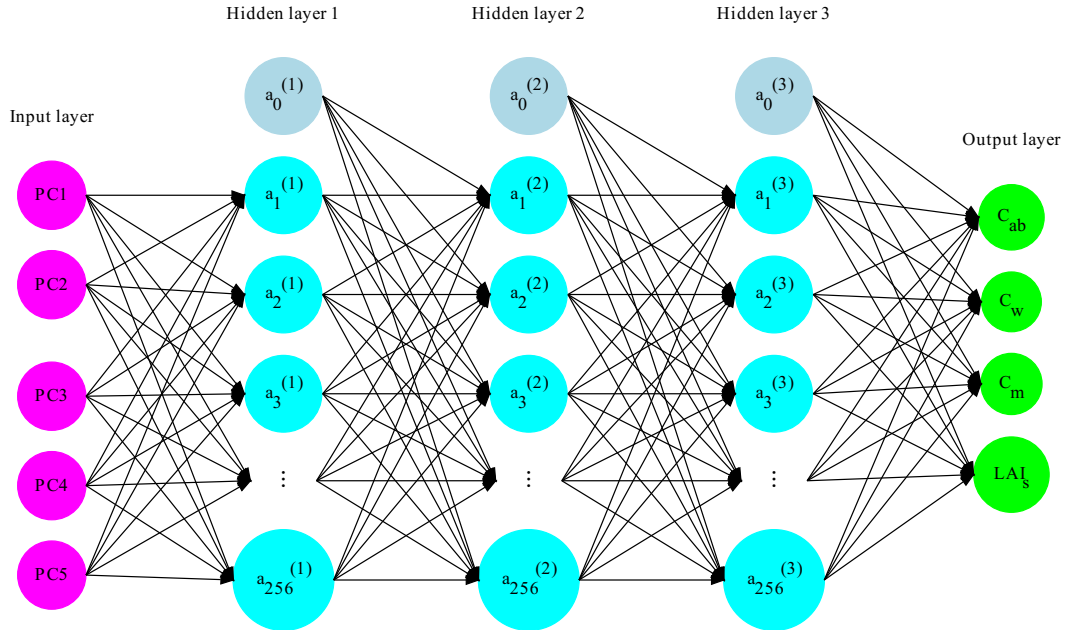


Figure 8.1: Architecture of the Neural Network with Principal Components as inputs

In the Figure 8.1, a_n^l refers to the n th unit in the l th hidden layer. Among all the tested neural network architectures when PCs as an input layer were used,

8.9. Artificial Neural Networks (ANN)

this architecture yielded the best results in terms of the time it took the model to converge, the minimized loss, performance on training, validation and testing sets. However, it is important to note that a different neural network architecture that was not tested in this study could yield better results. Also, slightly simpler architecture could have yielded a similar result if it was trained much longer. But, in this study, simpler architectures (e.g. NN with 1 or 2 hidden layers) did not perform as well as the chosen architecture (Figure 8.1) when it was trained for the same number of epochs. It should also be noted that, NN with deeper than 3 hidden layers were not tested in this study.

The Figure 8.2 shows the chosen NN architecture for the NN model where simulated 231 PRISMA image bands were used as input.

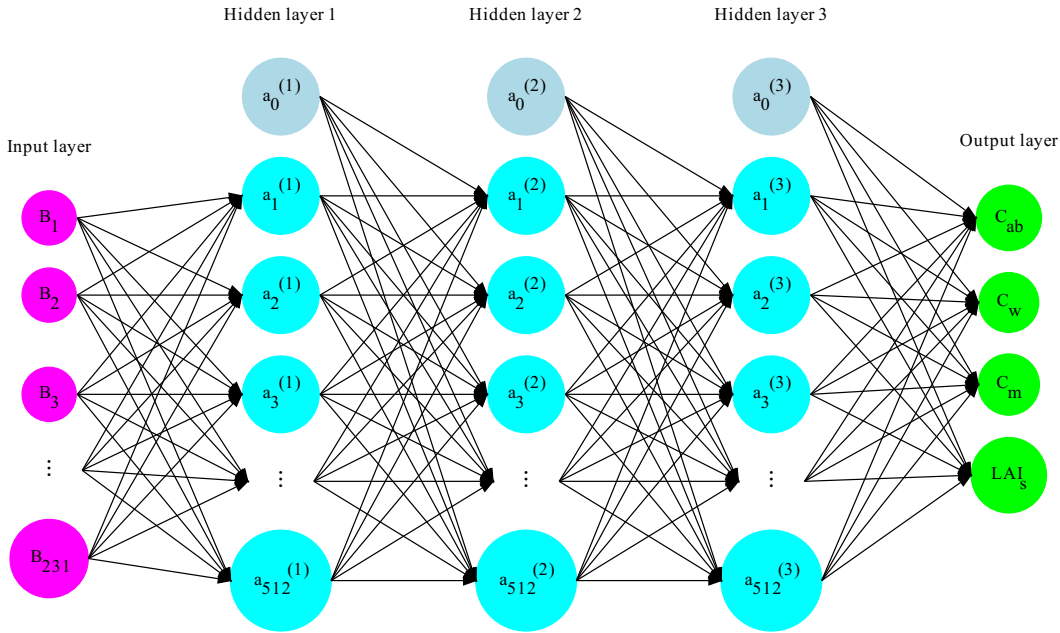


Figure 8.2: Architecture of the Neural Network with simulated PRISMA image bands as inputs

In the Figure 8.2, B_n refers to the simulated canopy reflectance for the n th PRISMA image band and a_n^l refers to the n th unit in the l th hidden layer. When simulated PRISMA bands were used as an input layer, varying number of hidden layers and number of units had a much more noticeable change. In general, the more complex architectures yielded better results. And, when the NN model with

8. Methods

less hidden layers were run for the same or more number of epochs they yielded significantly poorer results. This may potentially indicate that even the more complicated NN architecture could perform better than the chosen architecture.

All of the Neural Network building, testing and training processes were implemented using keras (Allaire and Chollet, 2021) in tensorflow API (Allaire and Tang, 2021).

8.10 Final prediction and map retrieval

Finally, the best performing NN model was used to retrieve C_{ab} , C_w , C_m and LAI_s maps for the National Park Hunsrück-Hochwald. Target variables (INFORM parameters) were normalized and standardized using the same Equation (8.13) before training NN models. This means that when we apply the trained NN model to make a prediction on the PRISMA image we get scaled predictions. Therefore, it is necessary to “un-scale” the predictions back. To achieve this we can use the Equation (8.13) and scaling (σ) and centering (μ) factors of training output variables as follows:

$$Y_{unscaled} = (Y_{predicted_{scaled}} \cdot \sigma_{Y_{train}}) + \mu_{Y_{train}} \quad (8.14)$$

In the Equation (8.14) $Y_{unscaled}$ is the final predicted value, $Y_{predicted_{scaled}}$ is the scaled version of the prediction, $\sigma_{Y_{train}}$ is the standard deviation and $\mu_{Y_{train}}$ is the mean of the corresponding target variable in the training set.

Each pixel in the retrieved maps shows the predicted value of the corresponding INFORM parameter. The final retrieved maps were saved as a 4-band raster file (e.g. ENVI and Tiff formats) and can be visualized and analysed further.

8.11 GitHub repository of the study

All the scripts that were written in this study for data handling, model simulation and machine learning training are available in the public repository of the author at https://github.com/zavud/msc_thesis.

9 Results

9.1 Local sensitivity analysis

Figure 9.1 shows the result of sensitivity analysis. Chlorophyll content (C_{ab}) appears to almost exclusively impact the visible spectra. In general, lower C_{ab} appears to increase reflectance in the visible spectra. Some effect can also be noticed in the red-edge, but there is not a significant effect of varying C_{ab} on the simulated spectra within the near-infrared (NIR) and short wave infrared (SWIR) (Figure 9.1.a). Conversely, equivalent water thickness (C_w) (Figure 9.1.b) and leaf dry matter content (C_m) (Figure 9.1.c) both have large effects on simulated spectra within the NIR and SWIR but, no significant effect within the visible spectra. Lower C_w result in higher and lower C_m result in higher simulated canopy reflectance, mainly within the NIR and SWIR. Leaf Area Index (single) (LAI_s) (Figure 9.1.d), Crown diameter (CD) (Figure 9.1.e)) and Stem density (SD) (Figure 9.1.f) all have noticeable effect on the simulated canopy reflectance almost all over the spectra. Increased LAI_s result in decreased reflectance. The effects of CD and SD on the simulated canopy reflectance is relatively more complex. For example, lower CD values seem to result in higher reflectance within the visible and SWIR spectra. However, higher values of CD can also result in higher reflectance mainly within the red-edge and some part of NIR. Increased SD values appear to result in increased simulated reflectance within the NIR, and decreased reflectance elsewhere.

9. Results

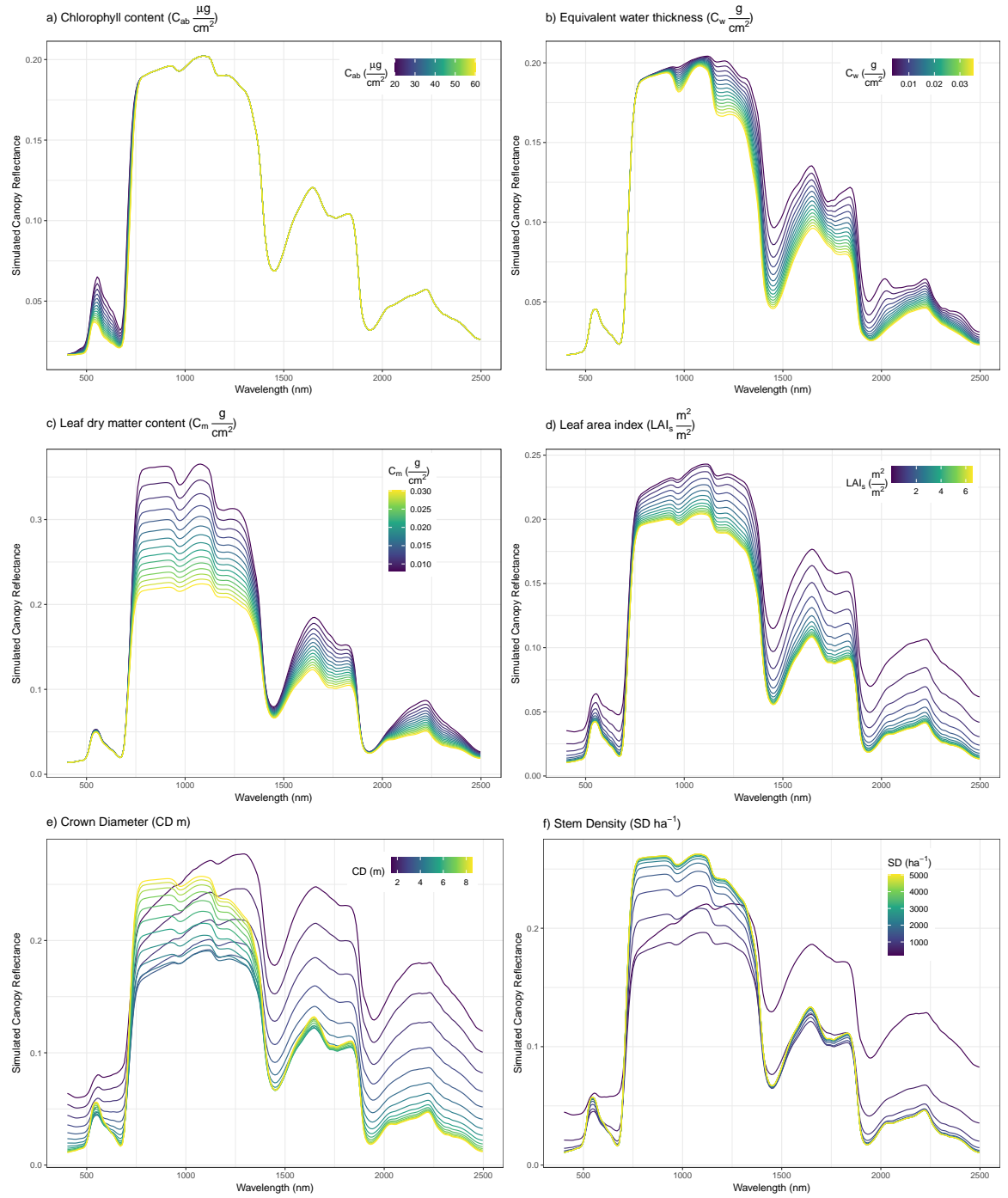


Figure 9.1: Effects of varying the chosen parameters on the simulated spectra

9.2 RTM simulation (INFORM)

Synthetic canopy reflectance data set were produced and stored in a LUT containing all 316,800 simulations. In this research, LUT was defined as a matrix. Each row of this matrix is a different simulated spectra and columns are simulated reflectance of wavelengths with the range of 400nm-2500nm with 1nm spectral resolution and 6 additional columns containing values of the corresponding variables C_{ab} , C_w , C_m , LAI_s , CD and SD that were used for each simulation. Hence the dimensions of the LUT matrix is 316,800 rows (number of simulations) by 2107 columns (2101 simulated “bands” + 6 INFORM variables):

$$\begin{bmatrix} 400nm_1 & \dots & 2500nm_1 & C_{ab1} & C_{w1} & C_{m1} & LAIs_1 & CD_1 & SD_1 \\ 400nm_2 & \dots & 2500nm_2 & C_{ab2} & C_{w2} & C_{m2} & LAIs_2 & CD_2 & SD_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 400nm_{316,800} & \dots & 2500nm_{316,800} & C_{ab316,800} & C_{w316,800} & C_{m316,800} & LAIs_{316,800} & CD_{316,800} & SD_{316,800} \end{bmatrix}$$

In the LUT matrix above, $400nm_n, \dots, 2500nm_n$ refer to the simulated reflectance for the corresponding wavelength in the simulation number n . C_{ab_n} , C_{w_n} , C_{m_n} , LAI_{s_n} , CD_n and SD_n are values of the INFORM parameters that were used in the n th simulation.

9.3 Spectral resampling

The output of INFORM simulations were resampled to 231 PRISMA bands. The LUT matrix was used for spectral resampling and the resulting matrix has a dimension of 316,800 rows (number of simulations) by 237 columns (231 PRISMA image bands + 6 INFORM variables):

$$\begin{bmatrix} Band1_1 & \dots & Band231_1 & C_{ab1} & C_{w1} & C_{m1} & LAIs_1 & CD_1 & SD_1 \\ Band1_2 & \dots & Band231_2 & C_{ab2} & C_{w2} & C_{m2} & LAIs_2 & CD_2 & SD_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Band1_{316,800} & \dots & Band231_{316,800} & C_{ab316,800} & C_{w316,800} & C_{m316,800} & LAIs_{316,800} & CD_{316,800} & SD_{316,800} \end{bmatrix}$$

In this matrix, $Band1_n, \dots, Band231_n$ correspond to the simulated reflectance for the corresponding image band in the simulation number n . C_{ab_n} , C_{w_n} , C_{m_n} , LAI_{s_n} , CD_n and SD_n refer to the values of the INFORM parameters that were used in the n th simulation.

9. Results

9.4 Statistics of simulated data and PRISMA image

The Figure 9.2 shows statistical information (mean and mean \pm standard deviation) calculated from the LUT and PRISMA image:

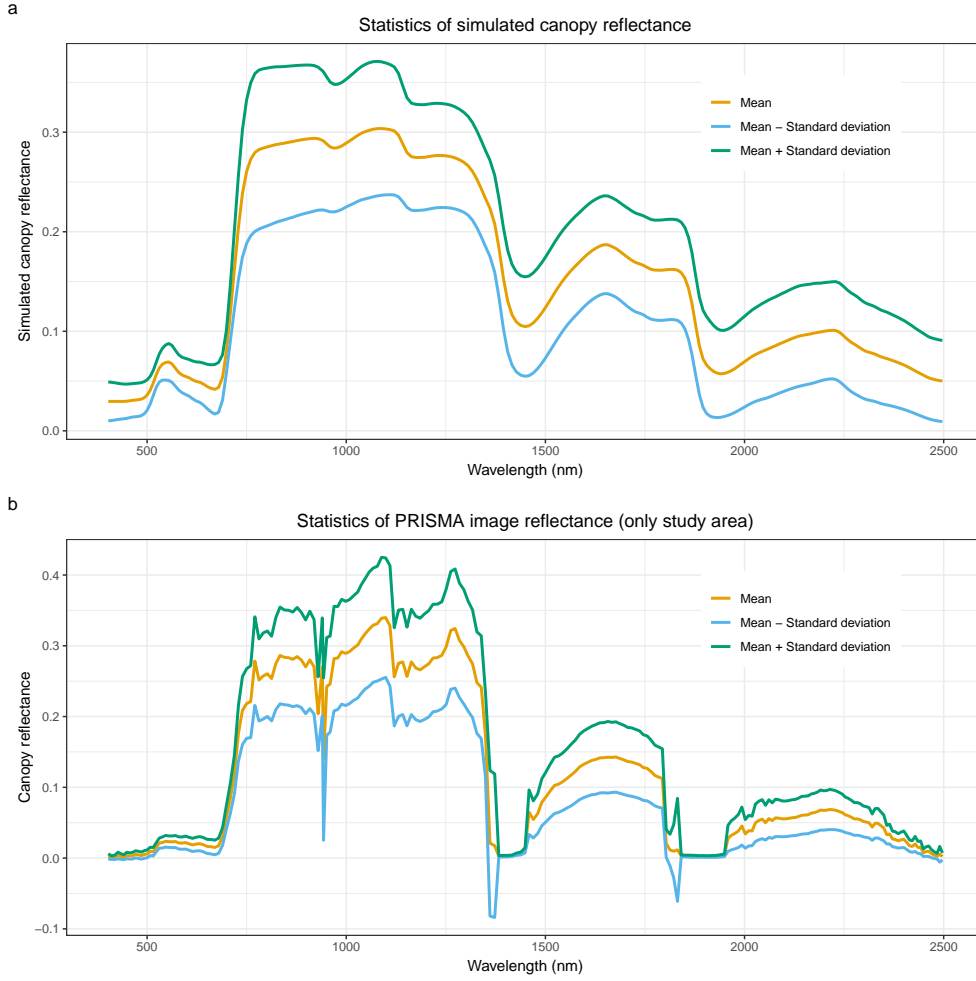


Figure 9.2: Mean and mean \pm standard deviation in the a) LUT and b) PRISMA image

Mean and standard deviation within the LUT are much smoother compared to mean and standard deviation within the PRISMA image spectra. This is due to the fact that INFORM model does not add noise during the simulation which can commonly exist in remote sensing images. There is a noticeable amount of noise in the PRISMA image spectra. Some of the noise in the image spectra could potentially be due to the fact that the PRISMA image contained cloud and shadow

9.4. Statistics of simulated data and PRISMA image

within the study area and although most of the cloud and shadow pixels were masked, the nearby pixels could still be affected.

The Figure 9.3 shows the difference between averaged reflectance within the simulated database (LUT) and PRISMA image.

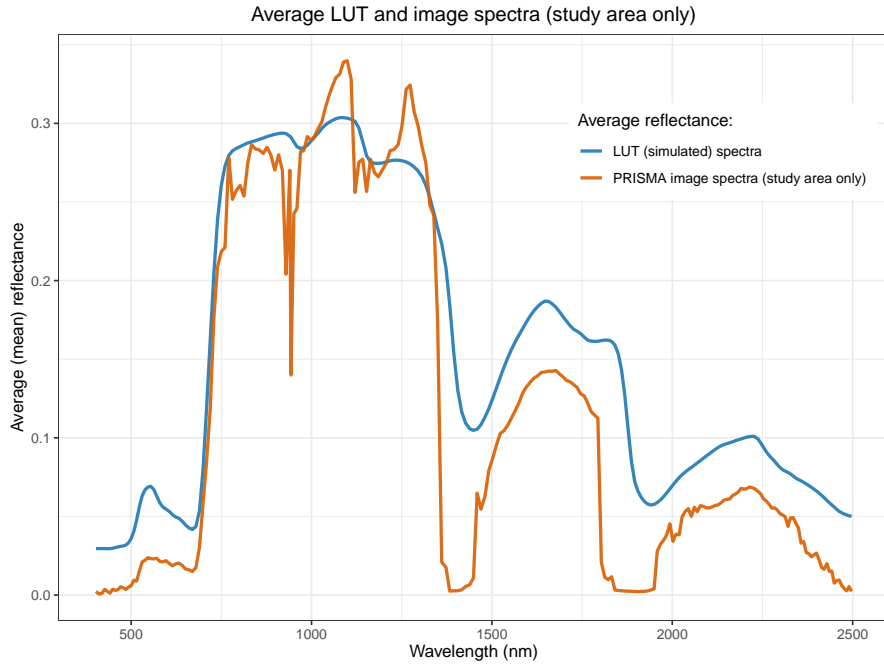


Figure 9.3: Difference between averaged LUT and PRISMA image reflectance

The LUT appears to have higher average reflectance within the visible spectra compared to the PRISMA image spectra. Differences within the water absorption bands can also be clearly seen. There is relatively good agreement within the NIR spectrum.

9.5 Gaussian noise

The Figure 9.4 shows the effect of adding 3% Gaussian noise to the simulated data.

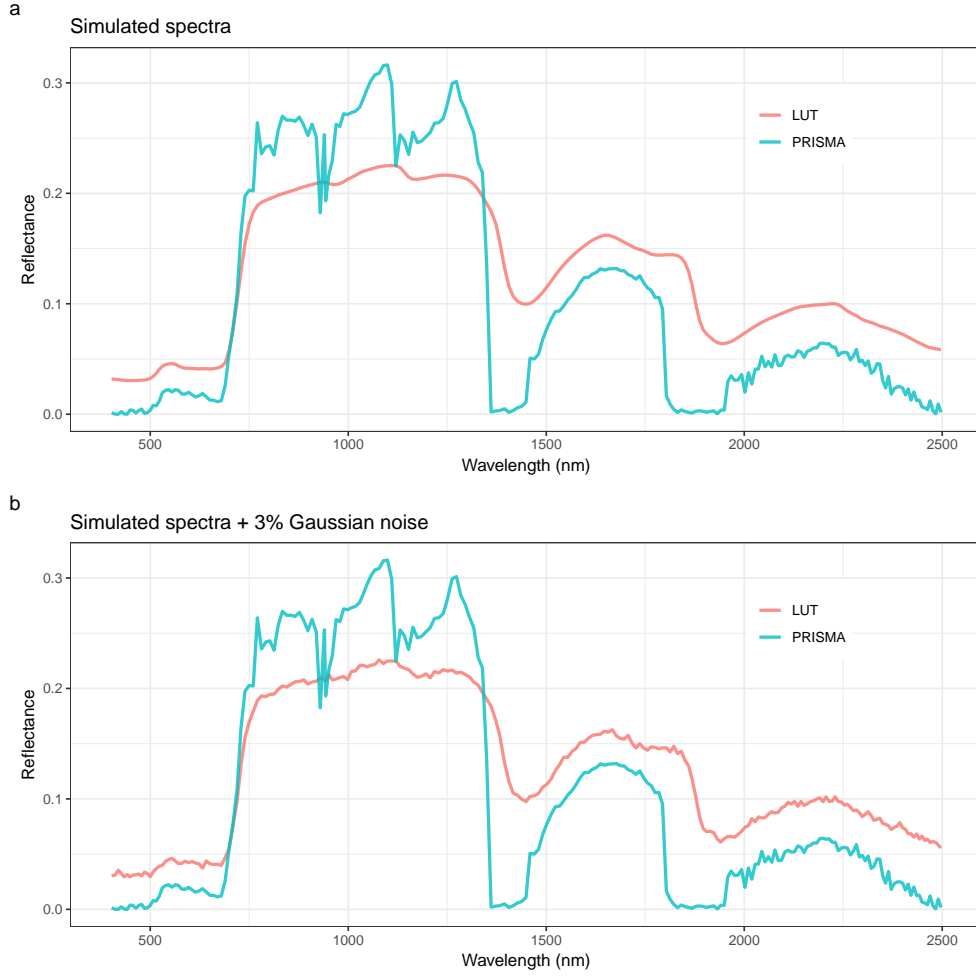


Figure 9.4: Effect of adding 3% Gaussian noise to the simulated spectra. The randomly chosen pixel from the PRISMA data was plotted to illustrate the noise typically found in the image

The Figure 9.4.a shows a simulated spectra that seems perfectly smooth. However, after adding 3% Gaussian noise, the simulated spectra is not as smooth anymore and contains random noise all over the whole spectra (Figure 9.4.b). This also makes the simulated spectra more similar to the pixel extracted from the PRISMA image.

9.6 Principal Component Analysis (PCA)

The result of PCA show that most of the variation in the simulated data can be explained by much fewer variables (PCs):

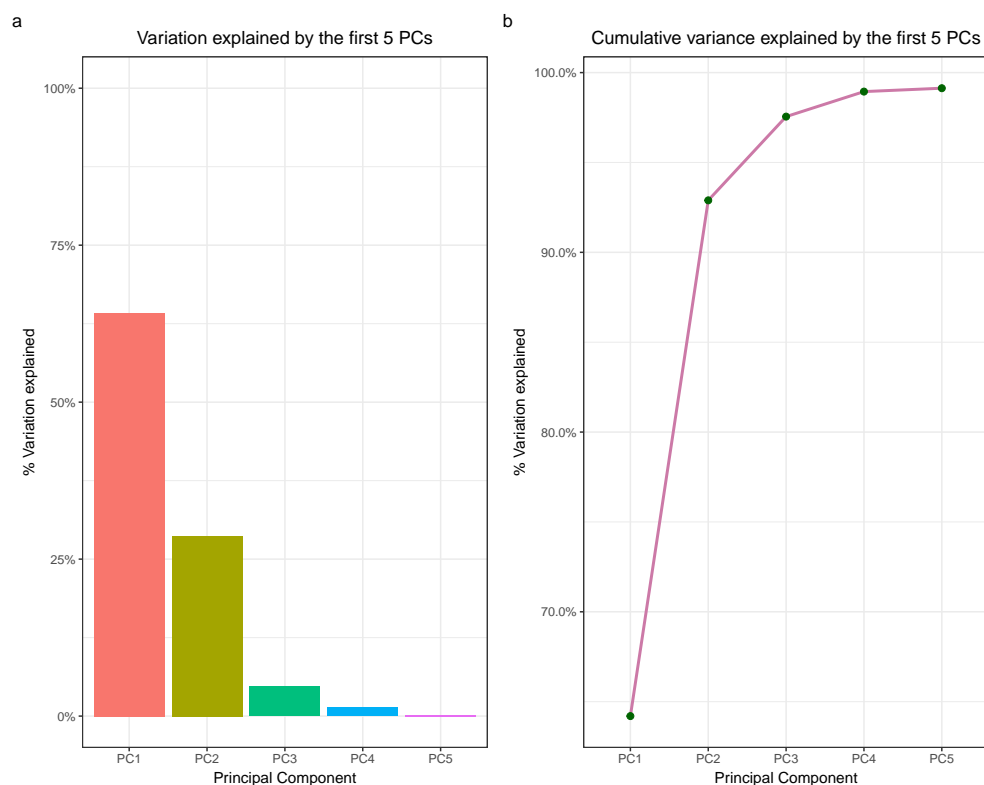


Figure 9.5: Principal Component Analysis: a) Screeplot, b) Cumulative variance explained by the first 5 PCs

The Figure 9.5.a shows the screeplot of the PCA result. The first PC explains the most of the variation and together with the next 4 PCs we can capture more than 99% of the variation that is present in the original data (Figure 9.5.b). This, once again shows the multicollinearity problem within the hyperspectral data.

9.7 Artificial Neural Networks (ANN)

9.7.1 Training

The Figure 9.6 shows the loss, MSE as a function of epochs for the first NN model.

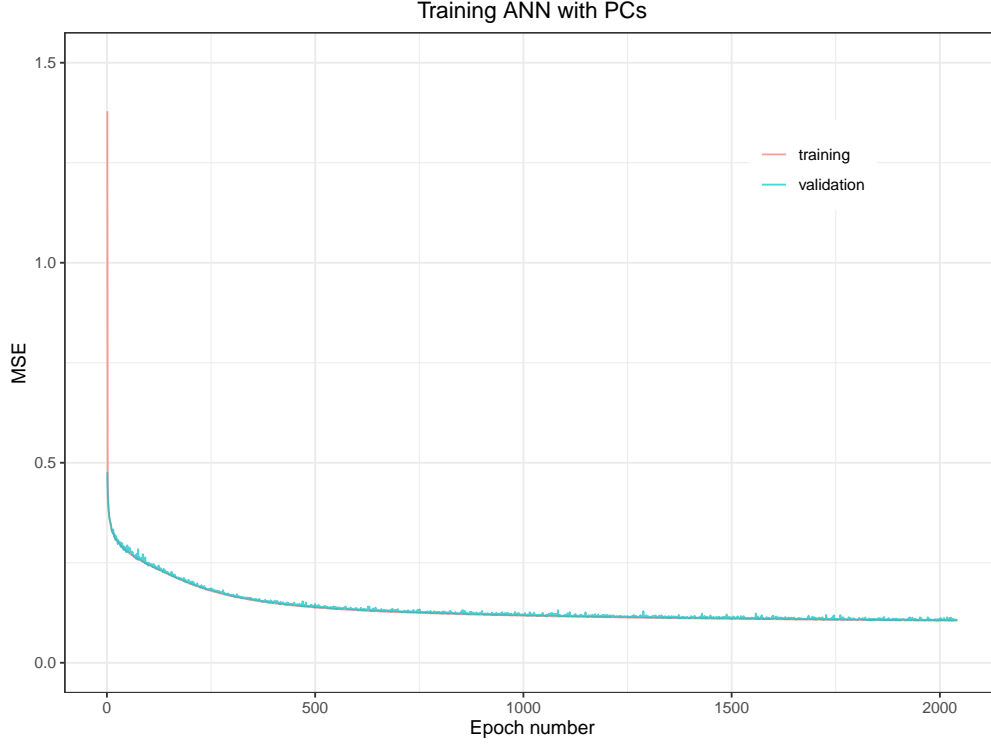


Figure 9.6: Training history of ANN with PCs

It took the NN (when PCs were used as input) more than 2000 epochs to converge to the potential global optima. The loss appears to decrease relatively rapidly during the first 500 epochs. After the epoch number 500 the loss decreases slowly. This may indicate that some of the non-linear relationships between the simulated spectral features and the chosen plant parameters are relatively easy to learn, but the NN needs more training to learn more complicated relationships. The final MSE and MAE this NN achieved when it was evaluated on the testing set was 0.1071568 and 0.1739102 respectively (Table 9.1). These values are a joint loss of all the target parameters (INFORM parameters).

The Figure 9.7 shows the training history of the NN where simulated 231 PRISMA bands were used as an input layer.

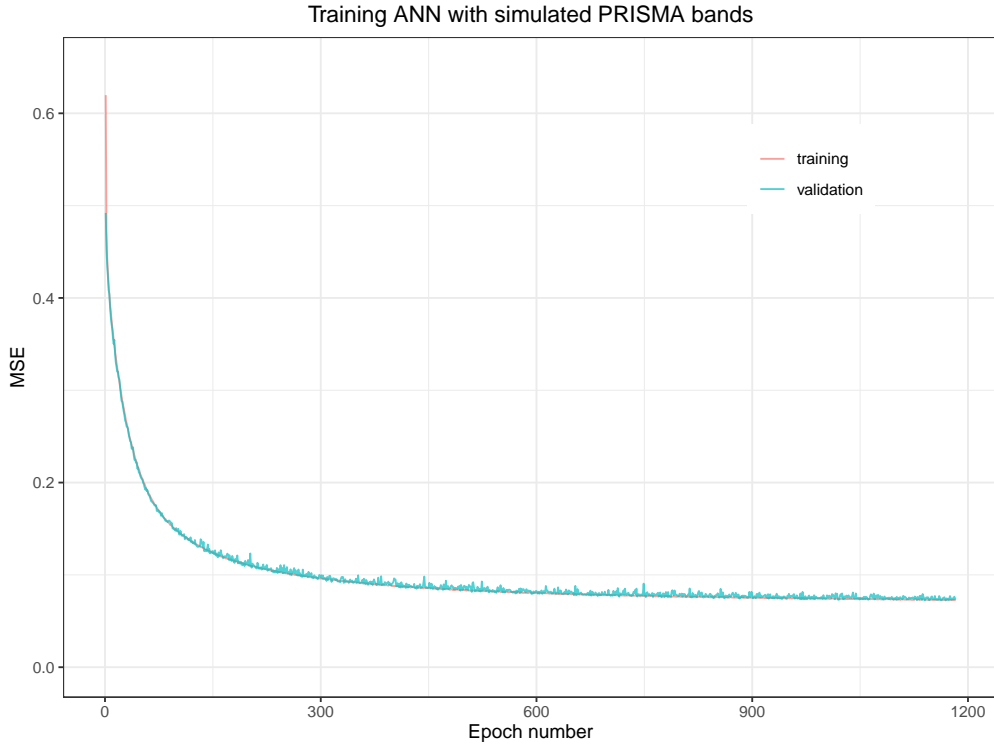


Figure 9.7: Training history of ANN with simulated PRISMA bands

In this model, less than 1200 epochs of training was enough for the NN to converge. Also, the training, validation and most importantly, testing losses were lower (compared to the first NN) when the simulated PRISMA bands were used as predictors (Table 9.1).

The lines that show training and validation loss over epochs (Figures 9.6 and 9.7) are close to each other and show more or less similar pattern of decrease. This is a sign that there is no a serious overfitting problem during the training. Before adding regularization term to the cost function, however, there were signs of overfitting during the training.

9. Results

The Table 9.1 summarizes the final results of the ANNs.

Table 9.1: Final results of the trained NNs

Dataset	Metric	NN with PCs	NN with sim. bands
Train	MSE	0.1049012	0.07522584
Train	MAE	0.1710693	0.13174930
Validation	MSE	0.1068904	0.07499193
Validation	MAE	0.1740899	0.13256425
Test	MSE	0.1071568	0.07670474
Test	MAE	0.1739102	0.13459726

This result indicates that, when designing the NN architecture properly and giving reasonable amount of hidden layers and units, the NN model can be trained with simulated bands and deal with multicollinearity very efficiently.

9.7.2 Evaluation of the NN models on the testing set

The Figure 9.8 shows the scatterplots of the predicted parameters versus modelled parameters for the PCA based NN (first column or the plots on the left side) and NN with simulated PRISMA bands (the second column or the plots on the right). For easier visualization of the scatter plots only 1500 data points were randomly sampled from the testing data set (full testing data could be too large to visualize). Also, there was an overplotting problem in the original scatter plots due to many similar points laying on top of each other. Therefore, the “jitter” technique was used, which adds a very small random noise to each point (location wise) that makes visualization easier (Wickham and Grolemund, 2016; Wickham, 2016). The blue diagonal line shows the 1:1 line that has an intercept 0 and slope 1.

9.7. Artificial Neural Networks (ANN)

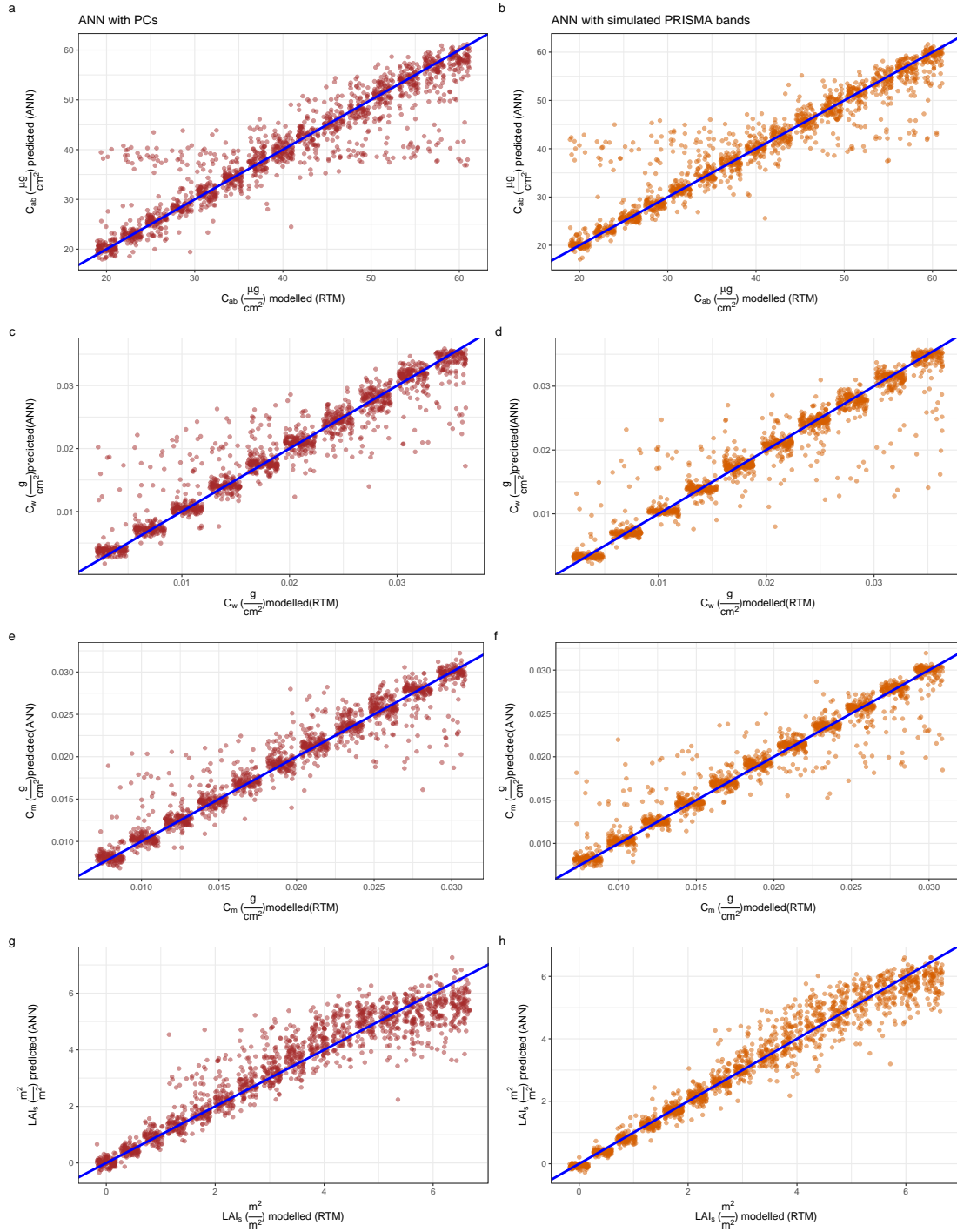


Figure 9.8: Predicted versus modelled RTM parameters for the PCA based NN (1st column) and NN with simulated PRISMA bands (2nd column)

In general, both of the NN models appears to perform well on the testing set that it was not trained on. The effect of adding random Gaussian noise can clearly be seen. The points that are relatively far from the diagonal line are results of

9. Results

adding 3% Gaussian noise. This was not the case on the NN model that was trained with simulated data that did not receive any amount of noise. The parameter C_{ab} appears to be impacted by the noise more significantly compared to other parameters. This is potentially due to the fact that C_{ab} has a very specialized impact on the simulated reflectance, almost exclusively on the visible spectra (Figure 9.1.a). Unlike C_{ab} , the other parameters had impact on more bands (Figure 9.1) and this might have helped the NN learn the relationship more easily for these parameters despite the existence of noise.

Although overall pattern seems very similar, accuracy of the NN model that was trained with simulated image bands is slightly better for all the parameters. Superiority of this NN model is more noticeable for the parameter LAI_s (Figure 9.8.h). The Table 9.2 shows the differences between the results of the two models on the testing set parameters.

Table 9.2: Differences between the performances of the trained 2 NN models on the testing set parameters

Parameter	R^2 (PCs)*	RMSE (PCs)*	R^2 (Bands)†	RMSE (Bands)†
C_{ab}	0.8733358	4.37369 ($\frac{\mu g}{cm^2}$)	0.8911385	4.064 ($\frac{\mu g}{cm^2}$)
C_{cw}	0.9246098	0.002765111 ($\frac{g}{cm^2}$)	0.9396872	0.00252202 ($\frac{g}{cm^2}$)
C_{cm}	0.9341707	0.001798028 ($\frac{g}{cm^2}$)	0.9428488	0.001702862 ($\frac{g}{cm^2}$)
LAI_s	0.896475	0.641572 ($\frac{m^2}{m^2}$)	0.9513208	0.443341 ($\frac{m^2}{m^2}$)

* PCs - refers to the result of NN with Principal Components

† Bands - refers the result of NN with simulated PRISMA bands

The NN trained with simulated 231 PRISMA bands predicts INFORM parameters C_{ab} , C_w and C_m with slightly better R^2 and $RMSE$, compared to the PCA based NN model. This NN model predicts LAI_s with much better accuracy than the PCA based NN. This could potentially be due to the fact that LAI_s has an important effect within the full spectra (see sensitivity subplot Figure 9.1.d) and the 5 PCs may not properly represent the full relationship between the plant parameter LAI_s and spectra.

9.8 Final prediction and map retrieval

The NN that was trained with 231 simulated PRISMA bands was used for map retrieval as the final optimal model. Four different maps for the parameters C_{ab} , C_w , C_m and LAI_s were produced as raster images. Each pixel in the retrieved maps is the predicted value of the corresponding INFORM parameter.

9.8.1 Retrieval of C_{ab} map

The Figure 9.9 shows the predicted map of the parameter C_{ab} .

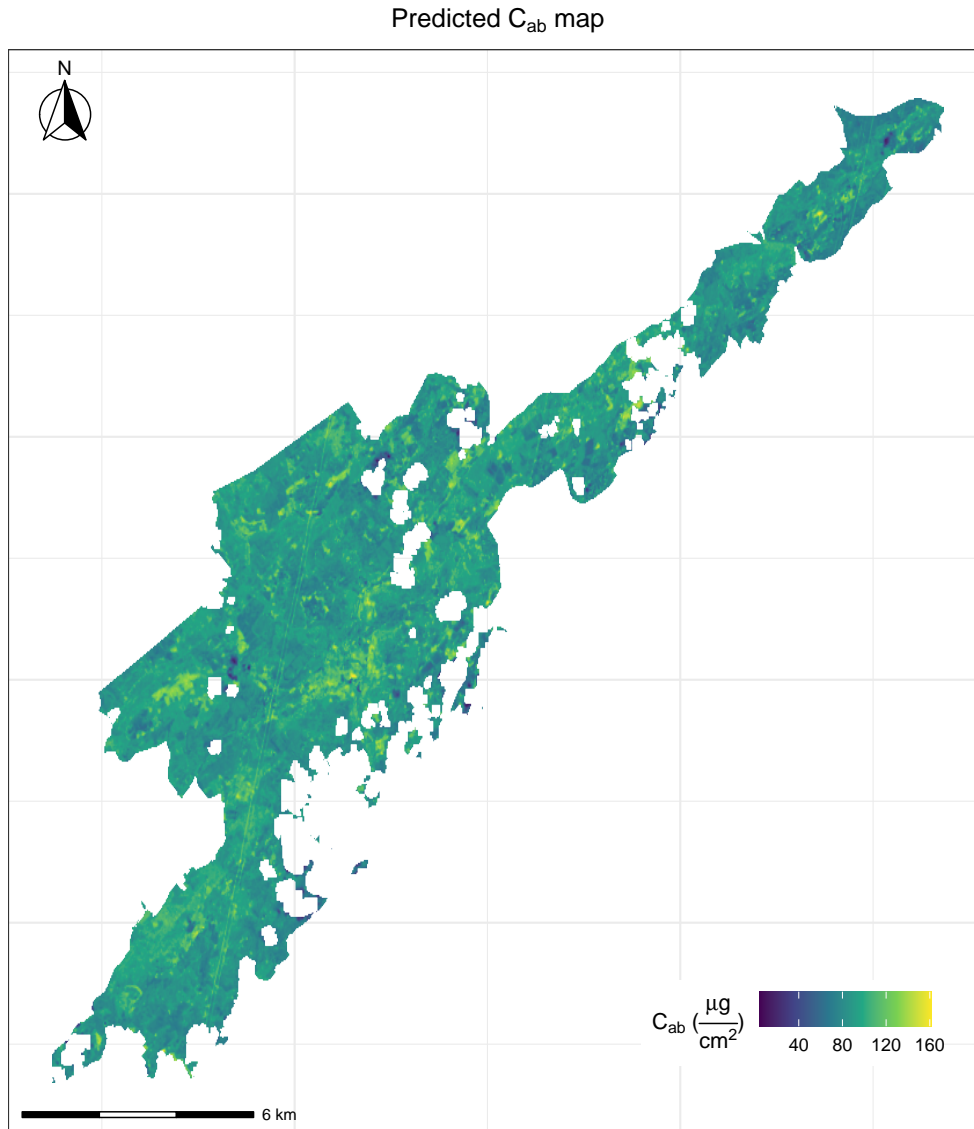


Figure 9.9: Predicted map of the parameter C_{ab}

9. Results

Although there are some pixels showing values that are similar to the range that was used during the RTM simulation, some pixels show higher C_{ab} values than the values that were used for simulation. The reason why the trained NN predicts higher values for the PRISMA image compared to the value range that was used in the simulation can be explained by the fact that the parameter C_{ab} only have a significant impact on the visible spectra (see Figure 9.1.a) and average reflectance in the visible spectra of the PRISMA image is lower compared to the average reflectance of the visible spectra in the LUT (simulated database) (see Figure 9.3). Specifically, the sensitivity analysis showed that the higher the C_{ab} value, the lower the reflectance in the visible spectra will be (Figure 9.1.a). This means that it is expected to get higher predicted C_{ab} values for the PRISMA image.

9.8.2 Retrieval of C_w map

The Figure 9.10 shows the predicted map of the parameter C_w .

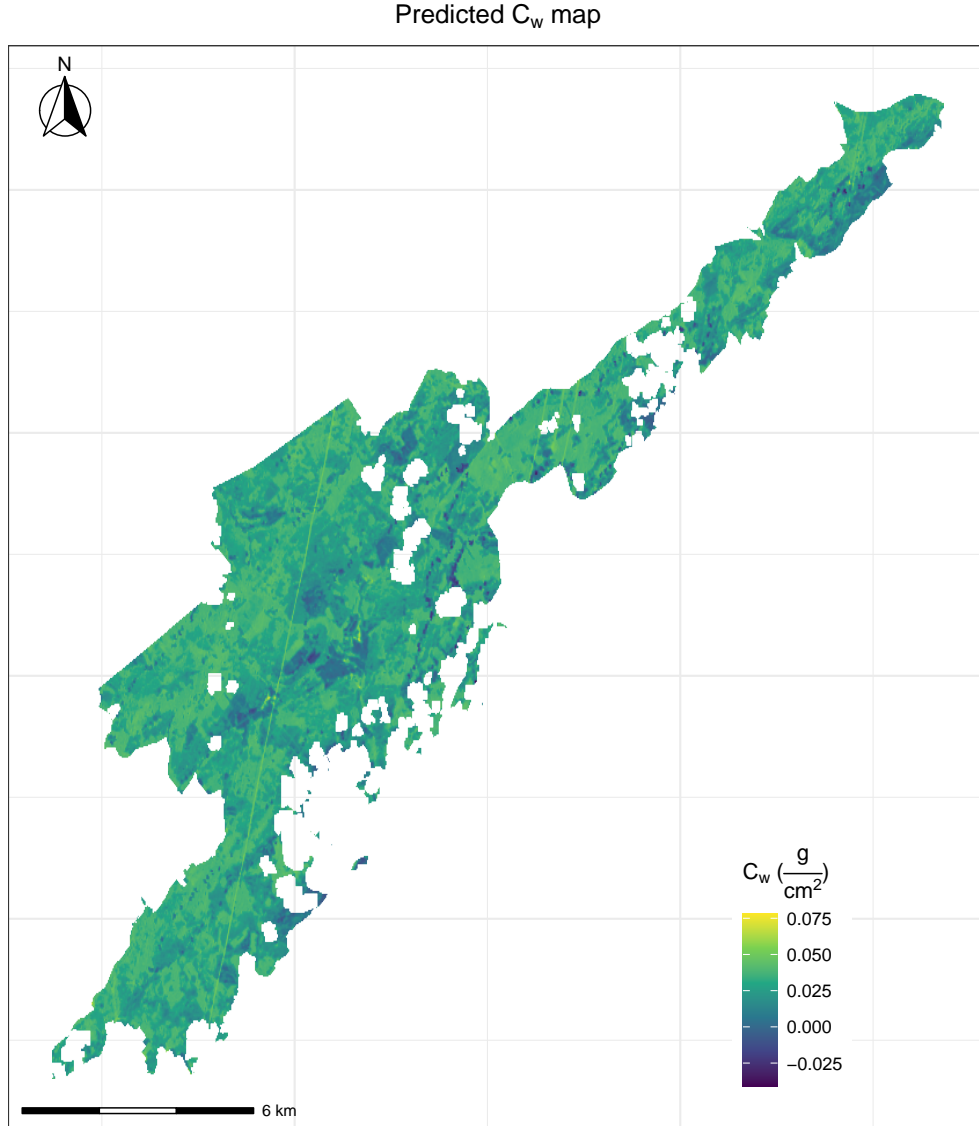


Figure 9.10: Predicted map of the parameter C_w

Majority of the pixels show predicted values that are within the same range of the values that were used during the simulation. The parameter C_w has a large effect on the NIR spectrum (Figure 9.1.b) and the average reflectance in the NIR region within both LUT and PRISMA image are relatively close to each other (Figure 9.3). This could explain why the range of the predicted C_w values is similar to the C_w range within the LUT. However, there is also a large impact of C_w on the

9. Results

SWIR and this region of the average simulated spectra is higher than the average reflectance of SWIR in the PRISMA image spectra.

9.8.3 Retrieval of C_m map

The Figure 9.11 shows the predicted map of the parameter C_m .

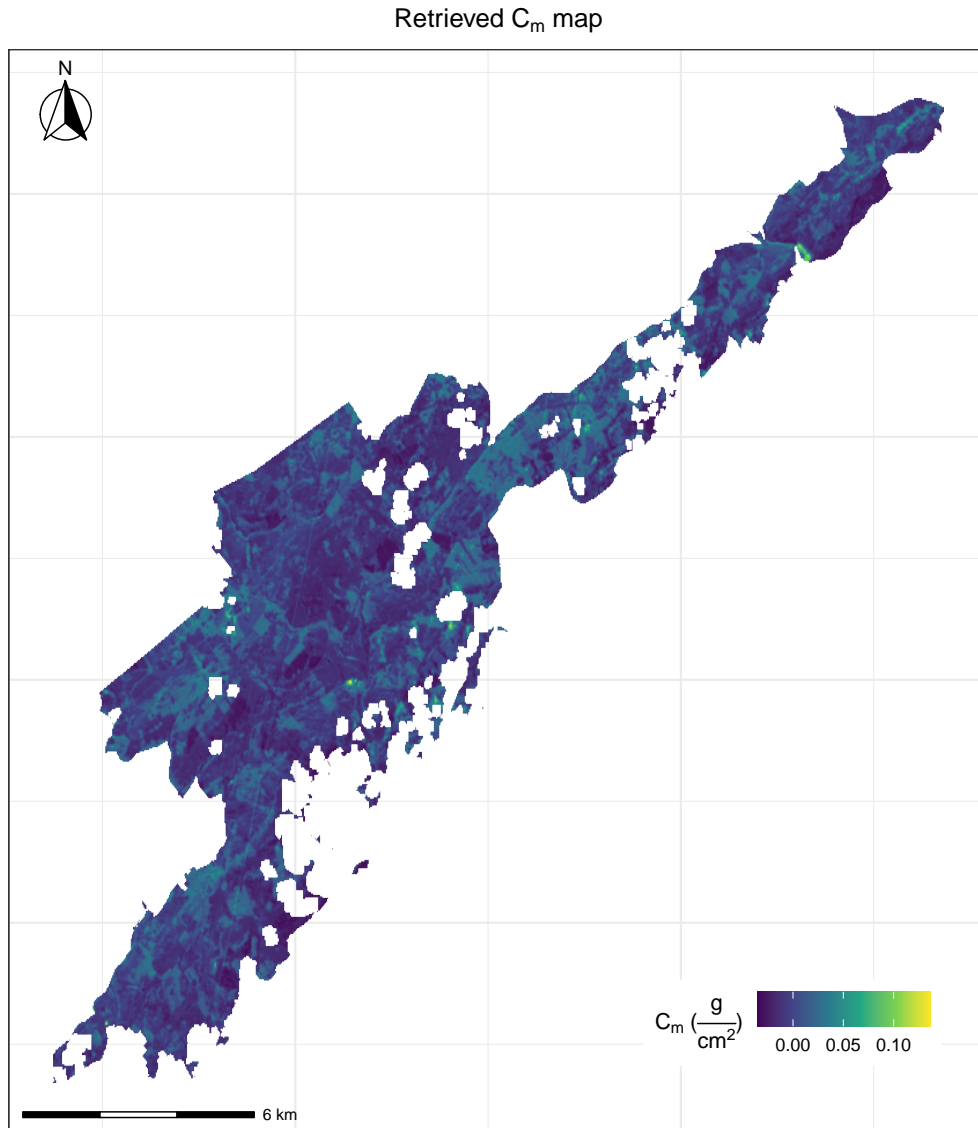


Figure 9.11: Predicted map of the parameter C_m

The predicted C_m map shows that some of the pixels in the predicted map is smaller than 0. The sensitivity analysis showed that increased C_m values decreases the reflectance over NIR and SWIR (Figure 9.1.c). In average, the PRISMA image

pixels have lower reflectance within the SWIR region of the spectra and this is the reason why the trained NN predicts low values C_m for some of the pixels.

9.8.4 Retrieval of LAI_s map

The Figure 9.12 shows the predicted map of the parameter LAI_s .

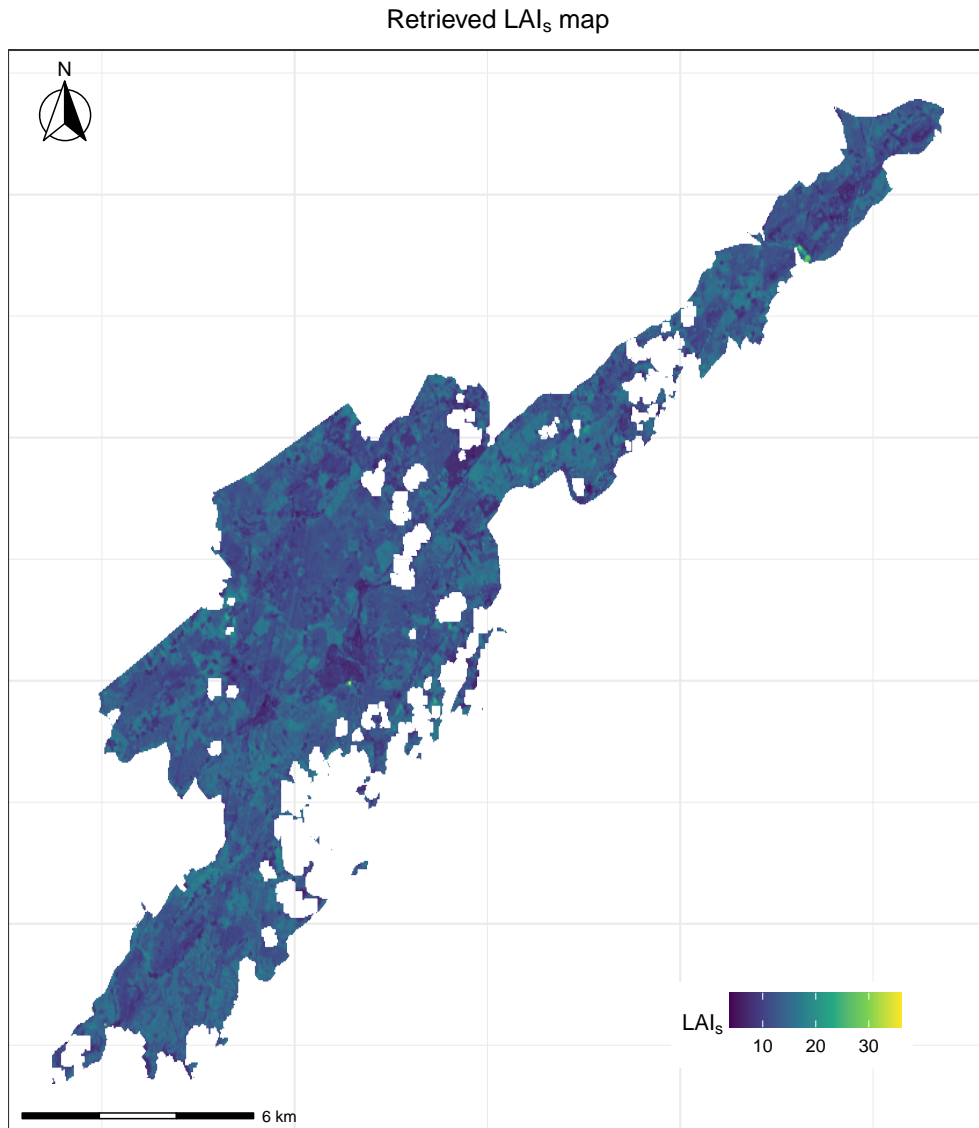


Figure 9.12: Predicted map of the parameter LAI_s

We can see that most of the pixels in the predicted LAI_s map show the value range of 10-15. We can also detect some pixels that have larger and smaller values than this range.

10 Discussion

This section discusses the results, limitations, improvements and a potential follow-up study.

10.1 Artificial Neural Networks for predicting plant traits

This study showed that, given enough hidden layers and neurons, artificial neural networks can efficiently deal with high multi-collinearity that exists in hyperspectral remote sensing data. This study also showed the applicability of deep neural networks to predict plant biophysical and biochemical parameters. For all the INFORM parameters, the NN that was trained with simulated 231 PRISMA bands had better results compared to the NN that was trained with 5 principal components capturing most of the variation in the original data set. Training deep neural networks with full simulated image bands to predict plant biophysical and biochemical parameters that have effect within the whole spectra (e.g. LAI) could specifically be beneficial. However, it should also be noted that, depending on the availability of the computational resources and time, using a dimensionality reduction technique such as PCA may be preferred. This is because training NN with hundreds of predictors (e.g. hyperspectral image bands) usually needs more time to run compared to a PCA based NN that is trained with only a few or several predictors (principal components). However, in this study, for the NN that was trained with hyperspectral image bands it took less than 1200 epochs to converge, while the PCA based NN needed to run more than 2000 epochs. This could potentially indicate that for the NN it may be easier to learn the relationships between the plant traits and simulated spectra when all the bands are available. The design of the NN had a significant effect on the results and predictions. Neural

networks with simpler architecture led to poor performance, while relatively deeper NN architecture had good prediction accuracies for the plant parameters.

10.2 Limitations

It is important to note that, in this research all the data analysis, simulation and machine learning model training were done using a personal computer, which is not designed for time-consuming scientific simulations, machine learning model training and prediction. Having an access to a more advanced and high performance computer that are usually used for scientific research would significantly improve the simulation and training efficiency. This could also help assessing the impacts of different (e.g more complex) neural network architecture designs. Having an access to a better computer would also help try out different machine learning algorithms that require higher computational power such as gaussian process regression. It is highly recommended to use Keras and Tensorflow deep learning frameworks on a GPU as opposed to CPU (Allaire, 2018). However, in this study a computer with the required GPU was not available and the training of the ML model was conducted using CPU. Otherwise, this could have improved the training efficiency further.

Another important limitation of this study is unavailability of the ground truth data. This means that, the trained model could not be validated against the ground truth data due to unavailable ground truth data during the time this research was conducted.

It is also important to mention that, all the synthetic training data (training set) was produced based on the typical values the spruce trees show for the chose plant variables. However, the trained model was applied to the whole study area which also contains other tree species. This was done due to inaccessible GIS data (e.g. polygon) that accurately separates European spruce trees from other tree species in the forest.

10.3 Future perspectives

The model that was built with the help of the INFORM RTM and ANN can now be used to predict the plant traits C_{ab} , C_w , C_m and LAI_s in the hyperspectral remote sensing data PRISMA. These plant traits could be very beneficial to study forest disturbances especially for Spruce trees.

Potential follow up study could be to use this model in order to retrieve forest biochemical and biophysical maps from the hyperspectral data PRISMA for multiple dates. Then, scientific techniques like change detection could be applied to assess the change of these variables between multiple years. For this, many different time-series analysis techniques can be used to better understand the effect of the bark beetle and drought within the study area through time.

The follow-up time-series study based on biophysical and biochemical maps does not have to be limited to the PRISMA data. As all the scripts that were written to develop this model is publicly available on GitHub, few modification can be made to build a new model for any other remote sensing image bands.

It is also important to mention that this model could still be improved further. For example, using destructive and non-destructive in-situ data for RTM simulations could potentially lead to better results. In this study, these types of data were not available and the referenced literature was used to choose the range of values for RTM simulation. This data could also help assess the performance of the models (RTM and ANN) in more detail.

In this study, only ANN ML model was used. Other machine learning models could also be trained for predicting plant traits. ANN mainly focuses on predicting the plant variables as closely as possible. However, it would also be very useful to assess the uncertainty available within the predictions of the machine learning model. For example, the Gaussian process regression method would be used to retrieve plant traits and also produce data that shows the amount of uncertainties that is available within the predicted maps. However this method is computationally very expensive and may not efficiently work on large amount of training data.

10.3. Future perspectives

Another technique that could potentially be utilized is active learning (AL). AL techniques can be specifically useful for ML algorithms that demand significant amount of computational power to be trained on large amount of training data. AL can help choose the most important training samples that are available in the simulated training data (Verrelst et al., 2016). For example, AL methods can be combined with kernel based ML algorithms, such as Gaussian process regression.

11 Conclusion

In this research, radiative transfer models PROSPECT5, 4SAIL and FLIM were coupled to simulate forest canopy reflectance for the National Park Hunsruck-Hochwald. The simulated data was used to train ANN and the trained optimal and final ANN was used to predict forest biophysical and biochemical parameters on the PRISMA image of the study area.

Two ANN models were trained, one with a dimensionality reduction technique PCA and one without reducing the dimension of the hyperspectral data. In general, both of the trained NN models were found to perform well. However, the ANN that utilized all the available bands were found to be superior in terms of minimized cost function and prediction errors for each plant parameter. Specifically, prediction of the parameter LAI_s resulted in noticeably higher accuracy when DR technique was not applied.

The NN model that was trained with the 231 simulated PRISMA bands, was chosen as the optimal model and used for the final predictions. This model achieved MSE of 0.07670474 and MAE of 0.13 on the testing data set. The plant parameter C_{ab} was predicted with an R^2 of 0.89 and RMSE of 4.064 ($\frac{\mu g}{cm^2}$). C_w was predicted with an R^2 of 0.94 and RMSE of 0.003 ($\frac{g}{cm^2}$). R^2 and RMSE values for C_m were 0.95 and 0.002 ($\frac{g}{cm^2}$) respectively. Finally, R^2 and RMSE values for LAI_s were 0.95 and 0.44 ($\frac{m^2}{m^2}$), respectively.

Reliability of the model on the real data could be checked with the help of destructive and non-destructive in-situ data for specifically spruce trees. At the time of this study, there was no data that specifically shows spruce trees while masking the other species, the model was applied to the whole image that does not only contain spruce trees. These types of data would help assess how well the model is working on the real imagery that contain spectral information of spruce trees in the National Park Hunsruck-Hochwald.

References

- Abdelbaki, A., Schlerf, M., Retzlaff, R., Machwitz, M., Verrelst, J., and Udelhoven, T. Comparison of crop trait retrieval strategies using uav-based vnir hyperspectral imaging. *Remote Sensing*, 13(9):1748, 2021.
- Abdullah, H., Darvishzadeh, R., Skidmore, A. K., Groen, T. A., and Heurich, M. European spruce bark beetle (*ips typographus*, l.) green attack affects foliar reflectance and biochemical properties. *International journal of applied earth observation and geoinformation*, 64:199–209, 2018.
- Abdullah, H., Darvishzadeh, R., Skidmore, A. K., and Heurich, M. Sensitivity of landsat-8 oli and tirs data to foliar properties of early stage bark beetle (*ips typographus*, l.) infestation. *Remote sensing*, 11(4):398, 2019a.
- Abdullah, H., Skidmore, A. K., Darvishzadeh, R., and Heurich, M. Sentinel-2 accurately maps green-attack stage of european spruce bark beetle (*ips typographus*, l.) compared with landsat-8. *Remote sensing in ecology and conservation*, 5(1):87–106, 2019b.
- Ali, A. M., Skidmore, A. K., Darvishzadeh, R., van Duren, I., Holzwarth, S., and Mueller, J. Retrieval of forest leaf functional traits from hyspex imagery using radiative transfer models and continuous wavelet analysis. *ISPRS journal of photogrammetry and remote sensing*, 122:68–80, 2016.
- Ali, A. M., Darvishzadeh, R., Skidmore, A., Gara, T. W., and Heurich, M. Machine learning methods’ performance in radiative transfer model inversion to retrieve plant traits from sentinel-2 data of a mixed mountain forest. *International Journal of Digital Earth*, pages 1–15, 2020.
- Ali, A. M., Abdullah, H., Darvishzadeh, R., Skidmore, A. K., Heurich, M., Roeoesli, C., Paganini, M., Heiden, U., and Marshall, D. Canopy chlorophyll content retrieved from time series remote sensing data as a proxy for detecting bark beetle infestation. *Remote Sensing Applications: Society and Environment*, 22:100524, 2021.
- Allaire, J. *Deep Learning with R*. Simon and Schuster, 2018.
- Allaire, J. and Chollet, F. *keras: R Interface to 'Keras'*, 2021. URL <https://CRAN.R-project.org/package=keras>. R package version 2.4.0.
- Allaire, J. and Tang, Y. *tensorflow: R Interface to 'TensorFlow'*, 2021. URL <https://CRAN.R-project.org/package=tensorflow>. R package version 2.5.0.
- Atzberger, C. Development of an invertible forest reflectance model: The infor-model. In *A decade of trans-european remote sensing cooperation. Proceedings of the 20th EARSeL Symposium Dresden, Germany*, volume 14, pages 39–44, 2000.

References

- Bale, J. S., Masters, G. J., Hodkinson, I. D., Awmack, C., Bezemer, T. M., Brown, V. K., Butterfield, J., Buse, A., Coulson, J. C., Farrar, J., et al. Herbivory in global climate change research: direct effects of rising temperature on insect herbivores. *Global change biology*, 8(1):1–16, 2002.
- Bentz, B. J. and Jönsson, A. M. Modeling bark beetle responses to climate change. In *Bark beetles*, pages 533–553. Elsevier, 2015.
- Berger, K., Atzberger, C., Danner, M., D’Urso, G., Mauser, W., Vuolo, F., and Hank, T. Evaluation of the prosail model capabilities for future hyperspectral model environments: A review study. *Remote Sensing*, 10(1):85, 2018.
- Berger, K., Verrelst, J., Féret, J.-B., Hank, T., Wocher, M., Mauser, W., and Camps-Valls, G. Retrieval of aboveground crop nitrogen content with a hybrid machine learning method. *International Journal of Applied Earth Observation and Geoinformation*, 92:102174, 2020.
- Berger, K., Rivera Caicedo, J. P., Martino, L., Wocher, M., Hank, T., and Verrelst, J. A survey of active learning for quantifying vegetation traits from terrestrial earth observation data. *Remote Sensing*, 13(2):287, 2021.
- Boser, B. E., Guyon, I. M., and Vapnik, V. N. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.
- Breiman, L. Random forests. *Machine learning*, 45(1):5–32, 2001.
- Bro, R. and Smilde, A. K. Principal component analysis. *Analytical methods*, 6(9): 2812–2831, 2014.
- Buddenbaum, H., Dotzler, S., Hill, J., Consortium, E., et al. Nationalpark hunsrück-hochwald, 2014-05-05-an enmap preparatory flight campaign. 2015.
- Caicedo, J. P. R., Verrelst, J., Muñoz-Marí, J., Moreno, J., and Camps-Valls, G. Toward a semiautomatic machine learning retrieval of biophysical parameters. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(4):1249–1259, 2014.
- Camps-Valls, G., Verrelst, J., Munoz-Mari, J., Laparra, V., Mateo-Jiménez, F., and Gómez-Dans, J. A survey on gaussian processes for earth-observation data analysis: A comprehensive investigation. *IEEE Geoscience and Remote Sensing Magazine*, 4(2): 58–78, 2016.
- Camps-Valls, G., Sejdinovic, D., Runge, J., and Reichstein, M. A perspective on gaussian processes for earth observation. *National Science Review*, 6(4):616–618, 2019.
- Candela, L., Formaro, R., Guarini, R., Loizzo, R., Longo, F., and Varacalli, G. The prisma mission. In *2016 IEEE international geoscience and remote sensing symposium (IGARSS)*, pages 253–256. IEEE, 2016.
- César de Sá, N., Baratchi, M., Hauser, L. T., and van Bodegom, P. Exploring the impact of noise on hybrid inversion of prosail rtm on sentinel-2 data. *Remote Sensing*, 13(4): 648, 2021.

- Combal, B., Baret, F., Weiss, M., Trubuil, A., Mace, D., Pragnere, A., Myneni, R., Knyazikhin, Y., and Wang, L. Retrieval of canopy biophysical variables from bidirectional reflectance: Using prior information to solve the ill-posed inverse problem. *Remote sensing of environment*, 84(1):1–15, 2003.
- Corporation, M. and Weston, S. *doParallel: Foreach Parallel Adaptor for the 'parallel' Package*, 2020. URL <https://CRAN.R-project.org/package=doParallel>. R package version 1.0.16.
- Danner, M., Berger, K., Wocher, M., Mauser, W., and Hank, T. Efficient rtm-based training of machine learning regression algorithms to quantify biophysical & biochemical traits of agricultural crops. *ISPRS Journal of Photogrammetry and Remote Sensing*, 173:278–296, 2021.
- Darvishzadeh, R., Skidmore, A., Atzberger, C., and van Wieren, S. Estimation of vegetation lai from hyperspectral reflectance data: Effects of soil type and plant architecture. *International journal of applied Earth observation and geoinformation*, 10(3):358–373, 2008.
- Darvishzadeh, R., Skidmore, A., Abdullah, H., Cherenet, E., Ali, A., Wang, T., Nieuwenhuis, W., Heurich, M., Vrieling, A., O'Connor, B., et al. Mapping leaf chlorophyll content from sentinel-2 and rapideye data in spruce stands using the invertible forest reflectance model. *International Journal of Applied Earth Observation and Geoinformation*, 79:58–70, 2019a.
- Darvishzadeh, R., Wang, T., Skidmore, A., Vrieling, A., O'Connor, B., Gara, T. W., Ens, B. J., and Paganini, M. Analysis of sentinel-2 and rapideye for retrieval of leaf area index in a saltmarsh using a radiative transfer model. *Remote sensing*, 11(6):671, 2019b.
- De Grave, C., Verrelst, J., Morcillo-Pallarés, P., Pipia, L., Rivera-Caicedo, J. P., Amin, E., Belda, S., and Moreno, J. Quantifying vegetation biophysical variables from the sentinel-3/flex tandem mission: Evaluation of the synergy of olci and floris data sources. *Remote Sensing of Environment*, 251:112101, 2020.
- Drucker, H., Burges, C. J., Kaufman, L., Smola, A., Vapnik, V., et al. Support vector regression machines. *Advances in neural information processing systems*, 9:155–161, 1997.
- Esquivel-Muelbert, A., Baker, T. R., Dexter, K. G., Lewis, S. L., Brien, R. J., Feldpausch, T. R., Lloyd, J., Monteagudo-Mendoza, A., Arroyo, L., Álvarez-Dávila, E., et al. Compositional response of amazon forests to climate change. *Global Change Biology*, 25(1):39–56, 2019.
- Feret, J.-B., François, C., Asner, G. P., Gitelson, A. A., Martin, R. E., Bidel, L. P., Ustin, S. L., Le Maire, G., and Jacquemoud, S. Prospect-4 and 5: Advances in the leaf optical properties model separating photosynthetic pigments. *Remote sensing of environment*, 112(6):3030–3043, 2008.
- Fernandez-Carrillo, A., Patočka, Z., Dobrovolný, L., Franco-Nieto, A., and Revilla-Romero, B. Monitoring bark beetle forest damage in central europe. a remote sensing approach validated with field data. *Remote Sensing*, 12(21):3634, 2020.

References

- Fernández-Guisuraga, J. M., Verrelst, J., Calvo, L., and Suárez-Seoane, S. Hybrid inversion of radiative transfer models based on high spatial resolution satellite reflectance data improves fractional vegetation cover retrieval in heterogeneous ecological systems after fire. *Remote Sensing of Environment*, 255:112304, 2021.
- Fischer-Stabel, P. and Mattern, M. Scientific research documentation-an information server for the national park hunsrück-hochwald. 2016.
- Ghasemi, F., Mehridehnavi, A., Perez-Garrido, A., and Perez-Sanchez, H. Neural network and deep-learning algorithms used in qsar studies: merits and drawbacks. *Drug Discov. Today*, 23(10):1784–1790, 2018.
- Giardino, C., Bresciani, M., Braga, F., Fabbretto, A., Ghirardi, N., Pepe, M., Gianinetto, M., Colombo, R., Cogliati, S., Ghebrehiwot, S., et al. First evaluation of prisma level 1 data for water applications. *Sensors*, 20(16):4553, 2020.
- Gómez-Chova, L., Muñoz-Marí, J., Laparra, V., Malo-López, J., and Camps-Valls, G. A review of kernel methods in remote sensing data analysis. *Optical Remote Sensing*, pages 171–206, 2011.
- Goodfellow, I., Bengio, Y., and Courville, A. *Deep learning*. MIT press, 2016.
- Grammatikopoulou, I. and Vačkářová, D. The value of forest ecosystem services: A meta-analysis at the european scale and application to national ecosystem accounting. *Ecosystem Services*, 48:101262, 2021.
- Han, Z., Zhu, X., Fang, X., Wang, Z., Wang, L., Zhao, G.-X., and Jiang, Y. Hyperspectral estimation of apple tree canopy lai based on svm and rf regression. *Guang pu xue yu Guang pu fen xi= Guang pu*, 36(3):800–805, 2016.
- He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.
- Hinton, G. Neural networks for machine learning coursera video lecturesgeoffrey hinton. 2012.
- Hughes, G. On the mean accuracy of statistical pattern recognizers. *IEEE transactions on information theory*, 14(1):55–63, 1968.
- Immitzer, M. and Atzberger, C. Early detection of bark beetle infestation in norway spruce (picea abies, l.) using worldview-2 data. *Photogramm. Fernerkund. Geoinf*, 2014 (351367):73, 2014.
- Ingram, J. C., Dawson, T. P., and Whittaker, R. J. Mapping tropical forest structure in southeastern madagascar using remote sensing and artificial neural networks. *Remote Sensing of Environment*, 94(4):491–507, 2005.
- Jacquemoud, S. and Baret, F. Prospect: A model of leaf optical properties spectra. *Remote sensing of environment*, 34(2):75–91, 1990.
- Jacquemoud, S. and Ustin, S. *Leaf optical properties*. Cambridge University Press, 2019.

- Jay, S., Maupas, F., Bendoula, R., and Gorretta, N. Retrieving lai, chlorophyll and nitrogen contents in sugar beet crops from multi-angular optical remote sensing: Comparison of vegetation indices and prosail inversion for field phenotyping. *Field Crops Research*, 210:33–46, 2017.
- Jensen, J., Qiu, F., and Ji, M. Predictive modelling of coniferous forest age using statistical and artificial neural network approaches applied to remote sensor data. *International Journal of Remote Sensing*, 20(14):2805–2822, 1999.
- Kalacska, M., Lalonde, M., and Moore, T. Estimation of foliar chlorophyll and nitrogen content in an ombrotrophic bog from hyperspectral data: Scaling from leaf to image. *Remote Sensing of Environment*, 169:270–279, 2015.
- Karimi, Y., Prasher, S., Madani, A., Kim, S., et al. Application of support vector machine technology for the estimation of crop biophysical parameters using aerial hyperspectral observations. *Canadian Biosystems Engineering*, 50(7):13–20, 2008.
- Kautz, M., Meddens, A. J., Hall, R. J., and Arneeth, A. Biotic disturbances in northern hemisphere forests—a synthesis of recent data, uncertainties and implications for forest monitoring and modelling. *Global Ecology and Biogeography*, 26(5):533–552, 2017.
- Ke, L., ZHOU, Q.-b., WU, W.-b., Tian, X., and TANG, H.-j. Estimating the crop leaf area index using hyperspectral remote sensing. *Journal of integrative agriculture*, 15(2):475–491, 2016.
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Kuhn, M. and Wickham, H. *Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles.*, 2020. URL <https://www.tidymodels.org>.
- Kuhn, M. and Wickham, H. *recipes: Preprocessing Tools to Create Design Matrices*, 2021. URL <https://CRAN.R-project.org/package=recipes>. R package version 0.1.16.
- Laurent, V. C., Verhoef, W., Clevers, J. G., and Schaepman, M. E. Inversion of a coupled canopy–atmosphere model using multi-angular top-of-atmosphere radiance data: A forest case study. *Remote Sensing of Environment*, 115(10):2603–2612, 2011.
- Lausch, A., Heurich, M., Gordalla, D., Dobner, H.-J., Gwilym-Margianto, S., and Salbach, C. Forecasting potential bark beetle outbreaks based on spruce forest vitality using hyperspectral remote-sensing techniques at different scales. *Forest Ecology and Management*, 308:76–89, 2013.
- Lechner, A. M., Foody, G. M., and Boyd, D. S. Applications in remote sensing to forest ecology and management. *One Earth*, 2(5):405–412, 2020.
- Lee, J. A. and Verleysen, M. *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007.
- Lehnert, L. W., Meyer, H., Obermeier, W. A., Silva, B., Regeling, B., Thies, B., and Bendix, J. Hyperspectral data analysis in R: The hsdar package. *Journal of Statistical Software*, 89(12):1–23, 2019. doi: 10.18637/jss.v089.i12.

References

- Lindner, M., Maroschek, M., Netherer, S., Kremer, A., Barbati, A., Garcia-Gonzalo, J., Seidl, R., Delzon, S., Corona, P., Kolström, M., et al. Climate change impacts, adaptive capacity, and vulnerability of european forest ecosystems. *Forest ecology and management*, 259(4):698–709, 2010.
- Liu, W. Y. and Pan, J. A hyperspectral assessment model for leaf chlorophyll content of pinus massoniana based on neural network. *Ying yong sheng tai xue bao= The journal of applied ecology*, 28(4):1128–1136, 2017.
- Malenovsky, Z., Homolová, L., Zurita-Milla, R., Lukeš, P., Kaplan, V., Hanuš, J., Gastellu-Etchegorry, J.-P., and Schaepman, M. E. Retrieval of spruce leaf chlorophyll content from airborne image data using continuum removal and radiative transfer. *Remote Sensing of Environment*, 131:85–102, 2013.
- Marini, L., Økland, B., Jönsson, A. M., Bentz, B., Carroll, A., Forster, B., Grégoire, J.-C., Hurling, R., Nageleisen, L. M., Netherer, S., et al. Climate drivers of bark beetle outbreak dynamics in norway spruce forests. *Ecography*, 40(12):1426–1435, 2017.
- Microsoft and Weston, S. *foreach: Provides Foreach Looping Construct*, 2020. URL <https://CRAN.R-project.org/package=foreach>. R package version 1.5.1.
- Millar, C. I. and Stephenson, N. L. Temperate forest health in an era of emerging megadisturbance. *Science*, 349(6250):823–826, 2015.
- Morcillo-Pallarés, P., Rivera-Caicedo, J. P., Belda, S., De Grave, C., Burriel, H., Moreno, J., and Verrelst, J. Quantifying the robustness of vegetation indices through global sensitivity analysis of homogeneous and forest leaf-canopy radiative transfer models. *Remote Sensing*, 11(20):2418, 2019.
- Neinavaz, E., Skidmore, A. K., Darvishzadeh, R., and Groen, T. A. Retrieval of leaf area index in different plant species using thermal hyperspectral data. *ISPRS journal of photogrammetry and remote sensing*, 119:390–401, 2016.
- Niemann, K. O., Visintini, F., et al. Assessment of potential for remote sensing detection of bark beetle-infested areas during green attack: a literature review. 2005.
- Öhrn, P., Långström, B., Lindelöw, Å., and Björklund, N. Seasonal flight patterns of ips typographus in southern sweden and thermal sums required for emergence. *Agricultural and Forest Entomology*, 16(2):147–157, 2014.
- Ortiz, S. M., Breidenbach, J., and Kändler, G. Early detection of bark beetle green attack using terrasars-x and rapideye data. *Remote Sensing*, 5(4):1912–1931, 2013.
- Paine, T., Raffa, K., and Harrington, T. Interactions among scolytid bark beetles, their associated fungi, and live host conifers. *Annual review of entomology*, 42(1):179–206, 1997.
- Polyak, B. T. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.

- Powell, S. L., Cohen, W. B., Healey, S. P., Kennedy, R. E., Moisen, G. G., Pierce, K. B., and Ohmann, J. L. Quantification of live aboveground forest biomass dynamics with landsat time-series and field inventory data: A comparison of empirical modeling approaches. *Remote Sensing of Environment*, 114(5):1053–1068, 2010.
- Pullanagari, R., Kereszturi, G., and Yule, I. Mapping of macro and micro nutrients of mixed pastures using airborne aisafenix hyperspectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117:1–10, 2016.
- Quan, X., He, B., Yebra, M., Yin, C., Liao, Z., Zhang, X., and Li, X. A radiative transfer model-based method for the estimation of grassland aboveground biomass. *International Journal of Applied Earth Observation and Geoinformation*, 54:159–168, 2017.
- R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021. URL <https://www.R-project.org/>.
- Raffa, K. F., Aukema, B., Bentz, B. J., Carroll, A., Erbilgin, N., Herms, D. A., Hicke, J. A., Hofstetter, R. W., Katovich, S., Lindgren, B. S., et al. A literal use of “forest health” safeguards against misuse and misapplication. *Journal of Forestry*, 107(5): 276–277, 2009.
- Rivera, J. P., Verrelst, J., Gómez-Dans, J., Muñoz-Marí, J., Moreno, J., and Camps-Valls, G. An emulator toolbox to approximate radiative transfer models with statistical learning. *Remote Sensing*, 7(7):9347–9370, 2015.
- Rivera-Caicedo, J. P., Verrelst, J., Muñoz-Marí, J., Camps-Valls, G., and Moreno, J. Hyperspectral dimensionality reduction for biophysical variable statistical retrieval. *ISPRS journal of photogrammetry and remote sensing*, 132:88–101, 2017.
- Rosema, A., Verhoef, W., Noorbergen, H., and Borgesius, J. A new forest light interaction model in support of forest monitoring. *Remote Sensing of Environment*, 42(1):23–41, 1992.
- Ruder, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- Schlerf, M. and Atzberger, C. Inversion of a forest reflectance model to estimate structural canopy variables from hyperspectral remote sensing data. *Remote sensing of environment*, 100(3):281–294, 2006.
- Schlerf, M. and Atzberger, C. Vegetation structure retrieval in beech and spruce forests using spectrodirectional satellite data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(1):8–17, 2012.
- Senf, C., Seidl, R., and Hostert, P. Remote sensing of forest insect disturbances: Current state and future directions. *International Journal of Applied Earth Observation and Geoinformation*, 60:49–60, 2017.
- Simon, H. *Neural networks: a comprehensive foundation*. Prentice hall, 1999.

References

- Sinha, S. K., Padalia, H., Dasgupta, A., Verrelst, J., and Rivera, J. P. Estimation of leaf area index using prosail based lut inversion, mlra-gpr and empirical models: Case study of tropical deciduous forest plantation, north india. *International Journal of Applied Earth Observation and Geoinformation*, 86:102027, 2020.
- Six, D. L., Harrington, T. C., Steimel, J., McNew, D., and Paine, T. Genetic relationships among leptographium terebrantis and the mycangial fungi of three western dendroctonus bark beetles. *Mycologia*, 95(5):781–792, 2003.
- Sprintsin, M., Chen, J. M., and Czurylowicz, P. Combining land surface temperature and shortwave infrared reflectance for early detection of mountain pine beetle infestations in western canada. *Journal of Applied Remote Sensing*, 5(1):053566, 2011.
- Suits, G. H. The calculation of the directional reflectance of a vegetative canopy. *Remote Sensing of Environment*, 2:117–125, 1971.
- Tuia, D., Verrelst, J., Alonso, L., Pérez-Cruz, F., and Camps-Valls, G. Multioutput support vector regression for remote sensing biophysical parameter estimation. *IEEE Geoscience and Remote Sensing Letters*, 8(4):804–808, 2011.
- Upreti, D., Huang, W., Kong, W., Pascucci, S., Pignatti, S., Zhou, X., Ye, H., and Casa, R. A comparison of hybrid machine learning algorithms for the retrieval of wheat biophysical variables from sentinel-2. *Remote Sensing*, 11(5):481, 2019.
- Verhoef, W., Jia, L., Xiao, Q., and Su, Z. Unified optical-thermal four-stream radiative transfer theory for homogeneous vegetation canopies. *IEEE Transactions on geoscience and remote sensing*, 45(6):1808–1822, 2007.
- Verhoef, W. Light scattering by leaf layers with application to canopy reflectance modeling: The sail model. *Remote sensing of environment*, 16(2):125–141, 1984.
- Verrelst, J., Dethier, S., Rivera, J. P., Munoz-Mari, J., Camps-Valls, G., and Moreno, J. Active learning methods for efficient hybrid biophysical variable retrieval. *IEEE Geoscience and Remote Sensing Letters*, 13(7):1012–1016, 2016.
- Verrelst, J., Malenovsky, Z., Van der Tol, C., Camps-Valls, G., Gastellu-Etchegorry, J.-P., Lewis, P., North, P., and Moreno, J. Quantifying vegetation biophysical variables from imaging spectroscopy data: a review on retrieval methods. *Surveys in Geophysics*, 40(3):589–629, 2019.
- Verrelst, J., Rivera-Caicedo, J. P., Reyes-Muñoz, P., Morata, M., Amin, E., Tagliabue, G., Panigada, C., Hank, T., and Berger, K. Mapping landscape canopy nitrogen content from space using prisma data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 178:382–395, 2021.
- Visser, M. D. *ccrtm: Coupled Chain Radiative Transfer Models*, 2021. URL <https://CRAN.R-project.org/package=ccrtm>. R package version 0.2.
- Walczak, S. Artificial neural networks. In *Advanced Methodologies and Technologies in Artificial Intelligence, Computer Simulation, and Human-Computer Interaction*, pages 40–53. IGI Global, 2019.

- Wang, L., Chang, Q., Yang, J., Zhang, X., and Li, F. Estimation of paddy rice leaf area index using machine learning methods based on hyperspectral data from multi-year experiments. *PloS One*, 13(12):e0207624, 2018a.
- Wang, Z., Skidmore, A. K., Wang, T., Darvishzadeh, R., Heiden, U., Heurich, M., Latifi, H., and Hearne, J. Canopy foliar nitrogen retrieved from airborne hyperspectral imagery by correcting for canopy structure effects. *International Journal of Applied Earth Observation and Geoinformation*, 54:84–94, 2017.
- Wang, Z., Skidmore, A. K., Darvishzadeh, R., and Wang, T. Mapping forest canopy nitrogen content by inversion of coupled leaf-canopy radiative transfer models from airborne hyperspectral imagery. *Agricultural and forest meteorology*, 253:247–260, 2018b.
- Wei, C., Huang, J., Mansaray, L. R., Li, Z., Liu, W., and Han, J. Estimation and mapping of winter oilseed rape lai from high spatial resolution satellite data based on a hybrid method. *Remote Sensing*, 9(5):488, 2017.
- Wermelinger, B. Ecology and management of the spruce bark beetle *ips typographus*—a review of recent research. *Forest ecology and management*, 202(1-3):67–82, 2004.
- Wickham, H. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4. URL <https://ggplot2.tidyverse.org>.
- Wickham, H. and Grolemund, G. *R for data science: import, tidy, transform, visualize, and model data*. " O'Reilly Media, Inc.", 2016.
- Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, 2019. doi: 10.21105/joss.01686.
- Wulder, M. A., White, J., Bentz, B., Alvarez, M., and Coops, N. Estimating the probability of mountain pine beetle red-attack damage. *Remote Sensing of Environment*, 101(2):150–166, 2006.
- Wulder, M. A., White, J. C., Carroll, A. L., and Coops, N. C. Challenges for the operational detection of mountain pine beetle green attack with remote sensing. *The Forestry Chronicle*, 85(1):32–38, 2009.
- Yang, J., Zhang, Y., Du, L., Liu, X., Shi, S., and Chen, B. Improving the selection of vegetation index characteristic wavelengths by using the prospect model for leaf water content estimation. *Remote Sensing*, 13(4):821, 2021.
- Yang, X., Huang, J., Wu, Y., Wang, J., Wang, P., Wang, X., and Huete, A. R. Estimating biophysical parameters of rice with remote sensing data using support vector machines. *Science China Life Sciences*, 54(3):272–281, 2011.
- Zhang, Y., Yang, J., and Du, L. Analyzing the effects of hyperspectral zhuhai-1 band combinations on lai estimation based on the prosail model. *Sensors*, 21(5):1869, 2021.

References

- Zhu, X., Skidmore, A. K., Darvishzadeh, R., and Wang, T. Estimation of forest leaf water content through inversion of a radiative transfer model from lidar and hyperspectral data. *International Journal of Applied Earth Observation and Geoinformation*, 74:120–129, 2019.