



Universiteit
Leiden
The Netherlands

Opleiding Informatica

Using a data-driven approach to estimate the vegetation height
using remote sensing data: a case study in the Oostvaardersplassen

Vincent Buitenhuis

Supervisors:

Nuno César de Sá (CML)

Dr. Mitra Baratchi (LIACS)

Prof. dr. ir. Peter van Bodegom (CML)

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)

www.liacs.leidenuniv.nl

4/9/2020

Abstract

The Oostvaardersplassen is a nature reserve in the Netherlands known as an example of rewilding conservation. The park hosts large herbivore populations which, due to the lack of predators, have grown to unsustainable numbers. Despite reactive culling actions, the populations have experienced high rates of death by starvation. An estimation of food availability as biomass could be used by the park management to reduce unnecessary suffering. In order to assess the amount of biomass in a certain area, we can use remote sensing techniques that are to be effective in providing accurate estimates of biomass in other grasslands. The main advantage of using this method to make estimations is that large amounts of data from every place on earth can be collected and used to analyse at unprecedented spatial and temporal scales. Previous studies have often used remote sensing data in combination with machine learning techniques, while conducting the whole process manually.

This study aims to assess which machine learning methods can be used to provide biomass estimates from remote sensing data. We have compared previous examples with more data-driven approaches to make estimations of vegetation height. In addition, we aim to capture the relationships of remote sensing features within the data. We have compared three different scenarios, using reference field data in the form of grass height from the Oostvaardersplassen, and remote sensing data from Landsat satellites. Firstly, we tested the standard approaches used in remote sensing studies, to estimate vegetation height. Secondly, we tested the usage of algorithms which automate the entire machine learning procedure. Thirdly, we have tested using raw remote sensing data, instead of derived features as done in the previous scenarios, in a neural network to make estimations. During the third scenario, we have also examined the properties of the neural network to discover the relations between the data. We discovered that the automated machine learning algorithms performed better than most conventional methods. The H2O AutoML and Auto-sklearn algorithms made estimations with a mean absolute error of 2.04 cm and 2.12 cm respectively. However, the SVM made estimations with a 2.08 cm mean absolute error. Moreover, we found no new relations other than what was already known in a remote sensing context.

Acknowledgements

I would first like to thank my supervisors, Nuno César de Sá, from the Institute of Environmental Sciences (CML), and Mitra Baratchi from Leiden Institute of Advanced Computer Science (LIACS) for the opportunity for me to work on this project, as well as their support and guidance on this thesis. This project has been a very interesting experience educationally and personally. I would also like to thank my family, friends and girlfriend for their support and for providing inspiration to work as hard as I can on this thesis.

Contents

1	Introduction	1
1.1	The situation	1
1.2	Related work	2
1.3	Research questions	3
1.4	Research Design	4
2	Methods	6
2.1	Study Area	6
2.2	Data	6
2.2.1	Field Data	6
2.2.2	Remote Sensing Data	7
2.2.3	Landsat mission	7
2.2.4	Google Earth Engine	8
2.2.5	Vegetation indices	8
2.2.6	Preprocessing	9
2.3	Machine Learning	10
2.3.1	Algorithms	10
2.3.2	Automated Machine Learning	12
3	Experiments	14
3.1	Experiment protocol	14
3.1.1	Data	14
3.1.2	Metrics	14
3.1.3	Cross Validation	15
3.1.4	Hyper-parameter tuning	15
3.2	Experimental setup scenario I	15
3.3	Experimental setup scenario II	17
3.4	Experimental setup scenario III	18
4	Results	19
4.1	Results scenario I	19
4.2	Results scenario II	21
4.3	Results scenario III	22
4.4	Comparison	23
4.5	Limitations	24
5	Conclusions	25
5.1	Discussion	25
5.2	Conclusion	26
5.3	Further research	27
	References	28

Appendices	34
A Field data locations	34
B Spectral bands	37
C Literature examination	38
D Hyper-parameter search spaces	39
E Scenario I results	40
E.1 MAE scores	40
E.2 MAPE scores	43
E.3 RMSE scores	46
E.4 All performance score values	49
F Scenario II results	50
G Scenario III results	54

1 Introduction

In this section we will describe the current situation at the focus of our research. We will also include a description of related work. This section also details the research goals and questions, as well as a description of the research design.

1.1 The situation

Grassland ecosystems can be classified as one of the most important types of ecosystems, as they provide functions such as soil and water conservation, and air purification (Zhao et al, 2014) [1]. Grassland biomass is an indicator of grassland productivity, which provides important information for animal husbandry [2].

There are various methods which can be used to make estimations of above ground biomass. The three main categories are field-measurement based, remote sensing based, and GIS based methods [3]. Remote sensing based methods can be described as acquiring data using an instrument that is not in direct contact with the object being analyzed [4]. In the case of estimating above ground biomass it is useful to monitor large areas, without the need to use methods that can be time consuming or harmful to nature. This method has been used numerous times to make biomass estimations, since it has a great advantage because of its global coverage and cost-effectiveness [5].

The Oostvaardersplassen (OVP) is a nature reserve located in the Dutch province Flevoland. The reserve has been created in 1968 after the area has been poldered, and consists of a wide variety of land types such as marshlands, wet and dry grasslands, and forest areas [6]. The vision of the park management is to rewind the nature nature reserve, meaning a number of animals will be introduced to the area, while the human intervention will be as little as possible [7]. The management of the park will downplay direct intervention in the natural processes so nature can develop on its own.

The OVP has been a frequent subject to controversy. Due to minimal human intervention, herbivore population has grown. While reactive culling actions have been implemented to control their numbers, it has not been able to curb the high rates of death by starvation [8]. As this has happened multiple times, it has been a long-running subject of public debate in the Netherlands, even involving politics on a national level [7, 9]. A prediction of the food availability as biomass in the Oostvaardersplassen could be used to reduce unnecessary suffering. The park management is able to anticipate and make decisions much sooner with such information.

1.2 Related work

Using remote sensing techniques to estimate biomass is nothing new, as explained in the previous section. It is regarded as the most effective way to make predictions on aboveground grassland biomass (AGB) [10]. The form of remote sensing that is interesting to our research, is the use of passive sensors on satellites. These sensors measure the spectral reflectance from the surface of the earth, and can be derived to features which have been successful in the assessment of AGB [11].

Machine learning techniques are interesting to our research as well, as they have been used often and successfully to make estimations from remote sensing data. Wolanin et al. (2019) [12] have used neural networks and random forests on data from the Sentinel-2 and Landat 8 satellite missions, to estimate the primary productivity of crops. Wu et al. (2016) [13] conducted a study which compared multiple machine learning methods from Landsat 8 data. The predictions were made on AGB values from a forest area in the Zhejiang province in China.

Zhang et al. (2015) [14] predicted the AGB in the Xilinhot region in Inner Mongolia, China. They have used an support vector machine regression model to predict data from Landsat 5 and MODIS. Xie et al. (2009) [15] also made predictions for AGB values in the Xilinhot region. The machine learning technique they used to accomplish this is a neural network, using input from an unspecified Landsat mission. Zeng et al. (2019) [16] estimated the AGB of the Tibetan Plateau, using data from the MODIS VI mission and meteorological data such as temperature and precipitation. Random forest has been the algorithm of choice for this study. López-Serrano et al. (2016) [18] compared several machine learning techniques for biomass estimation. The study has used data from Landsat 5 to test a support vector machine, neural network, and k-nearest neighbour algorithm.

Yang et al. (2017) [17] estimated the the AGB of the Three-River Headwaters Region of the Qinghai province in China. They have successfully applied a neural network to MODIS data. Wang et al. (2019) [19] estimated the Leaf Area Index of AGB using data from Sentinel-1, Sentinel-2, and Landsat 8. This data was used by a support vector machine and random forest algorithm to create predictive models. Gleason et al. (2012) [20] used LiDAR data to acquire biomass reference values of a forest in New York, USA. This study tested multiple algorithms, including random forest and support vector machine. Wu et al. (2019) [21] predicted the AGB of a forest spanning across multiple nature reserves in China. They have also tested multiple machine learning algorithms to make predictive models, including the artificial neural network and support vector machine.

These studies have made use of the conventional way to utilise machine learning algorithms, where all steps - feature selection, feature preprocessing, model selection, and hyper-parameter optimization - are executed manually. While the use of conventional methods has been effective, little to no research has been done to incorporate different approaches to use machine learning in the process of estimating AGB. These different approaches are the use of automated machine learning methods and using a data driven approach in which non-derived features can be used in a predictive model. This research has explored these different approaches.

1.3 Research questions

For this study, we have substituted AGB with vegetation height, due to the unavailability of biomass data for the OVP. So the main goals of this research were to assess whether data-driven approaches can deliver accurate estimations of vegetation height, and to find relations within this data other than what is previously known. This can be achieved by comparing these approaches to those that are commonly used in remote sensing research. Having this in mind, we have simultaneously focused on the OVP, being our main study site. Once we assessed the best performing method to predict vegetation height values, the resulting model would be the most suitable to assist in predicting the food availability in the OVP.

Having the main goals of this research in mind, we can define the research question as follows:

- RQ1: Are automated machine learning approaches more effective than standard methods in predicting the vegetation height of the Oostvaardersplassen?
- RQ2: Can a data-driven approach be used to capture other relationships in remote sensing data other than previously known in a remote-sensing perspective?

1.4 Research Design

To evaluate the effectiveness of data-driven approaches we have defined a number of scenarios that each represent a different approach of vegetation height estimation using machine learning algorithms. In this section we will explain the different scenarios, and the steps which will take place to test them. Figure 1.1 visualizes the steps for each of these scenarios, and the steps required for preparation. Diamonds represent an action, and rectangles represent input and output. For each scenario, we have documented the results in terms of accuracy. These were measured using Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE).

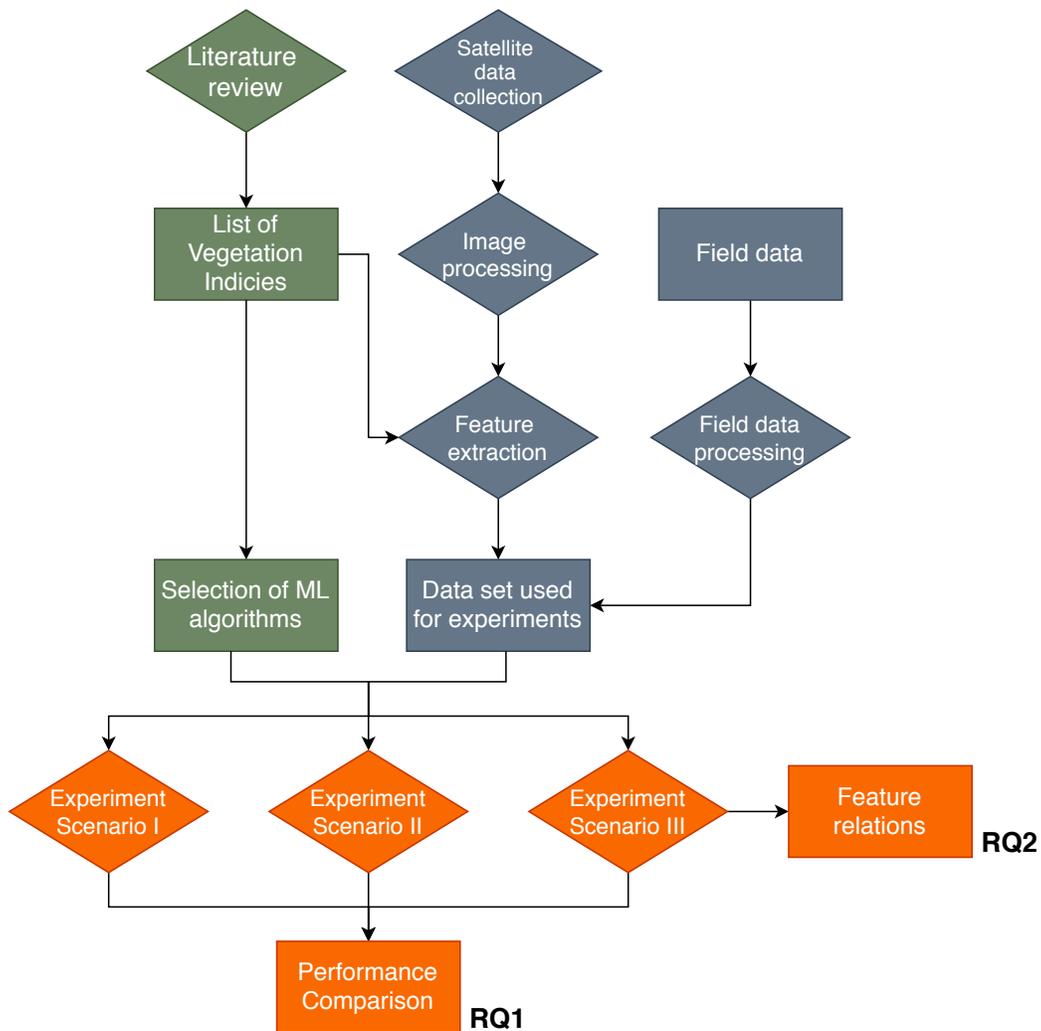


Figure 1.1: Flowchart of the research design

Scenario I is an emulation of standard approaches used in similar remote sensing studies. We used frequently used methods and frequently used features. To determine what these methods and features were, we compiled a number of remote sensing studies on similar topics. After this evaluation, we had a selection of the features, machine learning algorithms and preprocessing methods. As part of this scenario, we compared the results of each algorithm using the features in the selection.

Scenario II focused on an automated method which can be described as an input-output approach to make estimations on the vegetation height. All of the optimization - model, feature and hyper-parameter selection - will be done automatically. Automated machine learning is a fairly new method, and has not been used much in research, although there have been recent studies which used automated machine learning algorithms to make predictions successfully. Tanaka et. al (2019) [22] used the auto-sklearn Python library to predict software defects. In remote sensing, the usage of such algorithms is little to none.

In scenario III, we tested an automated machine learning algorithm for a neural network to predict vegetation height using only non-derived features, which are spectral band values in this case. The purpose of this scenario is for a neural network to discover relations on raw remote sensing data, so there would be no need to extract derived features from this data.

Before any of these scenarios are effectively able to be examined, we first obtained the data which was used for this study, and preprocessed it accordingly. After all results have been formed, we compared the results of each scenario with each other.

2 Methods

2.1 Study Area

As mentioned first in Section 1, the main study site for this research is the Oostvaardersplassen (OVP) nature reserve. In particular, we looked at the grasslands area. This area lies on the out brim of the OVP, as indicated with the teal-colored outline in Figure 2.1.

This grassland contains both dry and wet grassland areas, and are being grazed by a population of large herbivores introduced by humans in support of the rewilding vision of the park [23]. This population consists of three species: heck cattle, red deer and konik horses. Besides these species, the OVP also knows a large population of geese [24]. All reference data used in this research has been collected from the OVP.



Figure 2.1: Image of the Oostvaardersplassen. Credits to Esri Nederland (2019) for the air photo

2.2 Data

This section will explain how the data used for this study has been collected, cleaned, and processed. This data can be split into two categories. First we have field data which functions as target variable to train and validate the model. Second, we will describe the data that has been collected through remote sensing techniques. This data is used to train the model and estimate the value of the target variable.

2.2.1 Field Data

For this research Dr. Perry Cornelissen from Staatsbosbeheer - the Dutch national office for forest management - has provided us with a database containing the grass height and cover percentage for grass and a number of additional plants, from a number of measurements. These measurements were conducted over a 5-year period on the Oostvaardersplassen grasslands from 2013 to 2017. The grasslands are divided in a set of plots, of which 10 are included in this database. For each plot, the measurements have been carried out in a straight line from one edge of the plot to the other. In appendix A we can see this visualised in images obtained from Staatsbosbeheer. Depending on the plot, the measurements were taken in a 30 or 50 meter interval.

For this study, we only extracted the grass height data from this data set. We have not been provided with exact biomass data for these locations, or models to calculate the biomass from the grass height. However, multiple studies [25, 26, 27] show significant correlations between AGB and grass height. For this research, instead of assuming a theoretical or empirical relationship between grass height and AGB we directly aimed to predict vegetation height.

2.2.2 Remote Sensing Data

For this study we have used remotely sensed data retrieved from satellite sensors. The satellite sensors measure the values of specific regions in the wavelength spectrum that have been reflected from the earth's surface, after this part of the surface has been hit by light [34]. In Figure 2.2 we can see a visualisation of this spectrum. The numbers in the rectangles represent the band number of the Landsat sensor. The location of each rectangle on the graph represents the wavelength region for that band. The upper row of bands are included in Landsat 8, and the lower row in Landsat 7.

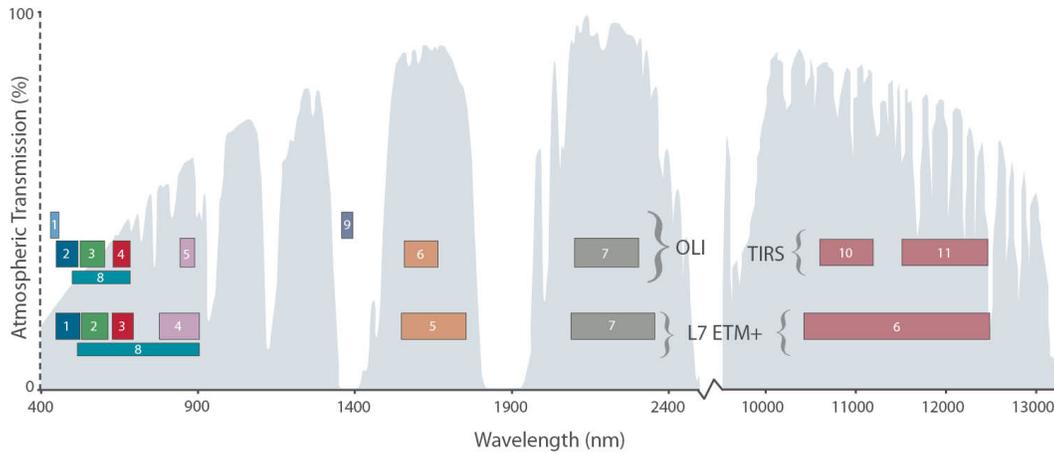


Figure 2.2: Wavelength spectrum. Credits to NASA for this image

2.2.3 Landsat mission

The Landsat mission is a program which collects images of the earth's surface through remote sensing. This is done by satellites which orbit the earth. The first satellite has been sent in orbit in the year 1972. This satellite, Landsat-1, started monitoring the globe for biospheric processes and the evolution of land-cover conditions [28]. Since then, the initial launch has been succeeded by upgraded satellites. As of today, Landsat 8 has been the latest satellite launched to orbit. The program is operated by the National Aeronautics and Space Administration (NASA) and United States Geological Survey (USGS).

The satellite sensors we retrieved data from during this research are from the Landsat 7 and Landsat 8 missions. The Landsat 7 satellite has been launched in 1999, and orbits the earth in a 16-day interval [29]. We have made use of the Surface Reflectance Tier 1 dataset for the Landsat 7 data, which measures at a 30-meter resolution. In Table B.1 the bands we have collected from Landsat 7 are specified. The newer Landsat 8 satellite has been launched in 2013, also orbiting the earth in a 16-day interval [30]. For the Landsat 8 data we accessed the Surface Reflectance Tier 1 dataset, which also has a 30-meter resolution. In Table B.2 the bands we have collected from Landsat 8 are specified. From both missions we acquired the same six bands. We have included the Blue, Green, Red, Near Infrared (NIR), Shortwave Infrared 1 (SWIR1), and the Shortwave Infrared 2 (SWIR2) spectral regions.

2.2.4 Google Earth Engine

We have collected this data by using Google Earth Engine (GEE) [31]. GEE is a cloud-based platform, operated by Google, which allows users to analyze and visualize geospatial data. [32] It can be used to integrate satellite imagery from various missions of the globe with a scientific analysis [33]. We have used the GEE API for Python to extract the spectral band values from the Landsat sensors. In accordance with the date range in which the field data was collected, we have only extracted satellite imagery from 2013 until 2017.

2.2.5 Vegetation indices

Vegetation indices have been proven to be successful in the assessment of biomass [11]. It has also been used previously to make estimations on grass height data [36, 39]. A vegetation index (VI) can be formed by combining two or more spectral bands. Jackson and Huete (1991) [11] state that that intention of VI's are to enhance the vegetation signal, while the solar irradiance and soil background effects are minimized. For this research, we have examined a number of studies where VI's have been used to estimate aboveground grassland biomass or grass height.

In Table C.1 we can see an overview of what vegetation indices, that can be formed using data from Landsat bands, have been used by these studies. From this set of indices, we have selected only some that will be used in this research. Table 2.1 shows this selection, joined with formulas used to calculate the indices.

Vegetation Index	Formula	Reference
Normalized Difference VI (NDVI)	$\frac{NIR-Red}{NIR+Red}$	Tucker (1979) [48]
Enhanced VI (EVI)	$2.5 * \frac{NIR-Red}{NIR+6*Red-7.5*Blue+1}$	Huete et al. (2002) [49]
Simple Ratio (RATIO)	$\frac{NIR}{Red}$	Tucker & Sellers (1986) [51]
Soil Adjusted VI (SAVI)	$\frac{NIR-Red}{NIR+Red+0.5} * (1 + 0.5)$	Huete (1988) [52]
Transformed VI (TVI)	$(\frac{NIR-Red}{NIR+Red} + 0.5)^{0.5}$	Tucker (1979) [48]
Normalized Difference Water Index (NDWI)	$\frac{NIR-SWIR1}{NIR+SWIR1}$	Gao (1996) [50]
Normalized Difference Tillage Index (NDTI)	$\frac{SWIR1-SWIR2}{SWIR1+SWIR2}$	van Deventer et al. (1997) [53]
Renormalized Difference VI (RDVI)	$\frac{NIR-Red}{\sqrt{(NIR+Red)}}$	Roujean & Breon (1995) [54]

Table 2.1: Vegetation indices used within this research

2.2.6 Preprocessing

To make use of this data, the locations of the band values have to be paired with the locations of the corresponding field data measurements. This has been done through a series of steps, as visualized in Figure 2.3.

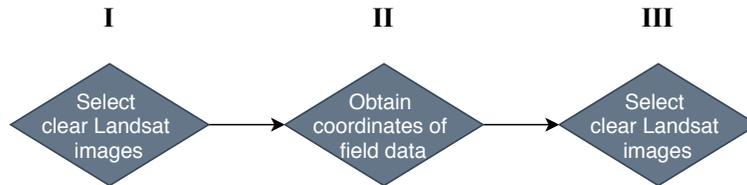


Figure 2.3: Process of pairing remote sensing and field data

Step I: Selecting clear Landsat images. During the first step of this process, we made a selection of the satellite imagery taken within a 10-day range from dates field data has been collected, that are fit for use. In this case, an image is fit for use when at least one of the plots from which the field data was collected is not covered with clouds. From those images, only the unclouded plots made the final selection. This final selection contained data from 15 different dates.

Step II: Obtaining the coordinates of field data measurements. The field data has not provided us with precise coordinates of the locations of the measurements. For this part, we have used the ArcGIS software [35]. ArcGIS is a geographic information system used for analysis of geospatial data. Through its graphic interface a line has been drawn to reproduce the paths on which the field data was collected. Thereafter, this line was split into equally long parts, resembling the field data locations. Then, a 30-meter buffer was added to these points, which formed a polygon shape for each measurement location. Lastly, the coordinates of all polygons were exported.

Step III: Extracting the band values of of the coordinates on the selected images. We developed a Python script which extracted all raw band values from the given coordinates on the selected satellite imagery. Then, these values were paired with their corresponding grass height values in a single table. Lastly, this table has been cleaned by removing all empty values.

After the remote sensing data has been paired with the field data, we developed a Python script to transform the remote sensing data to vegetation indices. After the transformation, a table was completed containing the vegetation indices from Table 2.1, raw band values and grass height.

As for the grass height data, instances containing empty values and zero-centimeter values have been cleaned from the data. The grass height values were represented by natural numbers \mathbb{N} (0,1,2,3,4...) in centimeters. This would not be an accurate representation of real grass height, so a Gaussian noise with a scale of 0.1 cm was added to the data. This has left us with a data set containing a total of 1607 instances.

2.3 Machine Learning

Machine learning algorithms learn from data so that a model that discovers patterns in the data to perform better in the future, and is able to predict unobserved behaviours [55]. In this section we will outline which machine learning algorithms are used in this research. We only looked at algorithms that offer regression models, as the purpose of this study is to estimate grass height.

2.3.1 Algorithms

For the first scenario, which tests the standard methods used in remote sensing studies, we made a selection of machine learning algorithms. In order to create this selection, we looked at remote sensing studies to find machine learning algorithms that are commonly used for the purpose of predicting AGB values. Table 2.2 shows that the Random Forest, Support Vector Machine, k-Nearest Neighbour, and Artificial Neural Network algorithms are the most commonly used, and were added to our selection of algorithms.

Study	Machine learning algorithms used
Wolanin et al. (2019) [12]	ANN, RF
Wang et al. (2019) [19]	SVM, RF
Yang et al. (2017) [17]	ANN
Zhang et al. (2015) [14]	SVM
Wu et al. (2016) [13]	KNN, RF, SVM, SGB
López-Serrano et al. (2016) [18]	KNN, RF, SVM
Xie et al. (2009) [15]	ANN
Zeng et al. (2019) [16]	RF
Wu et al. (2019) [21]	ANFIS, GMDH, ANN, SVM, GRNN
Gleason et al. (2012) [20]	LME, RF, SVM, Cubist

Table 2.2: Machine learning algorithms used in previous studies

We have also tested a novel algorithm, the Gaussian Process, to extend the scope of this experiment beyond what is standard practice in remote sensing research. What follows is a brief description of each of these algorithms.

- *Random Forest* (RF) is a machine learning algorithm first introduced by Breiman (2001) [56]. This algorithm uses a random selection of features and data instances to make binary decisions. By combining the resulting random trees, the algorithm will prevent outliers in the input data from having significant impact on the function of the model.
- *Support Vector Machines* (SVM) applies a decision boundary to a plane of data points to make classifications. What it does for non-linear data, is that it uses a function to transform the plane to a higher dimensional space. Vapnik et al. (1996) [57] have made the SVM usable for regression estimations. In the context of regression, an SVM considers the values within a certain margin from the decision boundary, and measures the errors of values outside this margin. This way the SVM creates a function which can map values within this function.
- *K-Nearest Neighbour* (KNN) is a method first proposed by Altman (1992) [58] which can be used for classification and regression purposes. To make regression estimations, the KNN algorithm assigns a continuous value to an input data instance. This value is calculated by taking the average of the target values of the k closest instances in the training set to the input instance.
- *Artificial Neural Network* (ANN) is a method which mimics the computations made in the human brain. An ANN uses a network of directed, weighted nodes. The way it works is that a signal is passed on to a node. Then, a summation of all incoming signals to the node will be passed on to an activation function. This delivers an output value. A network can consist of multiple (hidden) layers, which will always be followed in a consecutive way, making it an "input-output" situation [59]. In some cases, an artificial neural network can be referred to as deep learning. A deep learning model consists of multiple neural networks, thus having more hidden layers in its architecture.
- *Gaussian Process* (GPR) is a non-parametric algorithm used for highly non-linear data. It finds a distribution over all possible functions from the data [60]. A GPR uses a covariance matrix to find values which are similar. A covariance function shapes the functioning of the process. In combination with the matrix, the algorithm will produce output data. For a GPR there are several covariance functions, also called kernels, to use. Popular kernels include the radial-basis function (RBF) kernel and the Matérn-class kernels. [61].

2.3.2 Automated Machine Learning

Automated machine learning is a process in which certain steps of the process of applying machine learning methods will be done automatically. The general idea is that it automates the steps for data preprocessing, feature selection, algorithm selection and hyper-parameter tuning. These are complicated steps that require some knowledge in the domain of data mining and machine learning. Automated machine learning methods offer an "input-output" situation for its user. This process makes the application of machine learning more accessible to users with less knowledge about machine learning or data mining. In this research, we have made use of automated machine learning methods during scenario II. For this research, we have used two different methods: Auto-sklearn and H2O AutoML.

Auto-sklearn: The Auto-sklearn algorithm is introduced by Feurer et al. (2019) [62]. For this research we have utilised the Auto-sklearn Python package. This method employs eleven different Scikit-Learn regression machine learning algorithms, which are as follows:

- Adaboost
- K-Nearest Neighbour
- ARD Regression
- Support Vector Machine
- Decision Tree
- Random Forest
- Gaussian Process
- Ridge Regression
- Extra trees
- Stochastic Gradient Descent
- Gradient Boosting

Auto-sklearn tackles the *Combined Algorithm Selection and Hyperparameter optimization* (CASH) problem, in which the algorithm has to be picked simultaneously with the tuning of the hyper-parameters [63].

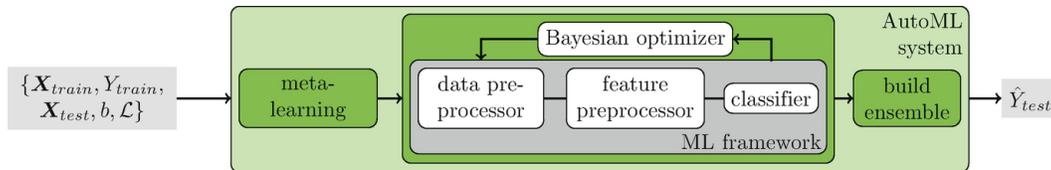


Figure 2.4: The approach of Auto-sklearn for automated machine learning, credits to Feurer et al. (2019) for this image

The Auto-sklearn approach is visualised in Figure 2.4. Firstly, the algorithm uses a meta-learning step in the process. Meta-learning is the usage of previous experience on related tasks to learn and build on. In the case of machine learning the previous experience encompasses algorithm configurations, which are evaluated based on performance results [65]. Secondly, it employs a Bayesian optimization procedure. In the context of hyper-parameter tuning, Bayesian optimization exists out of a probabilistic model and an acquisition function which decides which parameters are to be evaluated next [64]. The knowledge gained from the meta-learning step reduces the amount of time needed for this optimization [62].

At last, the Auto-sklearn algorithm builds an ensemble from the models that were searched during the Bayesian optimization. This ensemble consists of multiple models with each having used an accompanying method for preprocessing and feature selection.

H2O AutoML: The H2O AutoML is part of the machine learning platform H2O.ai [66]. For this research we have made use of its Python API. The algorithms picked by this method are developed by H2O themselves, which are the following:

- Random Forest
- Extremely Randomized Trees
- GLM
- XGBoost
- Deep learning

The approach H2O AutoML uses is different than the one used by Auto-sklearn. It uses a random grid search to find the best performing parameters for each model [67]. During a random grid search, the hyperparameters are picked at random for evaluation. The H2O AutoML method is able to produce either a stacked ensemble of all models or one from each algorithm family [68]. In contrast to Auto-sklearn, H2O AutoML is also able to produce a model based on a single algorithm. These results are presented on a leaderboard produced by the H2O AutoML method. This leaderboard shows the best performing models that have been evaluated by the algorithm.

A key difference between the two methods is that H2O AutoML offers capabilities for deep learning models, along with the option to perform a Neural Architecture Search (NAS). This allows any user to examine the relations between variables used in the deep neural network.

3 Experiments

In this section we will describe the experimental setup for each scenario described in Section 1.4. A series of experiments have been conducted for each scenario, to test the performance of the methods. In this section, we will go over the circumstances that have been present throughout all experiments, and the experiment design for each scenario we tested.

3.1 Experiment protocol

It is important to note that the same circumstances must be applied for each scenario. That means we used the same data set, method for cross validation, and amount of time consumed by the hyper-parameter optimization for all conducted experiments.

3.1.1 Data

The data set used as input is as described in Section 2.2 containing the vegetation indices from Section 2.2.5, and spectral bands from Table B.1 and B.2 as features. The grass height data as described in Section 2.2.1 has been used as the target variable.

3.1.2 Metrics

The performance of each method is based on the accuracy of the regression estimations it has produced. This performance has been evaluated by measuring the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) of the predictions. The formulas for these metrics can be found in Table 3.1. The variable n is the number of instances in the prediction, y represents the predicted values, and \hat{y} represents the true values.

Metric	Formula
Mean Absolute Error	$MAE = \left(\frac{1}{n}\right) \sum_{i=1}^n y_i - \hat{y}_i $
Root Mean Squared Error	$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n}}$
Mean Absolute Percentage Error	$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left \frac{y_i - \hat{y}_i}{\hat{y}_i} \right $

Table 3.1: Metrics used to measure model accuracy

3.1.3 Cross Validation

To test the performance of the predictions, we have split the data set into a training set containing 80% of total data, and a test set containing 20% of the total data. To ensure every model works well in general, we have made use of *k-fold cross validation*. As seen in Figure 3.1, we have used a k-value of 5 to split the training set into equally large parts. For each k-fold split we optimized the model and measured its performance against the test set.

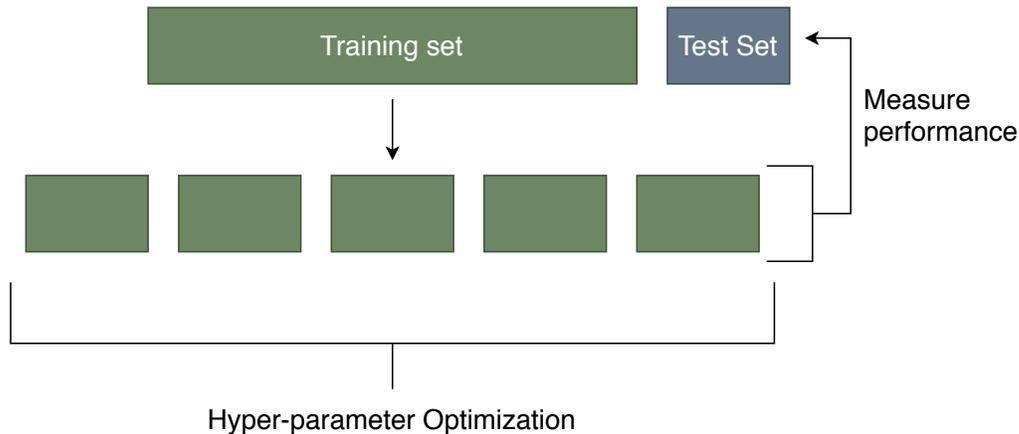


Figure 3.1: K-Folding process

3.1.4 Hyper-parameter tuning

The automated machine learning algorithms we described in Section 2.3.2 both hold the option for the user to configure the amount of time the algorithm uses to tune the hyper-parameters for its models. To ensure a fair comparison of the scenarios, we measured the average amount of time needed to complete its hyper-parameter tuning activities for the models used to test the first scenario. This average time has been applied to the automated machine learning algorithms, scaled to the amount of algorithms searched.

3.2 Experimental setup scenario I

In the first scenario we wanted to test the commonly used methods in remote sensing studies which use machine learning to estimate AGB. We will reproduce the methods used in the studies introduced in Section 1.2. In Section 2.3.1 we have established a selection of machine learning algorithms to be used for this experiment, and in Section 2.2.5 we have established which vegetation indices will be used as features. Besides the algorithms and features, we have also emulated the feature selection and hyper-parameter optimization methods from the studies we have examined.

Previous studies have used varying methods of feature selection. A number have used each extracted feature individually to evaluate the best vegetation index to use for estimations [12, 43, 41]. Others have used input data which were a combination of extracted features [19, 13]. For this research we have used both these methods. To select which individual features to use, we first correlated the vegetation indices against each other. The results may tell us which features are rudimentary to the experiment. As for the combination of vegetation indices as input features, we applied principal component analysis (PCA) to

reduce the dimensionality of the input. PCA is a technique used to simplify the description of a data set by computing new components from linear combinations of the features [69]. This prevents overfitting the model.

In this experiment, we want to evaluate the performance of each machine learning algorithm using every input type we have established. In this experiment, we have defined the testing of one input set on one machine learning algorithm as a *run*. For example, evaluating the performance of the NDVI vegetation index on the Random Forest is a run. Each algorithm has the same amount of runs, as each algorithm tests the same input data. During each run, we have used the k-folding cross validation method as mentioned in Section 3.1.3.

For each k-fold, we have used a cross-validation grid search to optimize the hyper-parameters. This method creates 5 sub-folds and searches the values of a set of parameters, and uses cross validation to determine which configuration of hyper-parameters will deliver the most accurate predictions. We have used this method for the machine learning algorithms that were commonly used in remote sensing studies: ANN, RF, SVM, KNN (as seen in Table 2.2), and the GPR. The values that have been searched can be found in Table D.1.

For each run we have produced a value for each metric described in Section 3.1.2 for every k-fold. Having these results allowed us to compare all combinations of algorithms and input type. We have also measured the average time needed to optimize the parameters on each run, so we could use comparable times for the other experiments.

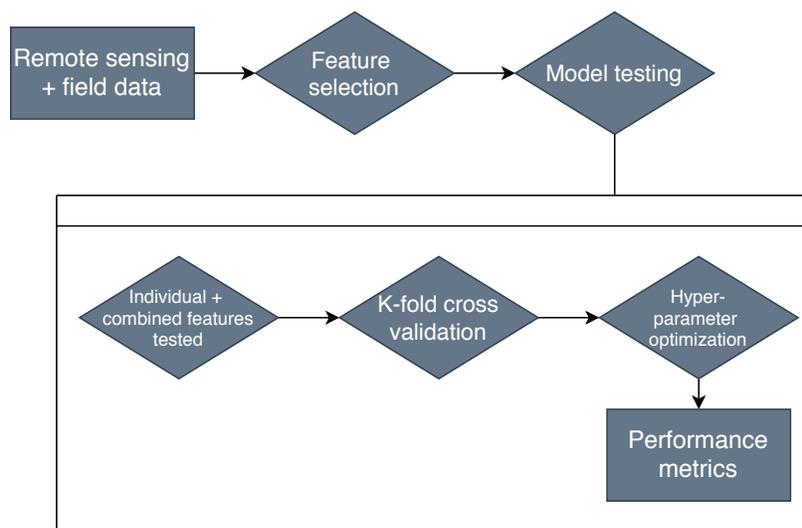


Figure 3.2: Design of the experiment setup for scenario I

3.3 Experimental setup scenario II

In Section 2.3.2 we have described the two automated machine learning algorithms we have used during the experiment for scenario II: Auto-sklearn and H2O AutoML. We have used the k-folding method as described in Section 3.1.3, meaning we have ran both algorithms for each k-fold, and collected the prediction metrics from each k-fold.

For both algorithms we needed to configure the amount of time (expressed in seconds) the algorithm takes to find the best model. The total amount of seconds we want to allocate as the search time budget was determined by the number of models searched by each algorithm. The Auto-sklearn algorithm searches eleven different models, and H2O AutoML five. We have multiplied these numbers by the average search time measured in the previous experiment.

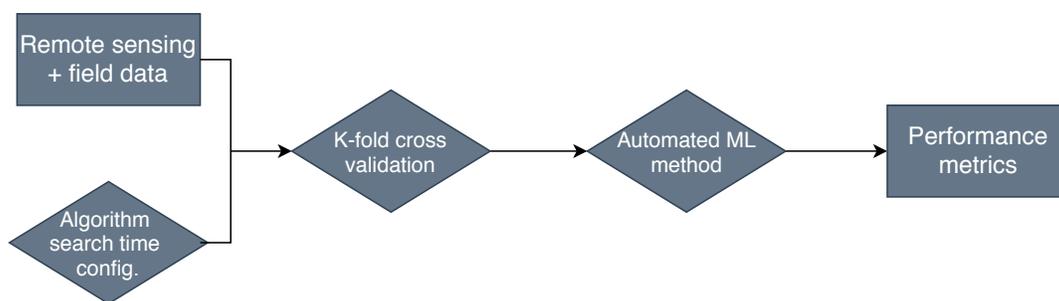


Figure 3.3: Design of the experiment setup for scenario II

3.4 Experimental setup scenario III

For this scenario we wanted to test an automated machine learning method to predict grass height using a neural network. We have only applied the raw band values as input data. The H2O AutoML algorithm supports neural networks. H2O AutoML will search a number of values for the parameters of the neural network, and applies a k-folding cross validation strategy to find the best performing neural architecture. We can find the search space in Table D.2

The H2O AutoML API allows the user to examine the results. It includes a description of the neural architecture, a table of the feature importance, and the values of various scoring metrics. It includes all metrics mentioned in Table 3.1, except for the MAPE. For this experiment it is also important to configure the search time used by the algorithm, in order to make a valid comparison with the results of the other experiments.

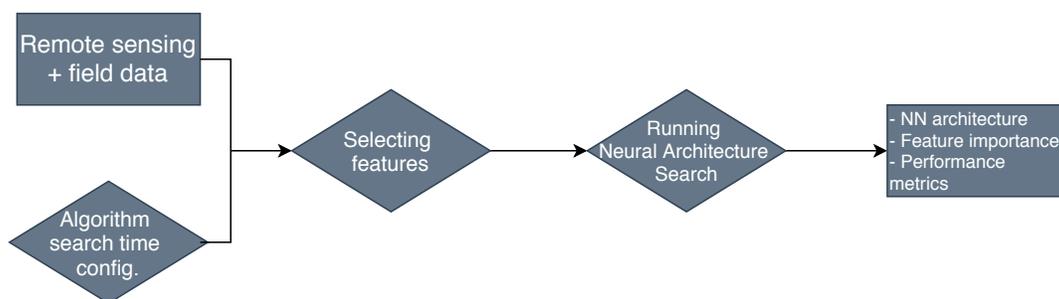


Figure 3.4: Design of the experiment setup for scenario III

4 Results

In this section, we will show the performance scores for the tests conducted as described in Section 3, and other notable results. Besides displaying the performance, we will also reflect on noteworthy patterns in the results.

4.1 Results scenario I

In Figure 4.1 we can see the correlation values between the vegetation indices in our selection. What stands out is that the NDVI, SAVI, and TVI features correlated perfectly. Since retaining all three features for our experiment will produce redundant results, two features will be discarded from the selection for this experiment. In this case, we have eliminated SAVI and TVI, and kept NDVI as it is a commonly used VI.

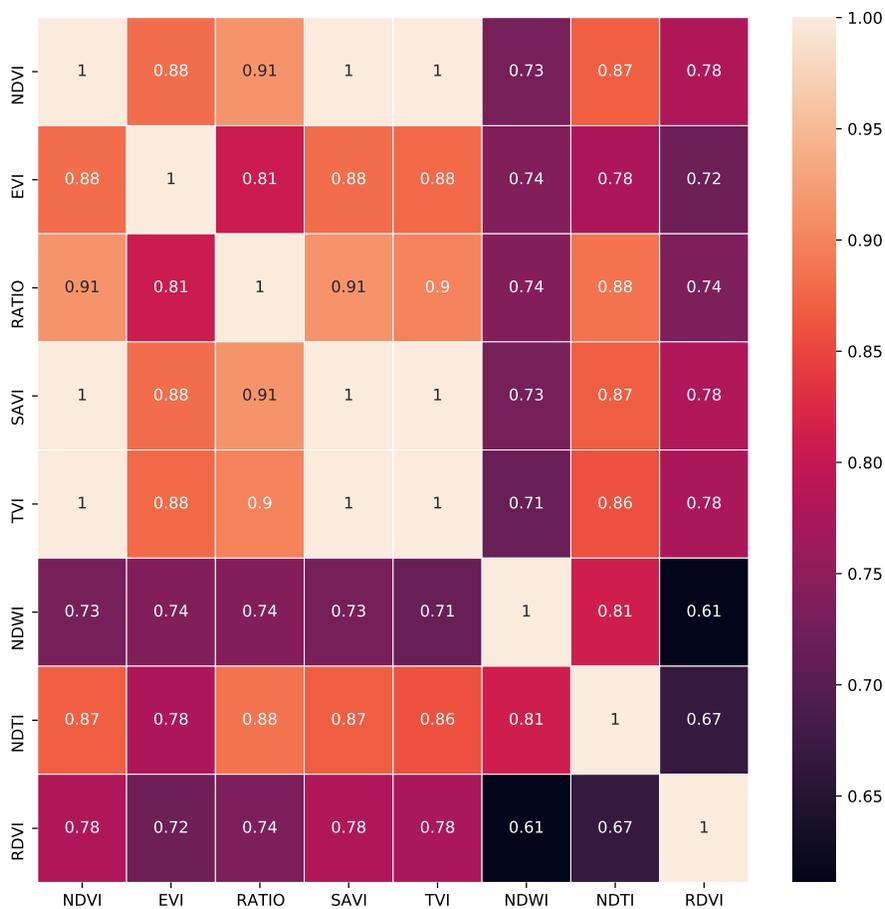


Figure 4.1: Heat map displaying the inter-correlation of vegetation indices

In Appendix E.1 we can see an overview of the MAE scores of all tested models. Appendix E.2 displays how the models scored in terms of MAPE, and Appendix E.3 for RMSE. The performance results are expressed in the form of boxplot figures. Boxplots can be interpreted in the following way: The bar in the center of each figure represents the median value, and within the box you can find 50 percent of the values. The upper and lower whiskers of the figure respectively represent the highest 25 percent and lowest 25 percent values.

To compare the performance of each algorithm, we have compared the models with the strongest performance produced by each algorithm. For each algorithm, one model produced the lowest values in terms of MAE, MAPE and RMSE. In Table 4.1 we can see which models performed the best based on the data in Table E.1. The best performing model terms of MAE and MAPE was the Support Vector Machine (SVM) using the RATIO vegetation index. In terms of RMSE we can see that the K-Nearest Neighbour using the NDVI vegetation index performed the best. Another distinct result is that the RATIO vegetation index performed well in comparison with the other inputs.

ML Algorithm	MAE	RMSE	MAPE
SVM	RATIO (2.079254)	RATIO (4.149188)	RATIO (38.599707)
RF	All VI's combined (2.393749)	All VI's combined (3.979208)	All VI's combined (57.804206)
KNN	RATIO (2.388331)	NDVI (3.951641)	RATIO (54.713806)
GPR	All VI's combined (2.676705)	All VI's combined (4.423031)	All VI's combined (80.322708)
ANN	NDVI (2.388331)	NDVI (3.961021)	NDVI (56.098928)

Table 4.1: Results of scenario I, showing the best performing model per algorithm, a cell represents the input used with the metric score in brackets

Besides the performance metrics, we have also measured the average time it took to optimize the hyper-parameters per machine learning algorithm. The average time needed to complete the optimization was 18.6 seconds. In Table 4.2 the average time per algorithm can be found.

Algorithm	Time consumption (in seconds)
ANN	24.9
KNN	22.6
RF	22.7
SVM	4.2
GPR	18.8

Table 4.2: Time consumption for optimization hyper-parameters

4.2 Results scenario II

In Section 4.1 we have determined the average search time per algorithm for the experiments conducted for scenario II to be 18.6 seconds. As the Auto-sklearn algorithm searches from 11 different machine learning algorithms, the time has been configured to 205 seconds. The H2O AutoML algorithm searches from 5 algorithms, and thus the time has been set to 93 seconds.

The performance results of both algorithms - Auto-Sklearn and H2O AutoML - have been visualised in appendix F. The scores described in the boxplots are the performance scores for each k-fold. The mean values of the results are also displayed in Table 4.3. The H2O AutoML algorithm scored better in terms of MAE, yet is out bested by the Auto-sklearn algorithm in the RMSE and MAPE scorings. There is a minimal difference between both MAE and RMSE scores, yet a huge difference in the MAPE scores.

Algorithm	MAE	RMSE	MAPE
Auto-sklearn	2.070337	3.480959	51.217733
H2O AutoML	2.042946	3.497677	131.699548

Table 4.3: Results of scenario II, showing the mean performance metrics of the H2O AutoML and Auto-sklearn automated machine learning algorithms

4.3 Results scenario III

In this section we will describe not only the results of the experiment for scenario III in terms of metrics. The resulting neural network architecture and feature importance will be specified as well. For this experiment we have also allocated a time budget. The H2O AutoML algorithm only searches the deep neural network for this case. Because of this, it has been allocated 18.6 seconds, which is the average time it took for algorithms in scenario I to optimize its parameters. In Figure G.1 and Figure G.2 the performance scores of the neural network are represented as a boxplot visualization. As the API for this feature of the H2O AutoML did not offer a performance score in the form of MAPE, we only possessed the MAE and RMSE scores. In Table 4.4 we can see that this experiment resulted in an MAE of 2.679430 and an RMSE of 4.421356.

Algorithm	MAE	RMSE
H2O AutoML Neural Network	2.679430	4.421356

Table 4.4: Results of scenario III, showing the performance metrics for the Neural Network using the H2O autoML algorithm

The resulting Neural Network Architecture produced by the H2O AutoML algorithm is described in Table G.1. The table describes the layout of the neural network by giving the number of nodes (neurons) per hidden layer in the network, along with the mean weight and bias of the nodes. In Table 4.5 the feature importance of the input data is displayed. This table shows how often each spectral band is used in the neural network to make a prediction on grass height data. This describes the relevance of each spectral band in the model.

Band name	Importance	Percentage used
NIR	1.000000	0.204854
SWIR2	0.873306	0.178900
BLUE	0.817366	0.167441
RED	0.774636	0.158687
SWIR1	0.748654	0.153365
GREEN	0.667571	0.126754

Table 4.5: Feature importance for the spectral bands

4.4 Comparison

In this section we will compare the results of all three scenarios. This will give us insights to the performance of the different methods we have tested under the same circumstances. In Figure 4.2 the Mean Absolute Error performances of the models are displayed. The models are ranked, the best performing model on the left and the least performing on the right. The H2O AutoML seems to have scored the best on this metric. In Figure 4.3 the models are ranked in the same fashion. The Support Vector Machine (SVM) using the RATIO vegetation index as input has scored the best on this metric, while the H2O AutoML algorithm is by far the worst performing model. Figure 4.4 illustrates a comparison of the different models with regard to the Root Mean Squared Error. There was not much of a difference in scoring between the models, although we can see the two models tested in scenario II performed the best.

Overall, we can see that the models from scenario II generally outperformed those from scenario I and III, except for the H2O AutoML MAPE value. The models from scenario I largely stayed in the mid-range of all ranking, with an exception from the Support Vector Machine, having performed the best in its category. Lastly, there is the model from scenario III having had the overall worst performances as can be seen in the aforementioned figures.

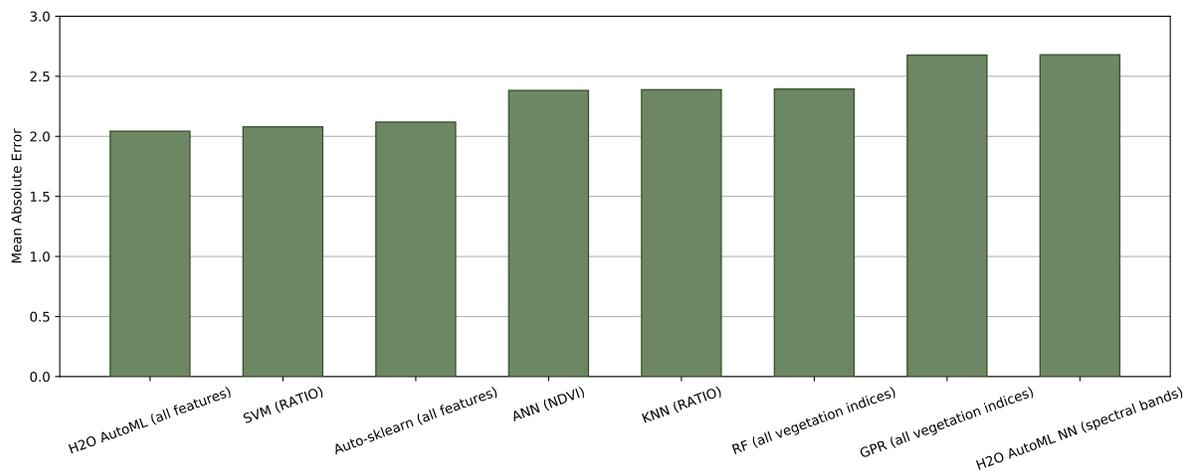


Figure 4.2: Mean Absolute Error comparison models (Best performing ranked left to right)

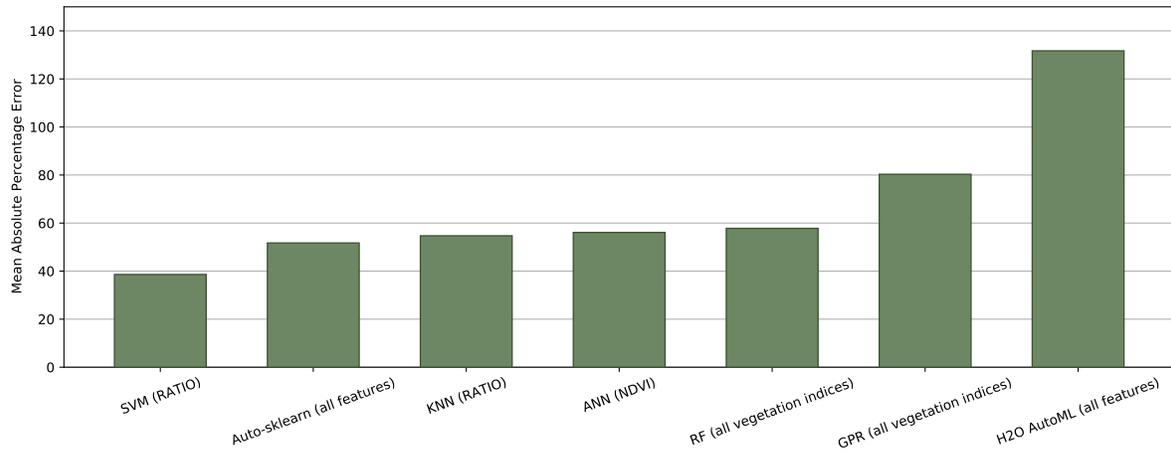


Figure 4.3: Mean Absolute Percentage Error comparison models (Best performing ranked left to right)

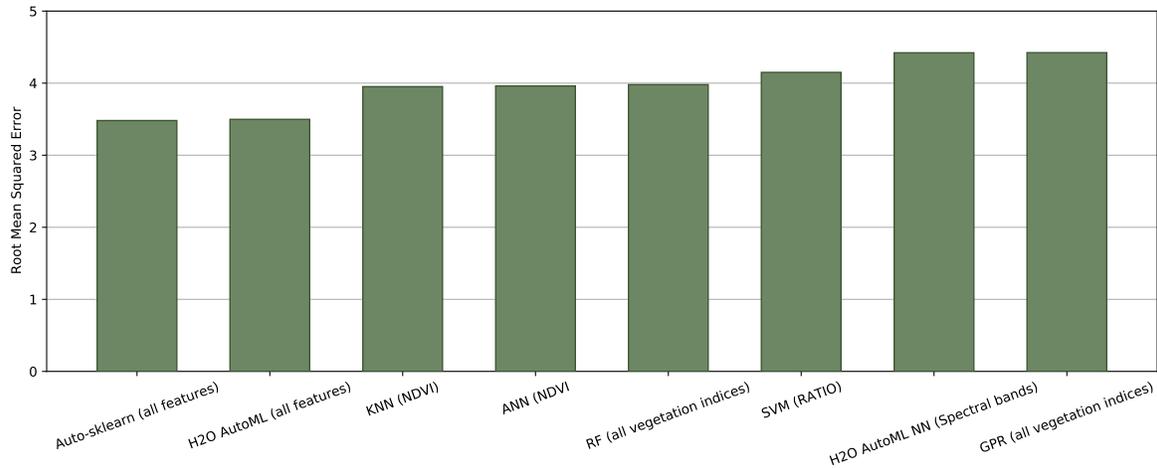


Figure 4.4: Root Mean Squared Error comparison models (Best performing ranked left to right)

4.5 Limitations

A number of limitations have been tied to this research, which impacted the data we have used. First, no reference data for AGB in the OVP was available. This resulted in grass height data being used as prediction variable instead of biomass. Secondly, the field data measurements have not been provided with actual coordinates. That also applies to the start- and endpoint of the visualization of the indication where the field data was collected. An accurate indication of the location of field data could impact the reliability of the results in a positive way. Third, the cloudiness of satellite images severely limited the amount of data available. A lack of clear weather prevents the satellite sensor from collecting reflectance light. Out of 23 dates field data were acquired on, only data from 15 of these have been suitable for at least one plot.

5 Conclusions

5.1 Discussion

We have researched three different approaches to predict grass height using remote sensing data and machine learning methods. We did this to find out whether data-driven approaches were as effective as the standard remote sensing practices in making predictions. These data-driven approaches in this case were firstly automated machine learning methods, and secondly a neural network using only raw spectral band values instead of derived vegetation indices. We have used remotely sensed data from the Landsat 7 and Landsat 8 satellites. This discovery could introduce new approaches for estimation of information such as biomass and grass height. In this section, we will discuss the results of all tested scenarios, and compare them.

Looking at the first scenario, we can see that the SVM using the RATIO VI scored the best in its category in terms of MAE (2.08 cm) and MAPE (38.60%). The RF algorithm using a combination of all VI's scored the best in terms of RMSE (3.98 cm). A study using data from the Sentinel-2 satellite has scored better, making grass height estimations with an RMSE of 2.89 cm [71]. Another study using an ultrasonic scanner estimated the vegetation height of Alfalfa at a MAPE of 0.30%, and Bermudagrass at a MAPE of 0.21% [36]. A study using LIDAR data was able to predict sward height at an RMSE of 3.98 cm [70], exactly the same as the RF algorithm using a combination of all VI's used. This implies that using data from an ultrasonic scanner or Sentinel-2 could improve our results.

As for the second scenario, we can see that that the Auto-sklearn algorithm outperformed the model made by Yuan et al. (2018) [70] by estimating grass height at an RMSE of 3.48 cm. It is outperformed by Pittman et al. (2015) [36] and Cimbelli et al. (2017) [71]. To improve the results for automated machine learning algorithms, we could firstly use other sources of remotely sensed data such as Sentinel-2 and the ultrasonic scanner. Another is to delimit the algorithm search time parameter of both the Auto-sklearn and H2O AutoML algorithms. For this research we have capped the search time to ensure a fair comparison across each scenario. Delimiting this parameter could produce a more accurate model, as the algorithm would have more time to search for optimal settings for feature selection, preprocessing, model selection and hyper-parameter configuration.

As for the third scenario, if we look at the predictive performance we can see it scored poor in comparison with the aforementioned studies, having an MAE of 2.68 cm and an RMSE of 4.42 cm. Also for this method it is possible to perform better if we use data from different sources, and delimit the search time for the H2O AutoML algorithm, which is used for this scenario to build a neural network. For this scenario, we have also researched the relationships between the spectral bands in this neural network. If we look at Table 4.5 we can see the NIR band is the most important band, which is expected. If we compare it with the bands that are most used in vegetation indices, referring to Table 2.1, we can see that the blue band is not common, yet holds a higher importance than the red band, which is more common. All in all, the importance ranking corresponds to the knowledge known in the domain of remote sensing. This implies that a neural network only using raw spectral band data formulates relationships which are used to transform the data to vegetation indices. This means that such a neural network can be used to predict vegetation height or biomass, without having to derive vegetation indices from the remote sensing data.

We have compared all approaches in similar circumstances, and referring to the results we can assume that automated machine learning algorithms such as Auto-sklearn and H2O AutoML are more effective at predicting the grass height in the Oostvaardersplassen than the conventional methods, in which all steps in the machine learning process have to be done manually. The automated machine learning algorithms will be more suitable for making predictions based on remotely sensed data, both for its high performance and "input-output" situation making it more accessible for researchers without ample machine learning knowledge. This does not mean the usage of manually configured machine learning algorithms should be abandoned. As we can see the SVM outperformed both automated machine learning algorithms in terms of MAPE.

For predicting the grass height in the Oostvaardersplassen, either of these methods are suitable. To predict the AGB for locations in the Oostvaardersplassen, one could apply the methods as described in this thesis. The methods would require reference field data in the form of the weight of dry pasture, and remotely sensed data measured from the locations where this field data was taken from.

5.2 Conclusion

We have successfully demonstrated that remote sensing data from the Landsat 7 and Landsat 8 satellites can be used to predict vegetation height, making use of manual and automated machine learning methods. Automated machine learning methods such as Auto-sklearn and H2O AutoML are the best pick to make these predictions. This research has also demonstrated that a neural network can be used to create relationships in raw spectral data which are often used in remote sensing studies.

5.3 Further research

This research has laid the groundwork for various other potential researches. First, the full potential of automated machine learning used to make predictions in the remote sensing field can be researched. In this research, we have limited the performance of the automated machine learning algorithms in order to assure the circumstances were consistent across all experiments. We have done this by setting a limit to the time span the algorithm uses for the feature selection, preprocessing, model selection and hyper-parameter optimization. Future research could study the potential of these algorithms when they have less limiting configurations. The same is implied for the neural network using spectral bands as input created by the H2O AutoML, as described in Section 3.4. Even though the performance for this approach is weak, these results paved a way for further research.

Secondly, as the Oostvaardersplassen has been the main study site for this research, and all data was collected from this location, this thesis can be the basis for further research on this location. As a prediction of future food availability is valuable information for the park management, future research could create a prediction of future grass height values using the methods from this research. Besides the Oostvaarder-splassen, the scale of the research can also consist of the entire Netherlands. Remote sensing data can be retrieved from Landsat satellites, which can be freely accessed using Google Earth Engine. As long as reference vegetation height data exists, the methods described in this thesis can be used for estimations.

References

- [1] Zhao, F., Xu, B., Yang, X., Jin, Y., Li, J., Xia, L., Chen, S., & Ma, H. (2014). Remote Sensing Estimates of Grassland Aboveground Biomass Based on MODIS Net Primary Productivity (NPP): A Case Study in the Xilingol Grassland of Northern China. *Remote Sensing*, 6(6), 5368–5386. <https://doi.org/10.3390/rs6065368>
- [2] Li, G., Wang, J., Wang, Y., Wei, H., Ochir, A., Davaasuren, D., Chonokhuu, S., & Nasanbat, E. (2019). Spatial and Temporal Variations in Grassland Production from 2006 to 2015 in Mongolia Along the China–Mongolia Railway. *Sustainability*, 11(7), 2177. <https://doi.org/10.3390/su11072177>
- [3] Lu, D. (2006). The potential and challenge of remote sensing-based biomass estimation. *International Journal of Remote Sensing*, 27(7), 1297–1328. <https://doi.org/10.1080/01431160500486732>
- [4] T. Vashum, K. (2012). Methods to Estimate Above-Ground Biomass and Carbon Stock in Natural Forests - A Review. *Journal of Ecosystem & Ecography*, 02(04), 1–7. <https://doi.org/10.4172/2157-7625.1000116>
- [5] Kumar, L., & Mutanga, O. (2017). Remote Sensing of Above-Ground Biomass. *Remote Sensing*, 9(9), 935. <https://doi.org/10.3390/rs9090935>
- [6] Vera, F.W. (2009). Large-scale nature development – the Oostvaardersplassen.
- [7] Lorimer, J., & Driessen, C. (2013). Wild experiments at the Oostvaardersplassen: rethinking environmentalism in the Anthropocene. *Transactions of the Institute of British Geographers*, 39(2), 169–181. <https://doi.org/10.1111/tran.12030>
- [8] NOS. (2018, April 25). Oostvaardersplassen voor dummies: waarom, hoe, wat en hoe nu verder? <https://nos.nl/artikel/2228894-oostvaardersplassen-voor-dummies-waarom-hoe-wat-en-hoe-nu-verder.html>
- [9] NOS. (2018, April 5). Meer dan de helft van de grote grazers in Oostvaardersplassen dood. <https://nos.nl/artikel/2225884-meer-dan-de-helft-van-de-grote-grazers-in-oostvaardersplassen-dood.html>
- [10] Gaitán, J. J., Bran, D., Oliva, G., Ciari, G., Nakamatsu, V., Salomone, J., Ferrante, D., Buono, G., Massara, V., Humano, G., Celdrán, D., Opazo, W., & Maestre, F. T. (2013). Evaluating the performance of multiple remote sensing indices to predict the spatial variability of ecosystem structure and functioning in Patagonian steppes. *Ecological Indicators*, 34, 181–191. <https://doi.org/10.1016/j.ecolind.2013.05.007>
- [11] Jackson, R. D., & Huete, A. R. (1991). Interpreting vegetation indices. *Preventive Veterinary Medicine*, 11(3–4), 185–200. [https://doi.org/10.1016/s0167-5877\(05\)80004-2](https://doi.org/10.1016/s0167-5877(05)80004-2)
- [12] Wolanin, A., Camps-Valls, G., Gómez-Chova, L., Mateo-García, G., van der Tol, C., Zhang, Y., & Guanter, L. (2019). Estimating crop primary productivity with Sentinel-2 and Landsat 8 using machine learning methods trained with radiative transfer simulations. *Remote Sensing of Environment*, 225, 441–457. <https://doi.org/10.1016/j.rse.2019.03.002>

- [13] Chaofan Wu, Huanhuan Shen, Aihua Shen, Jinsong Deng, Muye Gan, Jinxia Zhu, Hongwei Xu, Ke Wang, “Comparison of machine-learning methods for above-ground biomass estimation based on Landsat imagery,” *J. Appl. Remote Sens.* 10(3), 035010 (2016), doi: 10.1117/1.JRS.10.035010.
- [14] Zhang, B., Zhang, L., Xie, D., Yin, X., Liu, C., & Liu, G. (2015). Application of Synthetic NDVI Time Series Blended from Landsat and MODIS Data for Grassland Biomass Estimation. *Remote Sensing*, 8(1), 10. <https://doi.org/10.3390/rs8010010>
- [15] Xie, Y., Sha, Z., Yu, M., Bai, Y., & Zhang, L. (2009). A comparison of two models with Landsat data for estimating above ground grassland biomass in Inner Mongolia, China. *Ecological Modelling*, 220(15), 1810–1818. <https://doi.org/10.1016/j.ecolmodel.2009.04.025>
- [16] Zeng, N., Ren, X., He, H., Zhang, L., Zhao, D., Ge, R., . . . Niu, Z. (2019). Estimating grassland aboveground biomass on the Tibetan Plateau using a random forest algorithm. *Ecological Indicators*, 102, 479–487. <https://doi.org/10.1016/j.ecolind.2019.02.023>
- [17] Yang, S., *Remote Sensing of Environment* (2017), <http://dx.doi.org/10.1016/j.rse.2017.10.011>
- [18] López-Serrano, P. M., López-Sánchez, C. A., Álvarez-González, J. G., & García-Gutiérrez, J. (2016). A Comparison of Machine Learning Techniques Applied to Landsat-5 TM Spectral Data for Biomass Estimation. *Canadian Journal of Remote Sensing*, 42(6), 690–705. doi:10.1080/07038992.2016.1217485
- [19] Wang, J., Xiao, X., Bajgain, R., Starks, P., Steiner, J., Doughty, R. B., & Chang, Q. (2019b). Estimating leaf area index and aboveground biomass of grazing pastures using Sentinel-1, Sentinel-2 and Landsat images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 154, 189–201. <https://doi.org/10.1016/j.isprsjprs.2019.06.007>
- [20] Gleason, C. J., & Im, J. (2012). Forest biomass estimation from airborne LiDAR data using machine learning approaches. *Remote Sensing of Environment*, 125, 80–91. doi:10.1016/j.rse.2012.07.006
- [21] Wu, C., Chen, Y., Peng, C., Li, Z., & Hong, X. (2019). Modeling and estimating aboveground biomass of *Dacrydium pierrei* in China using machine learning with climate change. *Journal of Environmental Management*, 234, 167–179. doi:10.1016/j.jenvman.2018.12.090
- [22] K. Tanaka, A. Monden and Z. Yücel, ”Prediction of Software Defects Using Automated Machine Learning,” 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Toyama, Japan, 2019, pp. 490-494, doi: 10.1109/SNPD.2019.8935839.
- [23] Rijkswaterstaat, & Cornelissen, P. (2007, January). Vegetatie en grote herbivoren in de randzone van de Oostvaardersplassen. <https://library.wur.nl/ebooks/hydrotheek/1885367.pdf>
- [24] Nijland, R. (2008). Nieuwe Ontwikkelingsvisie Oostvaardersplassen; fictie of feiten?. *De Levende Natuur*, 109(4), 147-149.
- [25] Dept of Physical Geography and Ecosystem Science, & Palm, E. (2013, November). Finding a method for simplified biomass measurements on Sahelian grasslands (No. 286). <http://lup.lub.lu.se/student-papers/record/4145588>

- [26] SAHRAMAA, M., IHAMÄKI, H., & JAUHAINEN, L. (2003). Variation in biomass related variables of reed canary grass. *Agricultural and Food Science*, 12(3–4), 213–225. <https://doi.org/10.23986/afsci.5749>
- [27] CATCHPOLE, W. R., & WHEELER, C. J. (1992). Estimating plant biomass: A review of techniques. *Austral Ecology*, 17(2), 121–131. <https://doi.org/10.1111/j.1442-9993.1992.tb00790.x>
- [28] Williams, D. L., Goward, S., & Arvidson, T. (2006). Landsat. *Photogrammetric Engineering & Remote Sensing*, 72(10), 1171–1178. <https://doi.org/10.14358/pers.72.10.1171>
- [29] NASA. (n.d.). Landsat 7 - Landsat Science. [Landsat.Gsfc.Nasa.Gov. https://landsat.gsfc.nasa.gov/landsat-7/](https://landsat.gsfc.nasa.gov/landsat-7/)
- [30] Landsat 8. (n.d.). USGS. https://www.usgs.gov/land-resources/nli/landsat/landsat-8?qt-science_support_page_related_con=0#qt-science_support_page_related_con
- [31] GEEcite: Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*.
- [32] Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, 202, 18–27. <https://doi.org/10.1016/j.rse.2017.06.031>
- [33] FAQ – Google Earth Engine. (n.d.). Google Earth Engine. <https://earthengine.google.com/faq/>
- [34] Humboldt State University. (n.d.). Image Bands. [Gsp.Humboldt.Edu. http://gsp.humboldt.edu/OLM/Courses/GSP_216_Online/lesson3-1/bands.html](http://gsp.humboldt.edu/OLM/Courses/GSP_216_Online/lesson3-1/bands.html)
- [35] ESRI 2020. ArcGIS Desktop: Release 10.6.1. Redlands, CA: Environmental Systems Research Institute.
- [36] Pittman, J., Arnall, D., Interrante, S., Moffet, C., & Butler, T. (2015). Estimation of Biomass and Canopy Height in Bermudagrass, Alfalfa, and Wheat Using Ultrasonic, Laser, and Spectral Sensors. *Sensors*, 15(2), 2920–2943. <https://doi.org/10.3390/s150202920>
- [37] Psomas, A., Kneubühler, M., Huber, S., Itten, K., & Zimmermann, N. E. (2011). Hyperspectral remote sensing for estimating aboveground biomass and for exploring species richness patterns of grassland habitats. *International Journal of Remote Sensing*, 32(24), 9007–9031. <https://doi.org/10.1080/01431161.2010.532172>
- [38] Peng, Y., Gitelson, A. A., & Sakamoto, T. (2013). Remote estimation of gross primary productivity in crops using MODIS 250m data. *Remote Sensing of Environment*, 128, 186–196. <https://doi.org/10.1016/j.rse.2012.10.005>
- [39] J. O. Payero, C. M. U. Neale, & J. L. Wright. (2004). COMPARISON OF ELEVEN VEGETATION INDICES FOR ESTIMATING PLANT HEIGHT OF ALFALFA AND GRASS. *Applied Engineering in Agriculture*, 20(3), 385–393. <https://doi.org/10.13031/2013.16057>

- [40] Frank, A. B., & Aase, J. K. (1994). Residue effects on radiometric reflectance measurements of Northern Great Plains Rangelands. *Remote Sensing of Environment*, 49(3), 195–199. [https://doi.org/10.1016/0034-4257\(94\)90015-9](https://doi.org/10.1016/0034-4257(94)90015-9)
- [41] KARAKOÇ, A., & KARABULUT, M. (2019). Ratio-based vegetation indices for biomass estimation depending on grassland characteristics. *TURKISH JOURNAL OF BOTANY*, 43(5), 619–633. <https://doi.org/10.3906/bot-1902-50>
- [42] Jansen, V., Kolden, C., & Schmalz, H. (2018). The Development of Near Real-Time Biomass and Cover Estimates for Adaptive Rangeland Management Using Landsat 7 and Landsat 8 Surface Reflectance Products. *Remote Sensing*, 10(7), 1057. <https://doi.org/10.3390/rs10071057>
- [43] Lussem, U., Bolten, A., Menne, J., Gnyp, M. L., Schellberg, J., & Bareth, G. (2019). Estimating biomass in temperate grassland with high resolution canopy surface models from UAV-based RGB images and vegetation indices. *Journal of Applied Remote Sensing*, 13(03), 1. <https://doi.org/10.1117/1.jrs.13.034525>
- [44] Hoving, I. E., Starmans, D. A. J., Booij, J. A., Kuiper, I., & Holshof, G. (2019). Amazing Grazing: grass growth measurements with remote sensing techniques. *Grassland Science in Europe*, 23, 860–862. https://www.amazinggrazing.eu/upload_mm/2/9/c/0ef04282-6151-4eba-b977-bcee8d033770.Paper_EGF2018_Amazing%20Grazing-grass%20growth%20measurements%20with%20remote.pdf
- [45] Vescovo, L., Zorer, R., Belli, C., Cescatti, A., & Gianelle, D. (2004). Use of vegetation indexes to predict biomass and LAI of Trentino grasslands . *Land Use Systems in Grassland Dominated Regions*, 9, 811–813.
- [46] Cho, M. A., & Skidmore, A. K. (2008). Hyperspectral predictors for monitoring biomass production in Mediterranean mountain grasslands: Majella National Park, Italy. *International Journal of Remote Sensing*, 30(2), 499–515. <https://doi.org/10.1080/01431160802392596>
- [47] Prabhakara, K., Hively, W. D., & McCarty, G. W. (2015). Evaluating the relationship between biomass, percent groundcover and remote sensing indices across six winter cover crop fields in Maryland, United States. *International Journal of Applied Earth Observation and Geoinformation*, 39, 88–102. doi:10.1016/j.jag.2015.03.002
- [48] Tucker, C. J. (1979). Red and photographic infrared linear combinations for monitoring vegetation. *Remote Sensing of Environment*, 8(2), 127–150. [https://doi.org/10.1016/0034-4257\(79\)90013-0](https://doi.org/10.1016/0034-4257(79)90013-0)
- [49] Huete, A., Didan, K., Miura, T., Rodriguez, E. P., Gao, X., & Ferreira, L. G. (2002). Overview of the radiometric and biophysical performance of the MODIS vegetation indices. *Remote Sensing of Environment*, 83(1–2), 195–213. [https://doi.org/10.1016/s0034-4257\(02\)00096-2](https://doi.org/10.1016/s0034-4257(02)00096-2)
- [50] Gao, B. (1996). NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space. *Remote Sensing of Environment*, 58(3), 257–266. [https://doi.org/10.1016/s0034-4257\(96\)00067-3](https://doi.org/10.1016/s0034-4257(96)00067-3)
- [51] TUCKER, C. J., & SELLERS, P. J. (1986). Satellite remote sensing of primary production. *International Journal of Remote Sensing*, 7(11), 1395–1416. <https://doi.org/10.1080/01431168608948944>

- [52] Huete, A. R. (1988). A soil-adjusted vegetation index (SAVI). *Remote Sensing of Environment*, 25(3), 295–309. [https://doi.org/10.1016/0034-4257\(88\)90106-x](https://doi.org/10.1016/0034-4257(88)90106-x)
- [53] Deventer, A.P.; Ward, A.D.; Gowda, P.H.; Lyon, J.G. Using thematic mapper data to identify contrasting soil plains and tillage practices. *Photogramm. Eng. Remote Sens.* 1997, 63, 87–93.
- [54] Roujean, J.-L., & Breon, F.-M. (1995). Estimating PAR absorbed by vegetation from bidirectional reflectance measurements. *Remote Sensing of Environment*, 51(3), 375–384. [https://doi.org/10.1016/0034-4257\(94\)00114-3](https://doi.org/10.1016/0034-4257(94)00114-3)
- [55] Witten, I., & Frank, E. (2011). *Data Mining: Practical Machine Learning Tools and Techniques* (3rd ed.). Morgan Kaufmann.
- [56] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/a:1010933404324>
- [57] Vladimir Vapnik, Steven E. Golowich, and Alex Smola. 1996. Support vector method for function approximation, regression estimation and signal processing. In *Proceedings of the 9th International Conference on Neural Information Processing Systems (NIPS'96)*. MIT Press, Cambridge, MA, USA, 281–287.
- [58] Altman, N. S. (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician*, 46(3), 175–185. doi:10.1080/00031305.1992.10475879
- [59] Hsu, K., Gupta, H. V., & Sorooshian, S. (1995). Artificial Neural Network Modeling of the Rainfall-Runoff Process. *Water Resources Research*, 31(10), 2517–2530. doi:10.1029/95wr01955
- [60] Schulz, E., Speekenbrink, M., & Krause, A. (2018). A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*, 85, 1–16. doi:10.1016/j.jmp.2018.03.001
- [61] Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. Amsterdam University Press.
- [62] Feurer, M., Klein, A., Eggensperger, K., Springenberg, J. T., Blum, M., & Hutter, F. (2019). Auto-sklearn: efficient and robust automated machine learning. In *Automated Machine Learning* (pp. 113-134). Springer, Cham.
- [63] Kotthoff L., Thornton C., Hoos H.H., Hutter F., Leyton-Brown K. (2019) Auto-WEKA: Automatic Model Selection and Hyperparameter Optimization in WEKA. In: Hutter F., Kotthoff L., Vanschoren J. (eds) *Automated Machine Learning*. The Springer Series on Challenges in Machine Learning. Springer, Cham. https://doi.org/10.1007/978-3-030-05318-5_4
- [64] Hutter, F., Kotthoff, L., & Vanschoren, J. (2019). *Automated Machine Learning* (p. 9). Springer Publishing.
- [65] Vanschoren J. (2019) Meta-Learning. In: Hutter F., Kotthoff L., Vanschoren J. (eds) *Automated Machine Learning*. The Springer Series on Challenges in Machine Learning. Springer, Cham. https://doi.org/10.1007/978-3-030-05318-5_2

- [66] H2O.ai. (2020) h2o: Python Interface for H2O. Python package version 3.30.0.6. <https://github.com/h2oai/h2o-3>.
- [67] Gijsbers, P., LeDell, E., Thomas, J., Poirier, S., Bischl, B., & Vanschoren, J. (2019). An open source AutoML benchmark. arXiv preprint arXiv:1907.00909.
- [68] Halvari, T., Nurminen, J. K., & Mikkonen, T. (2020). Testing the Robustness of AutoML Systems. arXiv preprint arXiv:2005.02649.
- [69] Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3), 37-52. [https://doi.org/10.1016/0169-7439\(87\)80084-9](https://doi.org/10.1016/0169-7439(87)80084-9)
- [70] Yuan, W., Li, J., Bhatta, M., Shi, Y., Baenziger, P., & Ge, Y. (2018). Wheat Height Estimation Using LiDAR in Comparison to Ultrasonic Sensor and UAS. *Sensors*, 18(11), 3731. <https://doi.org/10.3390/s18113731>
- [71] Cimbelli, A., & Vitale, V. (2017). Grassland Height Assessment by Satellite Images. *Advances in Remote Sensing*, 06(01), 40-53. <https://doi.org/10.4236/ars.2017.61003>

Appendices

A Field data locations

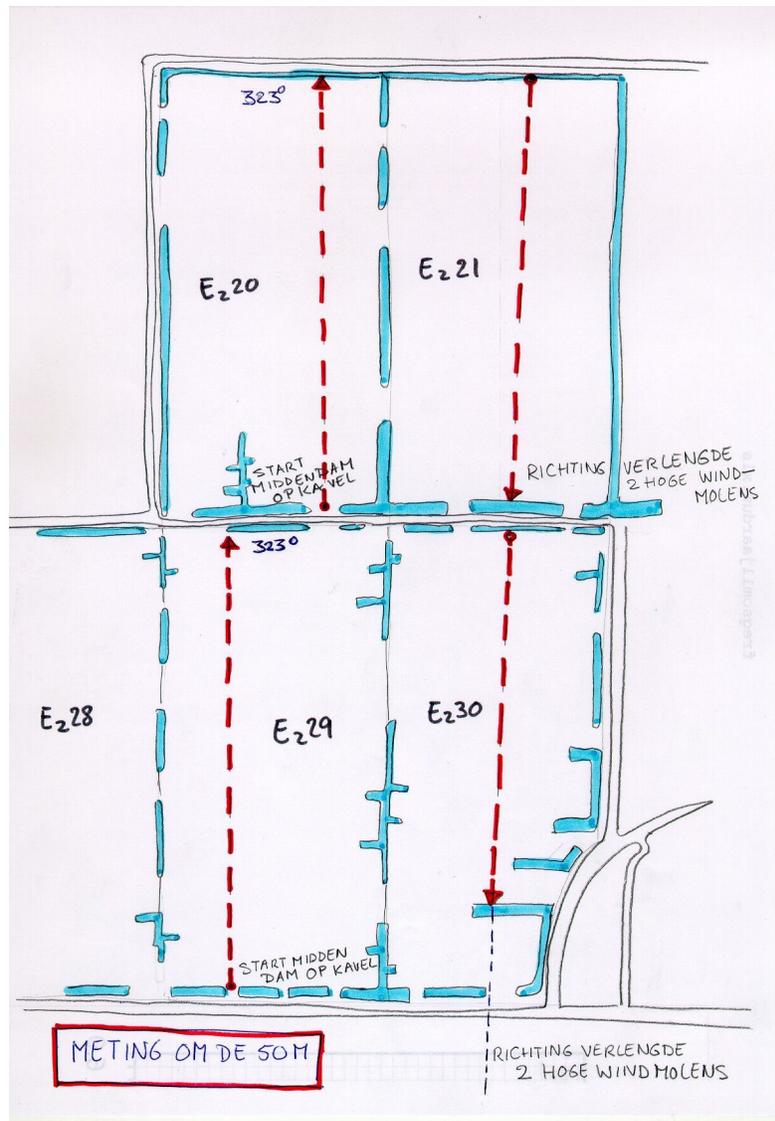


Figure A.1: Transects E28, E30, E29, E20, E21

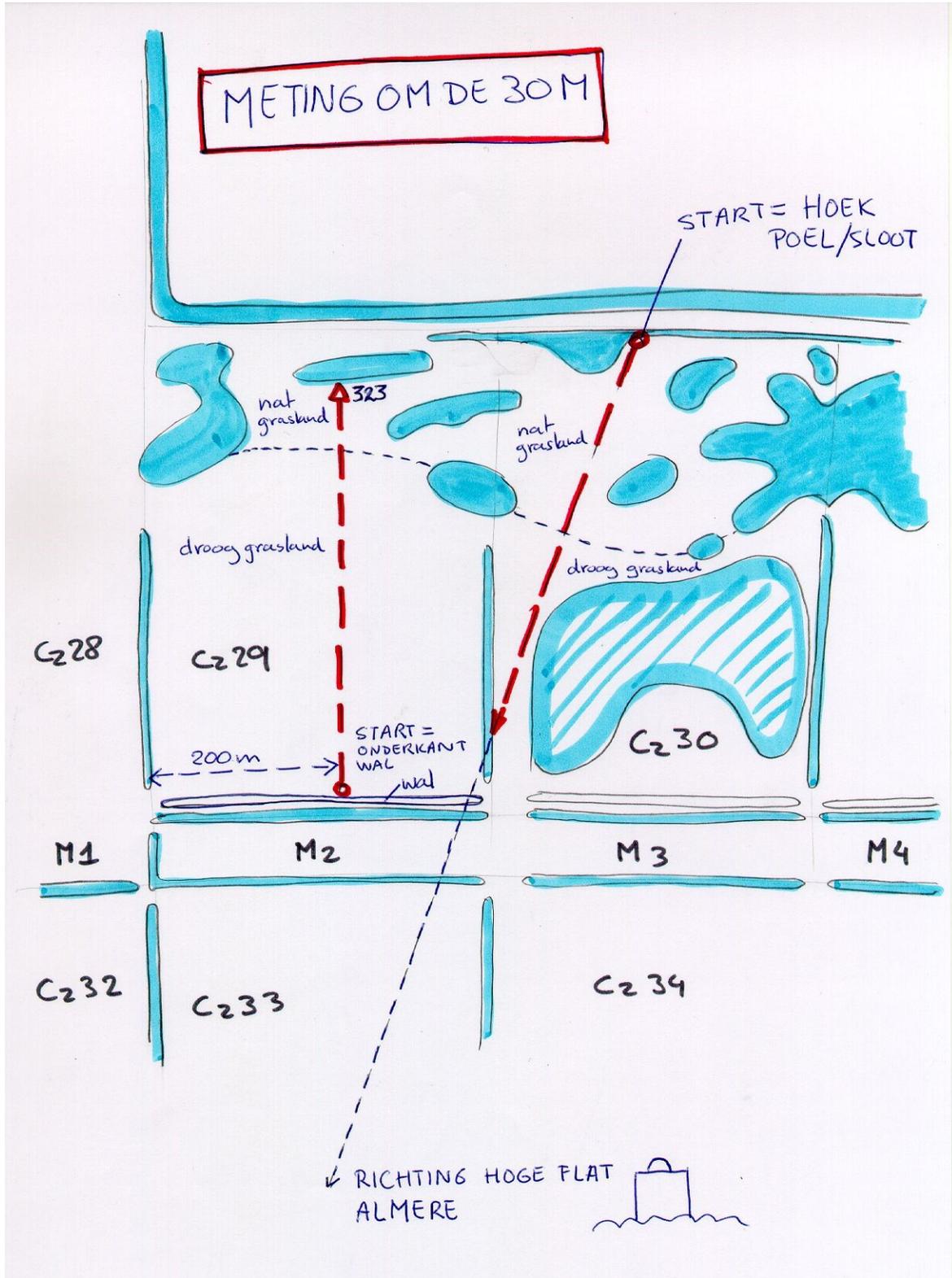


Figure A.2: Transects C29, C30

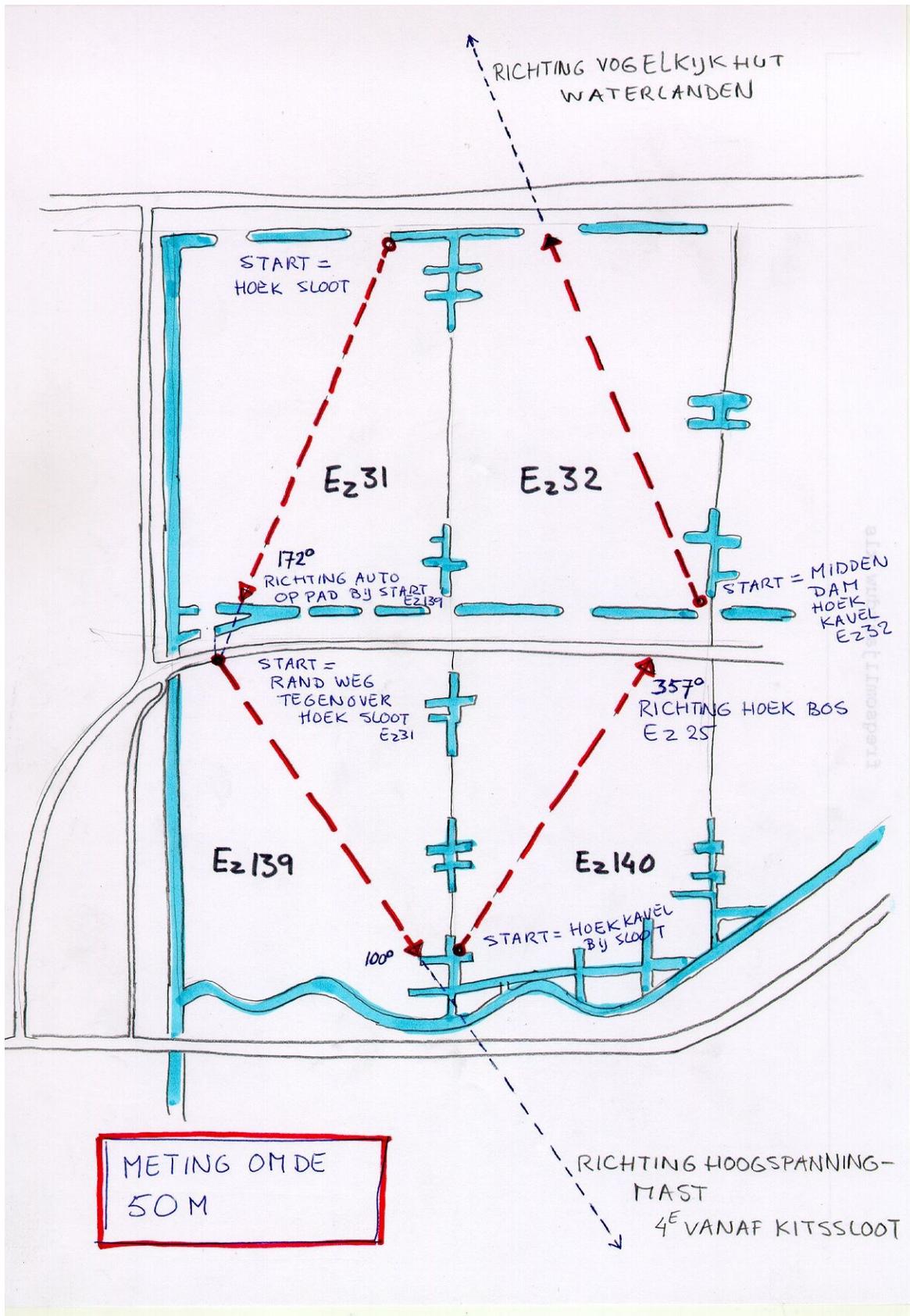


Figure A.3: Transects E139, E140, E32, E31

B Spectral bands

Band name	Wavelength in μm	Surface reflectance type
B1	0.45 - 0.52	Blue
B2	0.52 - 0.60	Green
B3	0.63 - 0.69	Red
B4	0.77 - 0.90	Near Infrared (NIR)
B5	1.55 - 1.75	Shortwave Infrared 1 (SWIR1)
B7	2.08 - 2.35	Shortwave Infrared 2 (SWIR2)

Table B.1: Landsat 7 Spectral Bands

Band name	Wavelength in μm	Surface reflectance type
B2	0.452 - 0.512	Blue
B3	0.533 - 0.590	Green
B4	0.636 - 0.673	Red
B5	0.851 - 0.879	Near Infrared (NIR)
B6	1.566 - 1.651	Shortwave Infrared 1 (SWIR1)
B7	2.107 - 2.294	Shortwave Infrared 2 (SWIR2)

Table B.2: Landsat 8 Spectral Bands

C Literature examination

Study	Vegetation Indices used
Psomas et al. (2011)	NDWI, NDVI, TSAVI, RDVI
Peng et al. (2013)	NDVI, EVI, WRDVI
Payero et al. (2004)	RATIO, NDVI, TVI, IPVI, SAVI, DVI, RDVI, MSAVI
Frank & Aese (1994)	NDVI, RATIO
Karakoc & Karabulut (2019)	NDVI, RATIO
Jansen et al. (2018)	NDVI, NDTI
Wang et al. (2019)	LSWI, NDVI, EVI
Prabhakara et al. (2015)	NDVI, RATIO, EVI, TVI, SAVI, VARI, NDREI
Lussem et al. (2019)	NDVI, RDVI
Hoving et al. (2018)	NDVI, WDV _{Ir} , WDV _{Ig}
Vescovo et al. (2004)	NDVI
Cho & Skidmore (2009)	NDVI, MSAVI, NDWI, SARVI

Table C.1: Vegetation indices used in previous studies

D Hyper-parameter search spaces

Algorithm	Hyper-parameters optimized	Values searched
Artificial Neural Network	Hidden layer sizes	1, 2, 3, 5
	Activation	tanh, relu, logistic
	Solver	sgd, admin, lbfgs
	Learning rate	constant, adaptive
Gaussian Process	Kernel	RBF(), Matern(), WhiteKernel()
	Number of restarts	10, 20
Random Forest	Number of estimators	50, 75, 100, 250, 500, 750
	Max. number of features	0, 1, 2, 3
Support Vector Machine	Kernel	linear, rbf, sigmoid
	C	1, 5, 10
	Gamma	0.1, 0.01, 0.001, 0.0001
K-Nearest Neighbours	Number of neighbours	1 - 30
	Leaf size	1 - 50

Table D.1: Search space for the machine learning algorithms

Parameter	Searchable Values
activation	Hard coded: RectifierWithDropout
epochs	Hard coded: 10000 (true value found by early stopping)
epsilon	{1e-6, 1e-7, 1e-8, 1e-9}
hidden	<ul style="list-style-type: none"> •Grid search 1: {50}, {200}, {500} •Grid search 2: {50, 50}, {200, 200}, {500, 500} •Grid search 3: {50, 50, 50}, {200, 200, 200}, {500, 500, 500}
hidden_dropout_ratios	<ul style="list-style-type: none"> •Grid search 1: {0.1}, {0.2}, {0.3}, {0.4}, {0.5} •Grid search 2: {0.1, 0.1}, {0.2, 0.2}, {0.3, 0.3}, {0.4, 0.4}, {0.5, 0.5} •Grid search 3: {0.1, 0.1, 0.1}, {0.2, 0.2, 0.2}, {0.3, 0.3, 0.3}, {0.4, 0.4, 0.4}, {0.5, 0.5, 0.5}
input_dropout_ratio	{0.0, 0.05, 0.1, 0.15, 0.2}
rho	{0.9, 0.95, 0.99}

Table D.2: Search space for the H2O AutoML Neural Network Architecture

E Scenario I results

E.1 MAE scores

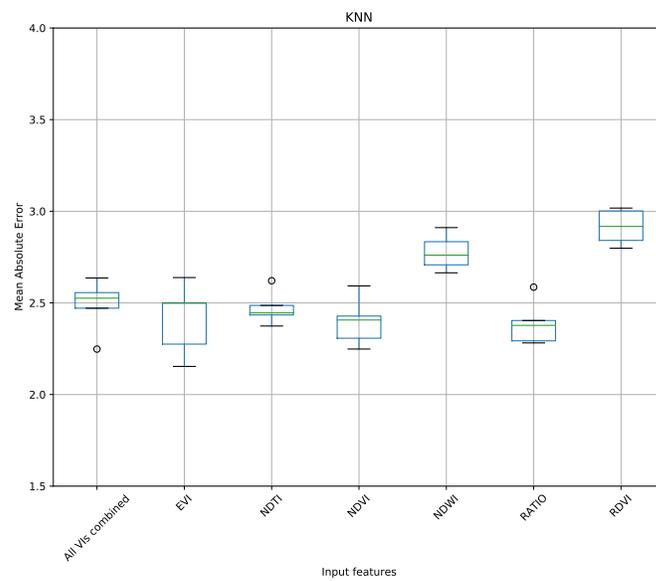


Figure E.1: KNN MAE performance scores

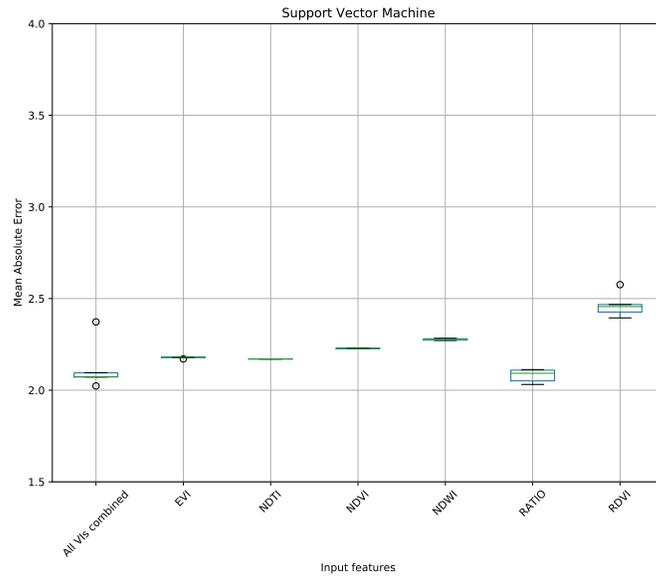


Figure E.2: SVM MAE performance scores

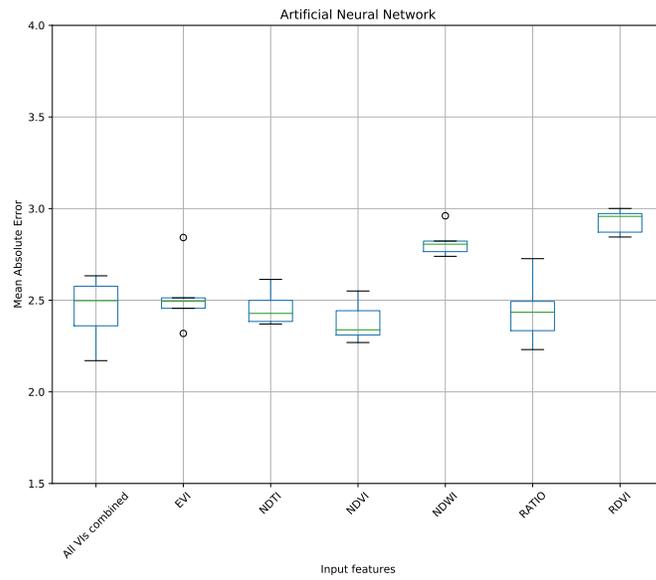


Figure E.3: ANN MAE performance scores

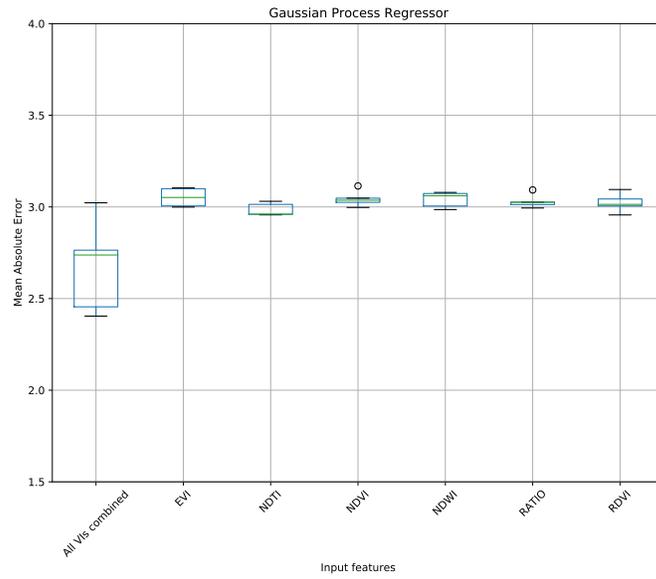


Figure E.4: GPR MAE performance scores

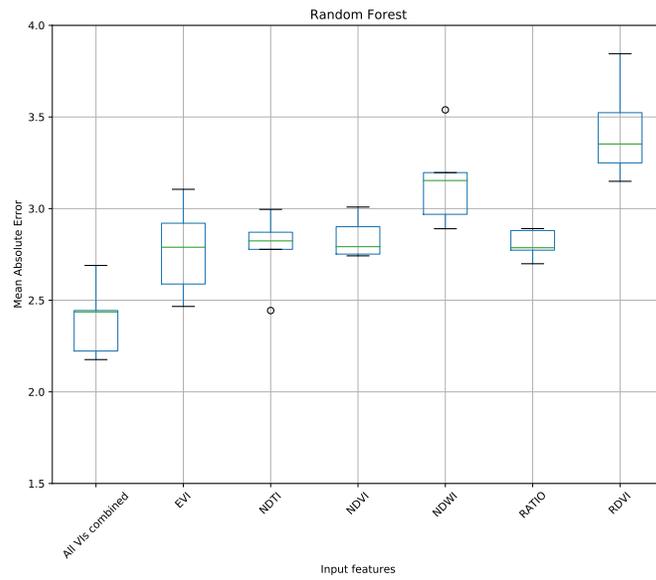


Figure E.5: RF MAE performance scores

E.2 MAPE scores

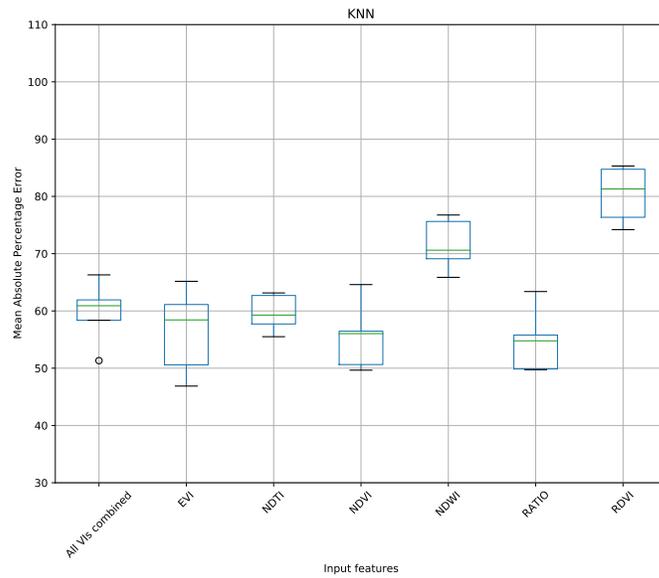


Figure E.6: KNN MAPE performance scores

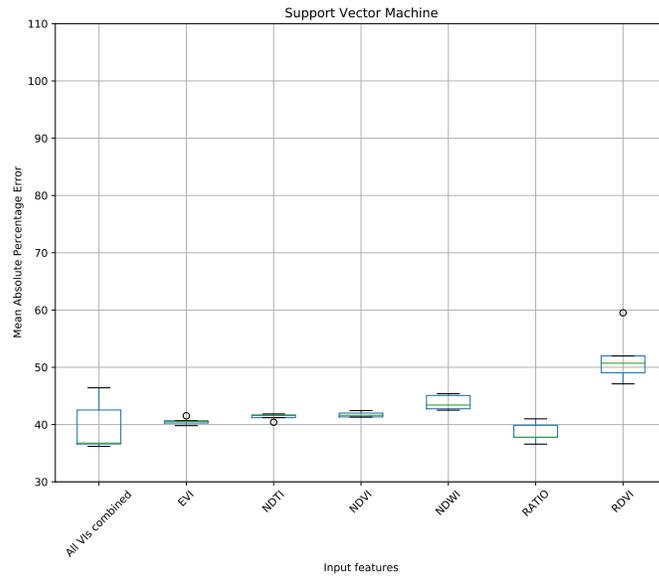


Figure E.7: SVM MAPE performance scores

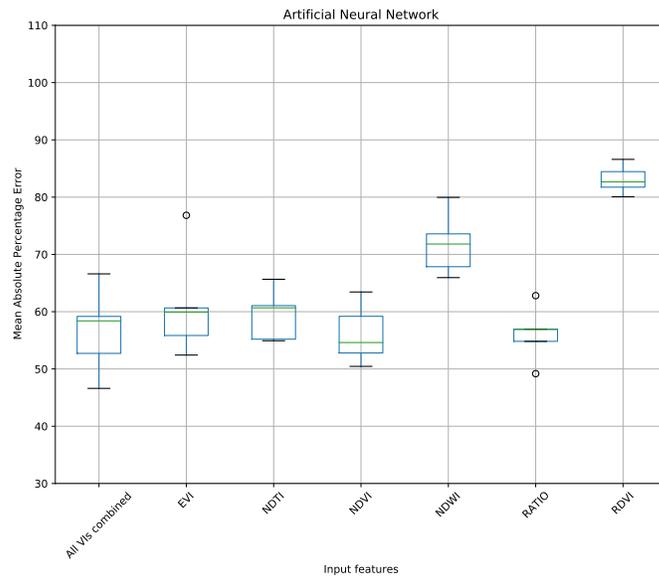


Figure E.8: ANN MAPE performance scores

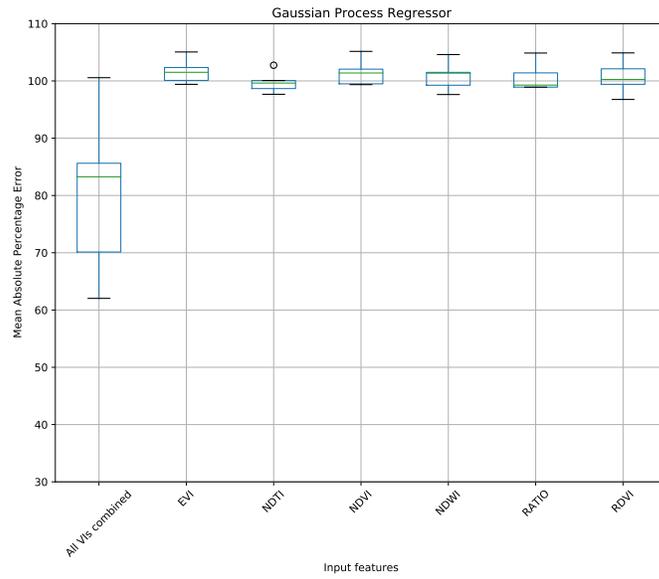


Figure E.9: GPR MAPE performance scores

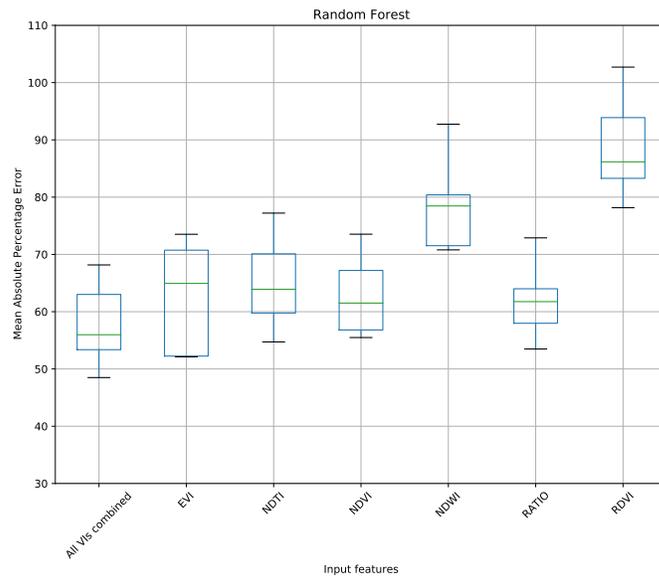


Figure E.10: RF MAPE performance scores

E.3 RMSE scores

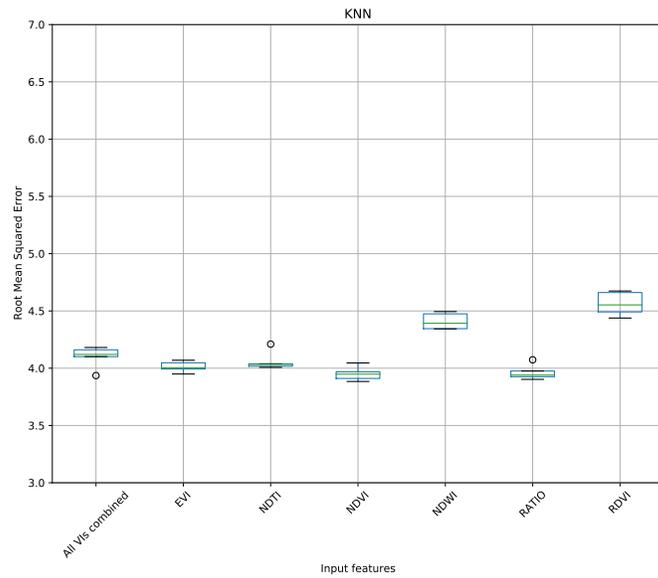


Figure E.11: KNN RMSE performance scores

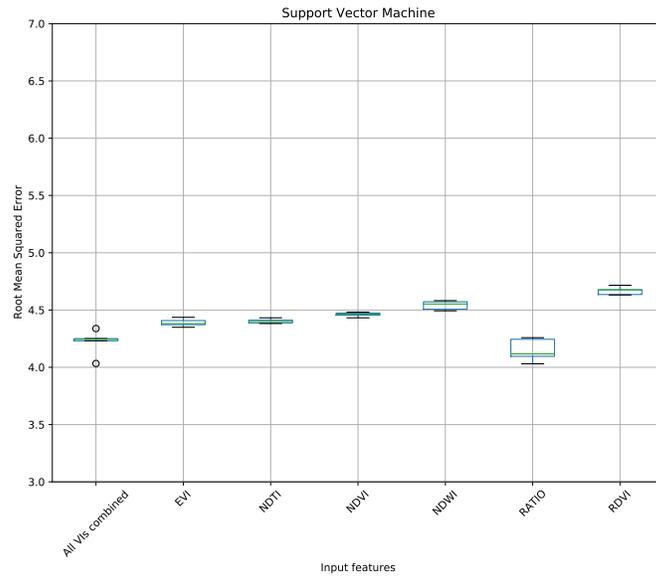


Figure E.12: SVM RMSE performance scores

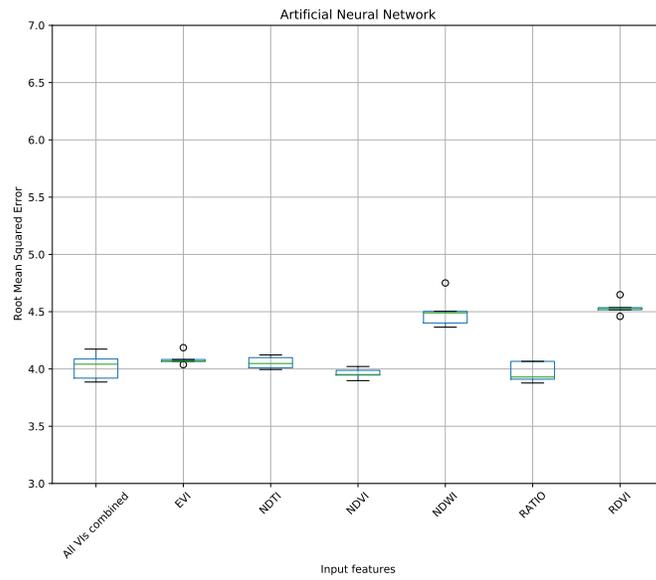


Figure E.13: ANN RMSE performance scores

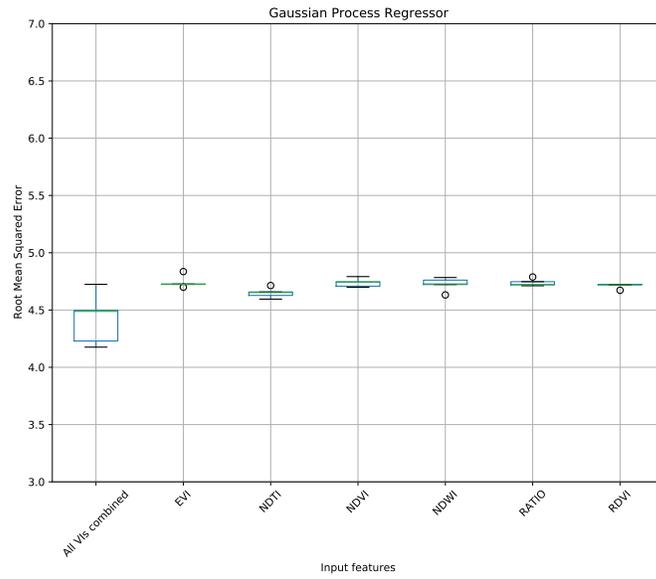


Figure E.14: GPR RMSE performance scores

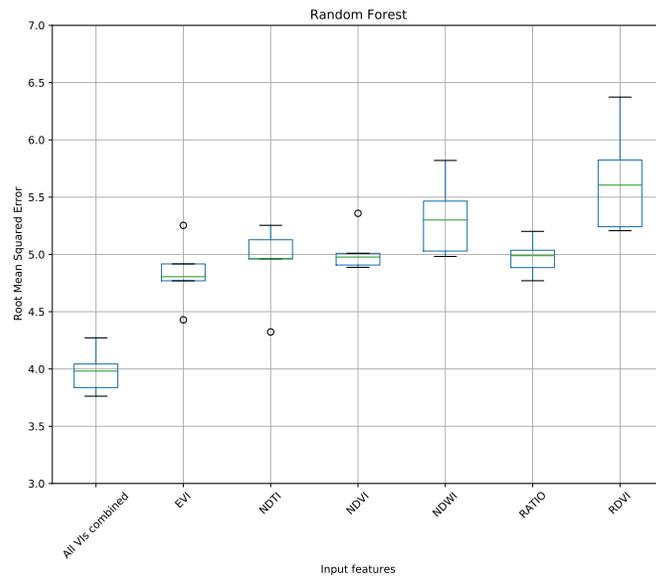


Figure E.15: RF RMSE performance scores

E.4 All performance score values

ML Algorithm	Input features	RMSE	MAE	MAPE
SVM	All VIs combined	4.219499	2.127358	39.704539
	EVI	4.389440	2.178482	40.558912
	NDTI	4.402553	2.170536	41.345459
	NDVI	4.461306	2.227793	41.743709
	NDWI	4.541283	2.276787	43.826464
	RATIO	4.149188	2.079254	38.599707
	RDVI	4.667263	2.464171	51.671345
RF	All VIs combined	3.979208	2.393749	57.804206
	EVI	4.834978	2.773854	62.711096
	NDTI	4.925699	2.782304	65.142014
	NDVI	5.027952	2.839207	62.904760
	NDWI	5.319518	3.149385	78.781659
	RATIO	4.976450	2.805902	62.020654
	RDVI	5.650295	3.423770	88.838945
KNN	All VIs combined	4.099739	2.487383	59.768151
	EVI	4.012767	2.413133	56.438365
	NDTI	4.061409	2.472119	59.669670
	NDVI	3.951641	2.396571	55.481201
	NDWI	4.410081	2.775395	71.590342
	RATIO	3.963621	2.388331	54.713806
	RDVI	4.563450	2.915198	80.379104
GPR	All VIs combined	4.423031	2.676705	80.322708
	EVI	4.742240	3.051942	101.680043
	NDTI	4.649250	2.984313	99.754205
	NDVI	4.737944	3.044196	101.489331
	NDWI	4.725422	3.040731	100.865977
	RATIO	4.737992	3.030284	100.676855
	RDVI	4.712671	3.022835	100.686846
ANN	All VIs combined	4.021516	2.447529	56.699423
	EVI	4.087979	2.524812	61.135107
	NDTI	4.055013	2.459123	59.502602
	NDVI	3.961021	2.382207	56.098928
	NDWI	4.501931	2.818823	71.831857
	RATIO	5.032713	2.443981	56.129954
	RDVI	4.537489	2.929461	83.104306

Table E.1: Mean performance scores for all tested models

F Scenario II results

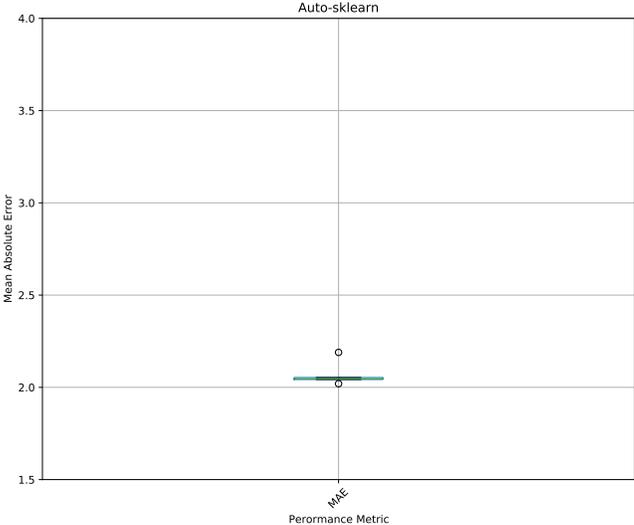


Figure F.1: Auto-sklearn MAE performance scores

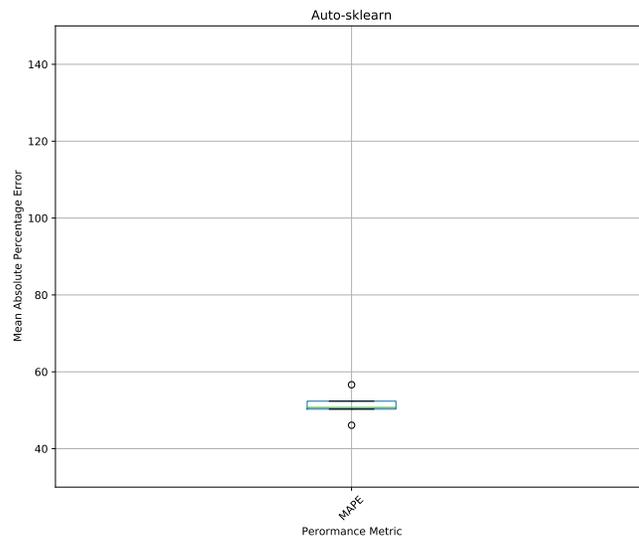


Figure F.2: Auto-sklearn MAPE performance scores

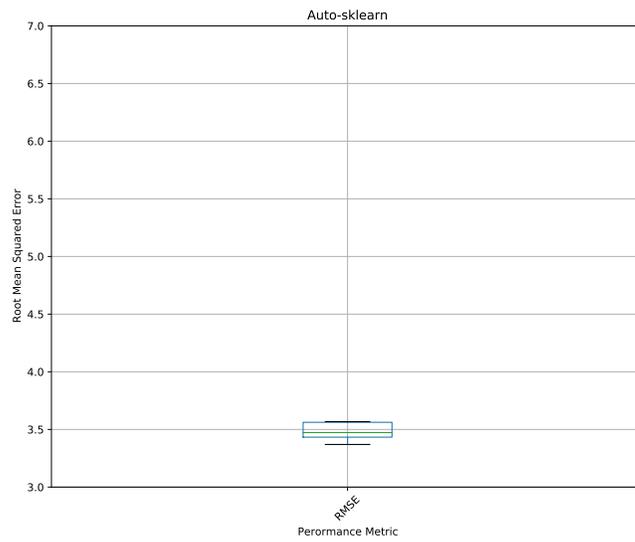


Figure F.3: Auto-sklearn RMSE performance scores

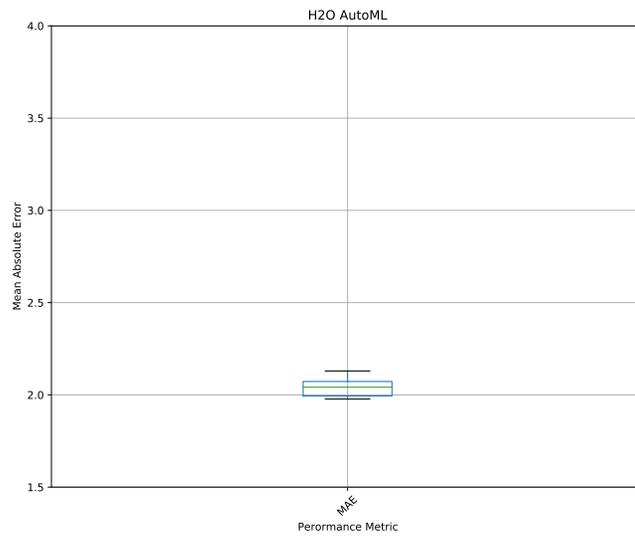


Figure F.4: H2O AutoML MAE performance scores

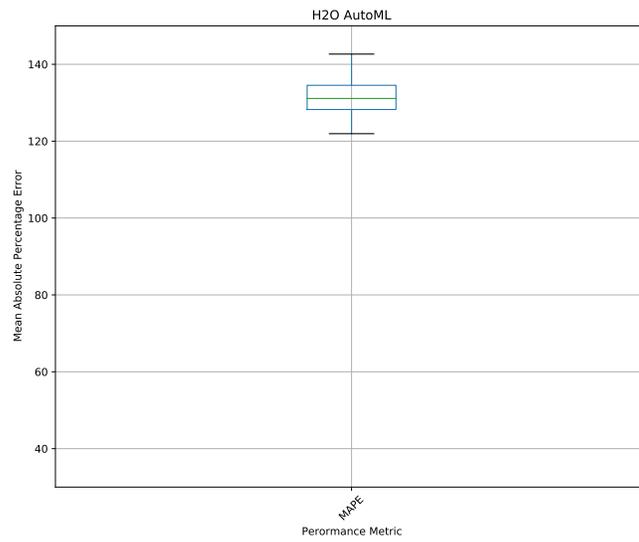


Figure F.5: H2O AutoML MAPE performance scores

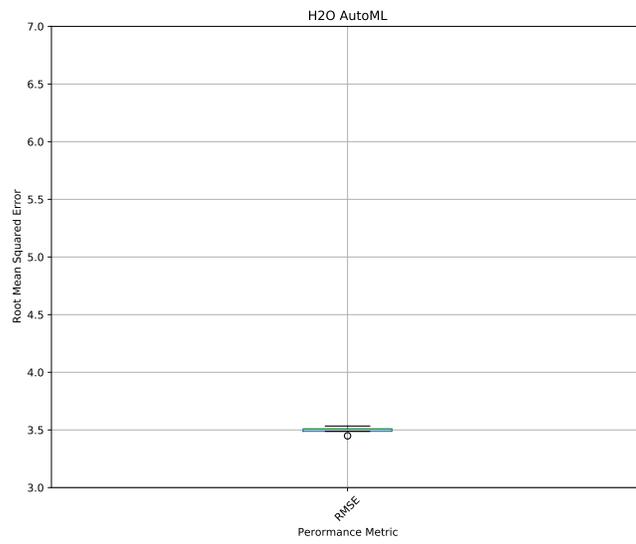


Figure F.6: H2O AutoML RMSE performance scores

G Scenario III results

layer	1	2	3	4	5
units	6	10	10	10	1
type	Input	Rectifier	Rectifier	Rectifier	Linear
dropout	0	0	0	0	
l1		0	0	0	0
l2		0	0	0	0
mean_rate		0.00048358	0.000828805	0.000763349	0.000534967
rate_rms		0.000238964	0.000609651	0.00146712	0.000495285
momentum		0	0	0	0
mean_weight		-0.0439295	-0.01273	-0.0146142	-0.113699
weight_rms		0.366273	0.309482	0.337802	0.564537
mean_bias		0.481552	0.990919	0.992945	0.0336012
bias_rms		0.0482909	0.0345512	0.030939	1.09713e-154

Table G.1: Neural Network Architecture produced by H2O AutoML

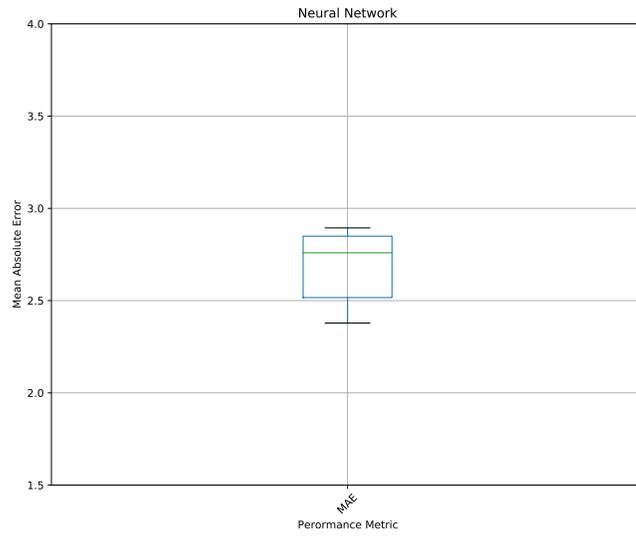


Figure G.1: Spectral band neural network MAE performance scores

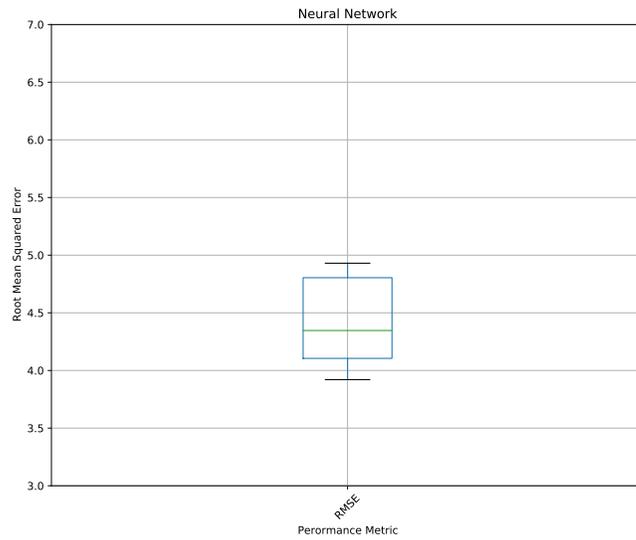


Figure G.2: Spectral band neural network RMSE performance scores