University of Tartu
Faculty of Science and Technology
Institute of Ecology and Earth Sciences
Department of Geography

Master thesis in Geoinformatics for Urbanised Society(30ECTS)

Compare the performance of applying Machine Learning concepts to landcover classification models using very high-resolution UAV data

MarjanSadat Barekaty

Supervisor: Edgar Sepp

Approved for defence:
Supervisor:
Head of Department:

Tartu 2021

Masinõppe meetodite rakendamisel saadud taimkatte klassifikatsioonimudelite tulemuslikkuse hindamine kasutades kõrge ruumilise lahutusega UAV andmeid

Abstrakt

Maakatte klassifitseerimisel kasutati kolme laialdase kasutusega masinõppe algoritmi: *Random Forest* (RF), *Support Vector Machine* (SVM) ja *K-Nearest Neighbours* (KNN). Lisaks nimetatud algoritmidele rakendati erinevaid masinõppes kasutatavaid meetodeid, lisatunnuste loomist ja nende erinevaid kombinatsioone. Masinõppe meetoditena kasutati andmete skaleerimist (*MinMaxScaler*), alaesindatud klasside võimendamist neid korduvalt juhuslikult valides (*random oversampling*) ja mudeli parameetrite häälestamist (*hyperparameter tuning*). UAV ortofotomosaiigi RGB väärtuste põhjal loodi täiendavad tunnused (vegetatsiooniindeksid): roheliste lehtede indeks (*Green Leaf Index* (GLI)) ja nähtav atmosfääritakistuse indeks (*Visible Atmospherically Resistance Index* (VARI)).

Kõrgeim kaalutud keskmine F1-skoor saadi RFi vaikemudeliga kombineerituna vegetatisooniindeksitega (0,59), sellele järgnesid KNN (0,58) ja SVM (0,57) kombineerituna vegetatsiooniideksite ja *MinMaxScaleriga*. Klassifitseerimist raskendas oluliselt UAV ortofoto kõrgest ruumilisest lahutusest tingitud müra ja maaktteklasside mitte tsakaalus olev koosseis.
Teistele uuringutele tuginedes saaks klassifitseerimistulemusi parandada kasutades objektipõhist pildianalüüsi (OBIA), mis annaks lihtsa ruumilise lahutuse vähendamisega võrreldes paremaid tulemusi ja kalibreeritud multispektraalsete piltide kasutamisega.

Võtmesõnad: Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbours (KNN), masinõppe algoritmid (MLA)
CERCS-i kood: S230 sotsiaalgeograafia

Compare the performance of applying Machine Learning concepts to landcover classification models using very high-resolution UAV data

Abstract

The performance of landcover classification models based on three widely used machine learning algorithms (MLA) such as Random Forest (RF), Support Vector Machine (SVM) and K-Nearest Neighbours (KNN) was compared in this study. Different models were created using ML concepts such as scaling (MinMaxScaler), oversampling (random oversampling) and hyperparameter tuning. The additional constructed features made from RGB values of UAV orthophoto are Green Leaf Index (GLI) and Visible Atmospherically Resistance Index (VARI). The highest average weighted f1-score was obtained by RF default model combined with vegetation indices (0.59) and followed consecutively by KNN (0.58) and SVM (0.57) combined with vegetation indices and MinMaxScaler.One of the primary data for this study (UAV orthophoto) tends to be very noisy. Also, landcover samples were imbalanced; as a result, the classification was more problematic.
Object-Based Image Analysis (OBIA) approach can make better accuracy results compared to lowering the spatial resolution. The reviews of previous studies with calibrated multispectral images confirmed the high accuracy results by machine learning classifiers.

Keywords: Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbours (KNN).Machine Learning Algorithms (MLA)
CERCS Code: S230 Social geography

Table of Contents

1 Introduction 1

2 Theory 2

2.1 UAV application in landcover monitoring 2

2.2 Machine learning (ML) 4

2.3 Machine learning concepts (ML concepts) 4

2.4 Accuracy assessment 6

2.5 Machine learning classification algorithms 8

2.5.1 Random forest classification (RF) 8

2.5.2 Support vector machine classification (SVM) 9

2.5.3 K-Nearest-Neighbour Classification (KNN) 9

3 Data and Methodology 11

3.1 Study area 11

3.2 Data 12

3.3 Additional constructed features 18

3.4 Data preparation and exploration 21

4 Results 29

4.1 Classified UAV Orthophoto 31

5 Discussion and Conclusion 33

5.1 Applying machine learning concepts 33

5.1.1 Additional constructed features 33

5.1.2 Data scaling 33

5.1.3 Oversampling 34

5.1.4 Hyperparameter tuning 34

5.1.5 Combination of ML concepts 34

5.2 Comparison of RF, KNN and SVM Classifier 35

5.2.1 Spatial resolution 36

5.2.2 Random sampling vs spatial sampling 38

5.3 Classification accuracy 40

6 Conclusion 44

Acknowledgements 45

Kokkuvõte 46

References 47

LIST OF ABBREVIATIONS

CV	**Cross-Validation**
DSM	**Digital Surface Model**
FP	**False Positive**
FN	**False Negative**
GLI	**Green Leaf Index**
KNN	**K-Nearest Neighbours**
MLA	**Machine Learning Algorithms**
ML	**Machine Learning**
MMS	**MinMaxScaler**
OBIA	**Object-Based Image Analysis**
OS	**OverSampling**
RF	**Random Forest**
SVM	**Support Vector Machines**
TN	**True Negative**
TP	**True Positive**
UAV	**Unmanned Arial Vehicle**
VARI	**Visible atmospherically resistant Index**
VHR	**Very High Resolution**

1 Introduction

Changes in the ecosystem have always drawn the researcher's attention to track and measure their natural habitat. This study is conducted to classify different land cover types in a specific area that has faced changes due to peat mining.
One option to measure and understand the changes during the restoration project is to monitor landcover types evolution. Using monitoring techniques such as Unmanned Aerial Vehicles (UAVs) will allow us to investigate the changes. In addition to UAV images, it is necessary to employ scientific methodology, such as machine learning classifiers, to classify land cover types. This thesis is composed of scientific conclusions for assisting researchers in monitoring landcover changes is also part of this study.
UAV surveys were decided as the best and cost-effective monitoring tool. The acquisition and development of Unmanned Aerial Vehicles (UAVs) data, commonly known as drones, have become a popular dataset. The UAVs are beneficial and promising standard technology. UAVs have proved their efficiency in gathering data in unreachable and restricted areas due to their small size and rapid deployment. Different authors have used UAV in mine monitoring, precision forestry, agriculture, automatic mapping of land surface elevation changes and disaster damage assessment.
Two sets of primary data, UAV mosaic very high resolution with a Digital Surface Model (DSM) of the study area gathered landcover samples (point shapefiles) with overall 4148 samples had been used in this study.
Random Forest (RF), Support Vector Machine (SVM) and K-nearest neighbours (KNN) are three machine learning classification algorithms used in this study. After training the model, each classifier model's performance based on different ML concepts and constructed additional features will be evaluated. The evaluation of each classifier performance is done by Average weighted f1-score.

This study is going to answer three research objectives:

1. Compare the performance of the three machine classifiers (RF, SVM, KNN).
2. Analyze the effect of added constructed features and a combination of machine learning concepts on each model's performance.
3. Analyze the usage of Very High Resolution (VHR) UAV data on machine learning classifiers' performance.

The thesis critically discusses comparing the classification performance based on different machine learning algorithms and applied constructed features. The
results can be a guide for image classification researches to learn from the classifiers' behaviour.

2 Theory

2.1 UAV application in landcover monitoring

A considerable amount of literature has been published on image classification with VHR UAV with different machine learning algorithms such as (Hall et al., 2018), (Böhler et al., 2018), (Pajares, 2015). They employed the high spatial resolution images provided by the UAV imagery system in their research. Also, (Poblete-Echeverría et al., 2017), (Melville et al., 2019) found that ultra-high-resolution UAV orthophoto is suitable for the classification, identification, detection improvement and discrimination between classes. Besides the benefits of using UAVs, there are some established concerns posed by various researchers like (Im et al., 2008), (Miao Li et al., 2014), (Hall et al., 2018) in which they claimed that pixel-based approaches are more affected by noise issues existed in UAV images. Hall et al. (2018) added that ultra-high-resolution UAVorthophoto has complicated and longer classification process.

UAV imagery has shown a powerful usage in several research fields. A study by (Feng et al., 2015) indicated a great potential of high-resolution UAV imagery data on the urban landscape vegetation mapping. In another study by (Jónsson, 2019) a distinction was made between the RGB images classification of the agricultural crop site in Sweden and multispectral UAV orthophoto with a machine learning algorithm. Effect of additional constructed features like vegetation indices and considering the spatial resolution, segmentation on classification accuracy has been evaluated as well. Overall, it confirms the considerable effect of pixel size on the accuracy of the classification. It changed the accuracy from 58% (1cm) to 88% (5cm), and by adding the additional constructed features, the overall accuracy increased by 10%.

Hung et al. (2014) have classified weeds on UAV high-resolution images. They conducted an approach with less processing time, including calibration of the camera and making image mosaic. Hung et al. (2014) have claimed that UAVs have the complementary capacity to traditional approaches (field works) and also in remote sensing services (unmanned aircraft, satellites). Hung et al. (2014) reported two benefits of using UAVs. Firstly, they are cheaper compared to satellite imagery approaches. Secondly, UAVs cover larger areas in a shorter time than field works process.

Furthermore, they mentioned that UAVs enable users to have higher spatial resolution since they can work in lower altitudes (Hung et al., 2014). Different researchers highly recommend finding adequate classification algorithms for UAV imagery data. Moeckel et al. (2018) and other researchers like (Song et al., 2019), (Ma et al., 2017), (Yuan et al., 2016), (Xie et al., 2018) have used machine learning algorithms such as Random Forest and Support vector machine for crop classification with remote sensing data. Guo et al., (2013) and two other researchers (Hamuda et al., 2016), (Torres-Sánchez et al., 2015) have classified vegetation in agricultural lands by employing VHR UAV orthophoto alongside with multispectral imagery data.

There is a large volume of published studies describing the machine learning algorithms in different vegetation classification by UAV imagery data such as (Garcia-Ruiz et al., 2015), (Guerrero et al., 2012), (M. Pérez-Ortiz et al., 2015), (María Pérez-Ortiz et al., 2016). Thanh Noi & Kappas, (2018) have compared the accuracy of three different machine learning algorithms such as Random forest (RF), Support Vector Machines (SVM) and K-nearest neighbours (KNN) in landcover classification. They have used satellite imagery data in their research and achieved high accuracy by both balanced and imbalanced datasets.

The highest accuracy was obtained by SVM due to the least sensitivity to training data and two other classifiers RF and KNN gained high accuracy, respectively. Thanh Noi & Kappas (2018) have claimed that the training data's size directly impacts the overall accuracy. Meaning having more training data will increase the overall accuracy. In research with multi-class data, it has reported that SVM classifier provided highly accurate result compared to the neural network and maximum likelihood classification algorithms. In other words, SVM worked perfectly with high dimensional data (Pal & Mather, 2005).

Rodriguez-Galiano et al. (2012) have mentioned that machine learning algorithms have outperformed compared to the alternative conventional parametric algorithms. Du et al., (2015) and (Rodriguez-Galiano et al., 2012) have chosen RF among the other multiple machine learning algorithms because RF classifier operates faster and has a generalization performance. Alternative studies (Lottes et al., 2017), (Ma et al., 2015) have reported that RF classifier is highly appropriate for UAV data classification, mostly in agricultural mapping, RF has performed well (Manchun Li et al., 2016).

Surveys have been conducted to find the best classification algorithm for land use/cover studies by comparing these classifiers' (RF, KNN, SVM) performance among themselves or other classification algorithms (Pan et al., 2017). However, their conclusions are quite different from our study results. For example, studies by (Adam et al., 2014) and (Ghosh & Joshi, 2014) have proved that SVM and RF have shown similar classification results. Khatami et al., (2016) found out that SVM, KNN and RF are operating significantly better than traditional supervised classifiers.

A study by (Pan et al., 2017) has considered a different application of machine learning classifiers (SVM, KNN and RF) with UAV data to distinguish between normal pavement and damaged pavement. This study pointed out the key parameter of each classifier. For example, in KNN the number of k has an important role, which determines the number of neighbours for classification. In RF, the most critical factor is the number of trees, and in SVM, the type of kernel significantly affects each classifier model's accuracy.

Some solutions, such as oversampling/ Undersampling, have been proposed (Ganganwar, 2012) due to an imbalanced dataset's negative effect on the accuracy assessment. An oversampling method is conducted in this study to overcome the imbalance dataset issue.

To enhance the classification result, using vegetation indices for additional constructed features drawn our attention to articles like (Jelínek et al., 2020) which used vegetation indices (VI) to monitor crops. It confirms that each vegetation indices belongs to a different spectrum, meaning it has different values. Some researches were focusing on RGB based vegetation indices such as (Lussem et al., 2018), (Jannoura et al., 2015), (Hunt et al., 2005), (Bendig et al., 2015). Most of the mentioned studies have researched to find correlations of different varieties of vegetation indices with RGB imagery bands.

2.2 Machine learning (ML)

ML is defined as a combination of various approaches like probability theory, statistics, decision theory and optimizations (Singh et al., 2015). The critical point of machine learning structure lies in the availability of big data for implying computerized modelling approaches to train the data based on the existing pattern. Data classification is done automatically based on the learned underlying structure. It was pointed out that Machine Learning is divided into supervised learning, unsupervised, and Reinforcement (Dwivedi, 2019).
In supervised learning, the model trains the labelled data (Dronova et al., 2012). Afterwards, it evaluates the performance of the model based on the validation data. The data used in this process can be both binary and multi labelled.

2.3 Machine learning concepts (ML concepts)

Features

Features are inputs of the ML process (Vickery, 2020). The key attributes of classification are each feature's labels. Selecting the best features combinations has critical role in model optimization (Vickery, 2020). It is essential to consider the most optimal training process features to have the best prediction result. The machine learning process initiates with training the model phase and evaluates the model by validation data.

Training phase

This phase produces learned output data based on the training dataset (Mitchell, 1997). The training phase could be named as fitting a model as well, in which the ML algorithm use labelled data and its value to figure out the pattern (Vickery, 2020). This training process is necessary for ML to evaluate the model's performance by using test data that are not used in the training phase.

Overfitting

A recent study by (Ying, 2019) has provided the underlying reasons for overfitting like having a small training dataset, and a few samples with noise. The model learned the noise in the training and have negative impacts on the model performance. The overfitting has defined by having a good performance on training and poor performance on the test dataset.

Hyperparameter Tuning

Hyperparameter tuning is a systematic approach of searching the best sets of model parameter values because the default parameter's value cannot always achieve the best results. Searching for the best results are determined by cross-validation. Numerous studies have conducted this approach (Friedrichs, 2004), (Bergstra & Bengio, 2012). Mantovani et al. (2015) indicated that it is better to use hyperparameter values than default values to have high accuracy. The importance of setting up the hyperparameter value before initiating of the model training is pointed out by (Preuveneers et al., 2020). The optimization process includes the optimization of kernels, number of iteration, Etc., which is specific to each classifier (Claesen et al., 2014). The benefits of using grid search, which is having a fast and straight forward construction instead of a random search for the optimal hyperparameters is claimed by (Bergstra & Bengio, 2012).

GridSearchCV

One of the hyperparameter tuning methods is GridSearch. GridSearchCV approach documentation library is (Sklearn.Model_selection.GridSearchCV — Scikit-Learn 0.23.2 Documentation, n.d). The main idea is that all model parameter values are predefined and GridSearch will form all the combinations, train the model using CV and reach the final accuracy. GridSearch approach used by different researchers like (Ramadhan et al., 2017), (Wenwen et al., 2014), (Ataei & Osanloo, 2004) for tuning and optimization of their ML model. The combination of grid search and cross-validation is necessary for obtaining the best tuning values (Hsu et al., 2003).

Cross-Validation

CV is a statistical approach where the train set is divided into k-folds (K-folds CV). One of the folds is selected as a validation set, and other folds are used for model training. Model results are checked against a validation set, and accuracy measures are calculated. It is repeated for all validation sets. In the end, average accuracy might be calculated. When is used with hyperparameter tuning the CV is done for all the combinations of parameters, and higher accuracy measures determine the best parameter set. Some study references (Refaeilzadeh et al., 2009), (Raschka, 2018) described cross-validation as a statistical approach in which data is divided into k parts. One division is for validation, and the rest k-1 part is employed for training the model for final evaluation.

Data preprocessing

In broad, data pre-processing is at the core of the machine learning algorithm. Its a process for cleaning raw data collected from various sources. Data pre-processing is necessary due to the drawbacks of raw data such as missing values, duplication, outliers and noised data (Pant, 2019). In detail, there are some approaches in machine learning that can help for pre-processing the data. Standardization and normalization of data are different types of data processing (Dorpe, 2018). Some distance-based algorithms such as KNN and SVM use the distance between points to distinguish similarity for the classification process. Using normalization will scale a distance-based algorithm and make features participate equally in the results ("Feature Scaling | Standardization Vs Normalization," 2020).
MinMaxScaler (1) is a standard normalization method which is available in "*sklearn. Processing"* library. In this method, the instances are divided by the difference between the original maximum and original minimum. It keeps the original distribution of data. This method's range of the features normalized is between 0 to 1 (Hale, 2020). MinMaxScaler is one of the recognized normalization methods (Raju et al., 2020).

$$X_{norm} = \frac{X - X_{min}}{X - X_{min}} \quad (1)$$

X is an original value of Xnorm value.

Imbalanced data set

Most of the real-world datasets are imbalanced, and it means there is no equal distribution of instances for each type of data. Having imbalanced data set can affect model training in machine learning algorithms considerably (Japkowicz, 2000), (Weiss, 2004), (Raeder et al.,2012). In the classification process, the model is skewed to the majority classes and ignores the minority classes. As a result, the classifier's performance is significantly affected by an imbalanced dataset (N. V. Chawla et al., 2004). Researchers have introduced different ways to overcome this problem. Oversampling and Undersampling are the most common strategies for balanced datasets (Drummond & Holte, 2003), (Weiss, 2004).

Oversampling and Random oversampling

Random oversampling is used to overcome the imbalanced dataset issue. Oversampling is a method to increase the number of the minority class by replicating their instances for balancing the dataset (Zheng et al., 2016). Random oversampling has explained the same pattern by randomly repeating sample instances, classes with minority instances replicated in training data (Batista et al., 2004). In (Ling & Li, n.d. 2010) it has claimed that, although oversampling is a robust approach, it replicates samples of minority data and adds no new information. Hence, the model tends to face overfitting. Besides, by increasing the number of training values, the learning process will take more time (Dataman, 2020), (Ganganwar, 2012).
Random oversampling is competitive to other oversampling methods such as synthetic Minority Over-sampling Technique (SMOTE) (Batista et al., 2004).

2.4 Accuracy assessment

For having promising results with optimal values, it is necessary to check the accuracy by different approaches. There are some devised features in this part to assess the accuracy of the achieved parameters. One of the practical methods for having information about the algorithm's performance is Classification report. The classification report is a practical solution based on different defined accuracy assessment features such as precision, recall, F1-score and average values. It also helps to detect the weakness of the performance faster. The report's metric is defined by True and False positives and True and False negatives. When the actual class is positive and correctly predicted as positive, it is counted as True positive. A false positive is when the actual is negative, but it has been predicted as a positive class.

Precision: Precision defines the percentages of the correctly predicted positive classes. The classifier cannot estimate the negatively labelled classes as positive (2).

Recall: The ratio of the positive labelled classes that have been estimated correctly to all instances (3).

F1-score: It is the average of precision and recall. The highest score is 1.0, and the lowest is 0.0 (4).

Support: It shows the actual number of each class occurrence in the dataset.

$$Precision = \frac{TP}{(TP + FP)}$$

(2)

$$Recall = \frac{TP}{(TP + FN)}$$

(3)

$$F1 - score = \frac{2 * (Recall * Precision)}{(Recall + Precision)}$$

(4)

Micro average: The classes with more number of instances dominate the average results**.** (Pereira & Nunes, 2017).

Macro average: Each class in this average have been considered similarly. The problem is that the result for classes with a low number of instances is not reliable. (Pereira & Nunes, 2017).

Weighted average: Calculates the average by assigning weights based on the number of instances per class. The sensitivity to classes with a lower number of classes will decrease. The weighted average is more sensitive to imbalanced data sets than other averaging methods (Pereira & Nunes, 2017).

Confusion Matrix

The confusion matrix is a (NxN) table (N is the number of the classes) with predicted and original values to analyze the performance of the model (Narkhede, 2019). This table includes the summary of correct and incorrect predictions made by the model, and it demonstrates where classification models are confused for making predictions (Brownlee, 2016). The main metrics in the confusion matrix are described below:

True positive (TP): It is assigned to the number of classes that have been correctly predicted as positive.

True negative (TN): It is assigned to the number of classes that have been correctly predicted as negative.

False-positive (FP): It is assigned to the number of negative classes that have been incorrectly predicted as positive

False-negative (FN): It is assigned to the number of positive classes that have been incorrectly predicted as negative

The most practical accuracy assessments described above (accuracy score, recall, F1-score and precision) can be calculated from the confusion matrix. For multiclass labelled data, TP, TN, FP, FN metrics are considered for each class separately (Mohajon, 2020).

2.5 Machine learning classification algorithms

2.5.1 Random forest classification (RF)

Random forest ensemble classifier's advent goes back to 1995 in research by (Ho, 1995). RF classification follows a series of steps to assign a pixel/segment to a class. As (Rodriguez-Galiano et al., 2012) proposed, these steps are based on multiple classification decision trees. The critical aspect of the random forest classifier is the creation of decision trees. The classification process starts with the selection of data from the training dataset randomly to create each tree. About 2/ 3 of the data is a subset of each tree and the other 1/ 3 of the training data called "Out-Of-Bag" (OOB), which tests the accuracy. Also, (Watts et al., 2011) introduced homogeneity of the classes as another parameter that has a leading role in splitting the trees.
Afterwards, the classification continues its process by making the second random sampling for each node in the tree (Horning, 2010). The model prediction is based on the class with the majority of votes. It has mentioned that two features N (Number of trees) and m (Number of variables used to split each node) has a significant role in this classifier and needs to be defined before the initiation of the process (Breiman & Cutler, 2005).
It is indicated that Random forest classifier could obtain high accuracy in data with high dimensions and multi classes (Breiman, 2001). Also, (Bosch et al., 2007) mentioned that random forest classifier usage in image classification has increased in recent years. Ozesmi & Bauer (2002) used RF classifier in their study with remote sensing data since RF can handle the high dimensional datasets. Millard & Richardson (2015) outlined that training size and its distribution can affect this classifier significantly. The high model accuracy will be obtained by big training size and random distribution. Having balanced or imbalanced dataset affects the accuracy of the RF model as well. Millard & Richardson (2015) have mentioned that the RF model with imbalanced training dataset will show discriminatory behaviour toward the class with the highest frequency. RF creates trees based on user-defined values, and it overcomes the overfitting issue (Breiman & Cutler, 2005).
Liaw & Wiener (2002) claimed that RF classifier compared to other ML classifiers such as Support Vector Machines and Neural Networks behaves more flexible.

The Random forest classifier ensemble library in machine learning algorithm is "*Sklearn. Ensemble.RandomForestClassifier"*. The most important features which have been conducted in this study are described below (Ippolito, 2019):

N_estimator: Number of trees in the ensemble.
Max_depth : Maximum number of levels allowed in each tree.
Max_features : Maximum number of features considered splitting a node.
Min_samples_leaf : Minimum number of samples which can be stored in tree leaf.
Min_sample_split : Minimum number of samples necessary in a node to cause node splitting.
Bootstrap: Having random sample replacement of observation with repeating.

Studies results by (H. K. Zhang & Roy, 2017), (Liaw & Wiener, 2001) and (Duro et al., 2012) confirmed that using default parameters of RF classifier achieved satisfying results. In documentation related to the RF classifier, the parameters are adjusted with different values to limit the computation time and memory consumption. (*3.2.4.3.1. Sklearn. Ensemble.RandomForestClassifier — Scikit-Learn 0.23.2 Documentation*, n.d.).

2.5.2 Support vector machine classification (SVM)

Support Vector Machine (SVM) classifier is an algorithm that can achieve a high level of accuracy in the classification of high dimensional datasets (Huang et al., 2002). Besides, SVM has surpassed all the best classifiers such as Neural Networks, K-Nearest Neighbours, Maximum Likelihood and Decision Tree classifiers for remote sensing imagery datasets. SVM conducts a new approach by creating hyper-plane in a n-dimensional space instead of taking each class's mean values (Foody & Mathur, 2004). This plane separates the classes based on a specific defined kernel and parameters of the ML algorithm. Heumann (2011) claimed that by maximizing the distance from the closest point to the hyperplanes, SVM classifiers could reach optimization. The main feature that assigns segments to different SVM classes is maximized margin from the hyperplane (Cortes & Vapnik, 1995).
Tzotsos & Argialas (2008) mentioned four different kernels for training data such as Linear, Polynomial, Radial Basis Function (RBF) and sigmoid in SVM classifier. The primary use of these kernels is for finding the optimal hyperplane. In pixel-based remote sense data, RBF kernel has accurate and useful results (Huang et al., 2002), (Mercier & Lennon, 2003). The important role of kernels for remote sense data was confirmed by (Varma et al., 2016).

There are two types of SVM classification algorithms, binary and multi-class (Crammer & Singer, 2001). Multi-class SVM has various applications because most of the real-world use cases include more than one category. Research papers have investigated applications of multi-class SVM, such as "optical character recognition (Mori et al., 1992), intrusion detection" (Khan et al., 2007), "speech recognition" (Spiess et al., 2007) "and bioinformatics" (Baldi & Pollastri, 2002). In a study by (Wang & Xue, 2014) the SVM classifier has shown high accuracy results with an unbalanced dataset. The Support Vector Machine classifier's library in machine learning is "*sklearn. SVM.SVC"*.
The most important features in this library are described below:
Kernel: The default version of the kernel is "RBF".
Studies results (Knorn et al., 2009; Shi & Yang, 2015) have shown the perfect performance of (RBF) kernel in SVM classifier in Landcover classification.
Cost and Gamma: The kernel parameters that are calculated based on a GridSearchCV.
C is a regulator of the model, and larger values provide a model with low performance because the errors are neglected. If C values are high, the model tends to face overfitting. Gamma can find the relation between the feature distance and similarity (*Sklearn.SVM.SVC — Scikit-Learn 0.23.2 Documentation*, n.d.).

2.5.3 K-Nearest-Neighbour Classification (KNN)

Different researchers such as (Cover & Hart, 1967), (Bremner et al., 2005) claimed that the K-nearest neighbour algorithm is a classification method based on choosing the nearest training instances in feature space to each class. This non-parametric classification was used for statistical applications in 1970 (Franco-Lopez et al., 2001).
The classification process initiates finding the number of K neighbours with distance equations (5) like Euclidean, Manhattan and Minkowski. The average of the K-group of trained samples calculates the data label (Akbulut et al., 2017; Wei et al., 2017). Qian et al.(2015) has mentioned that K plays an important role in this classification. K highly affects the parameter tuning. "metric distance" is counted is highly effective for optimization of the model. Euclidean is one of the most common metric distances; it calculates the distance between two points in a plane. (*Sklearn.Neighbours.KNeighborsClassifier — Scikit-Learn 0.23.2 Documentation*, n.d.)

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad \text{Euclidean distance}$$

$$\sum_{i=1}^{k}|x_i - y_i| \quad \text{Manhattan distance}$$

$$\left(\sum_{i=1}^{k}(|x_i - y_i|)\right)^{\frac{1}{q}} \quad \text{Minkowski}$$

(5) Distance Functions

It is better to use cross-validation with an independent dataset to determine K's value for achieving the best measuring distance. The optimum value of the K is between 3-10 (*KNN Classification*, 2020). The main drawback of KNN classification is that all distance-based approaches are affected by the scale of variables. It's vital to have a dataset that is not skewed to the data with high frequency. Normalization can avoid having a biased dataset. The recommended approach for normalizing the values is using the Min-Max scaling Equation (1) (Sharma, 2019).

The K-nearest neighbours classifier's library in machine learning is *Sklearn. Neighbours. KNeighborsClassifier*. Essential features in this library are described below:

N_neighbors: An odd number that defines the number of neighbours.

Weight: It gives weight to data based on the distance feature. It weights the inverse of point distance. Weighting distances is done by "Uniform", which weighs all points in the neighbourhood equally. The alternative way for weighting is using "Distance", it weighs the points based on their inverse distance, meaning the points are near weight more than further ones.

Metrics: Approaches for measuring the distance.

3 Data and Methodology

3.1 Study area

The study area is located near to Miama village, Pärnu County in south-west Estonia. (Latitude:58.5983, Longitude:24.4762) represented in Figure 1.

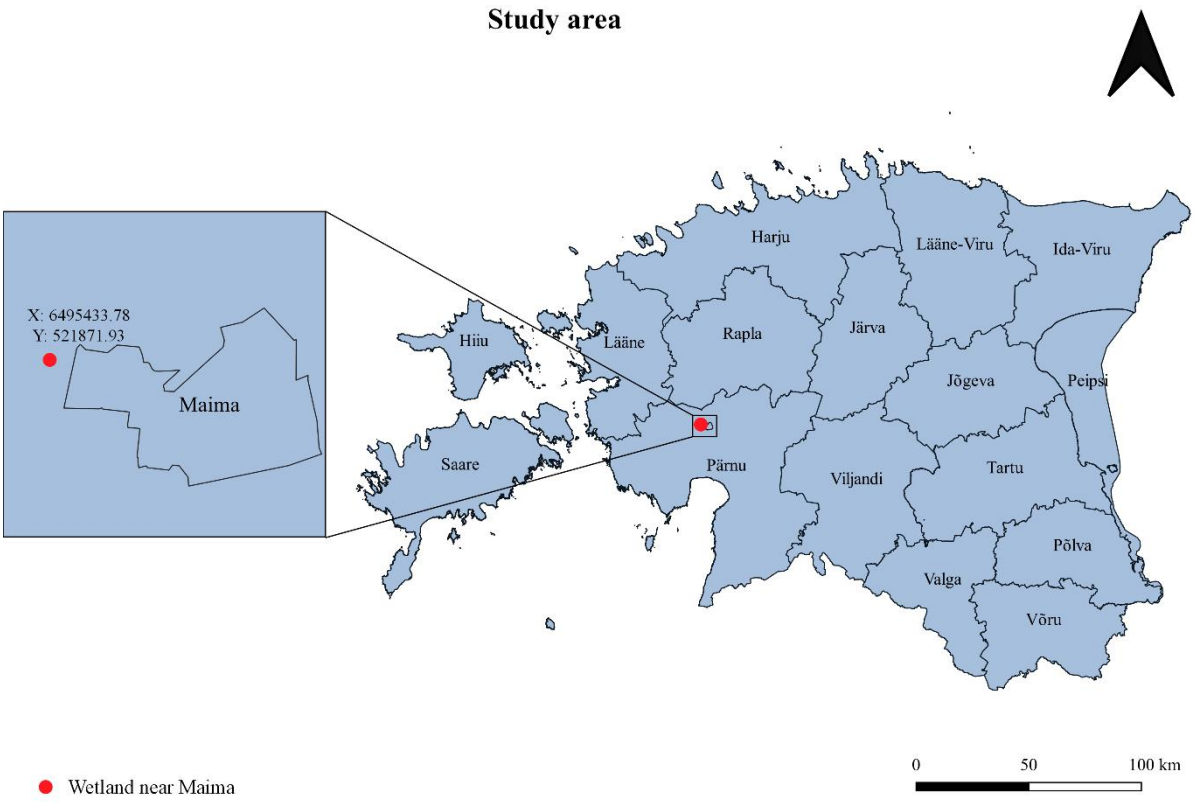


Figure 1. Study area Geographical location, (red dot on the left)

Figure 2. Study area location with orthophoto base map, source: Estonian Land board geoportal

This study area (Figure 2) has selected because it is part of cooperation projects between the University of Tartu and Estonian State Forest Management Center.

3.2 Data

UAV orthophoto and DSM

Provided data of this research includes an orthophoto (Figure 3) and a Digital Surface Model (DSM) of the study area (Figure 4). Orthophoto and DSM were assembled from 801 UAV orthophotos using DJI Mavic Pro2 drone (camera details in table 1), and 12 Ground Control Points (GCP) were used for georeferencing. Fieldworks to collect GCP's and conduct a UAV survey with additional processing to generate the orthophoto were not carried out by the author of this research. Raster images were processed using Agisoft Photoscan software.

Table 1. Camera resolution details

Camera Model	Resolution	Focal Length	Pixel Size
Hasselblad X1D-20c (10.26 mm)	5472 x 3648	28 mm	2.41 x 2.41 μm

Figure 3. Orthophoto of the study area

Table 2. Detail description of surveying data in the study area

Number of images	Flying altitude	Ground resolution	Coverage area	Tie points	Projections	Re-projection error
801	120m	2.86cm-pix	93.8 ha	1,653,793	6,508,285	0.788-pix

Digital Surface Model (DSM)

DSM represent the elevation of the surface land cover type (Figure 4). DSM is provided from stereo digital satellite images (El Garouani et al., 2014). The DSM values have been extracted for each sample point and combined with RGB values for the classification process.

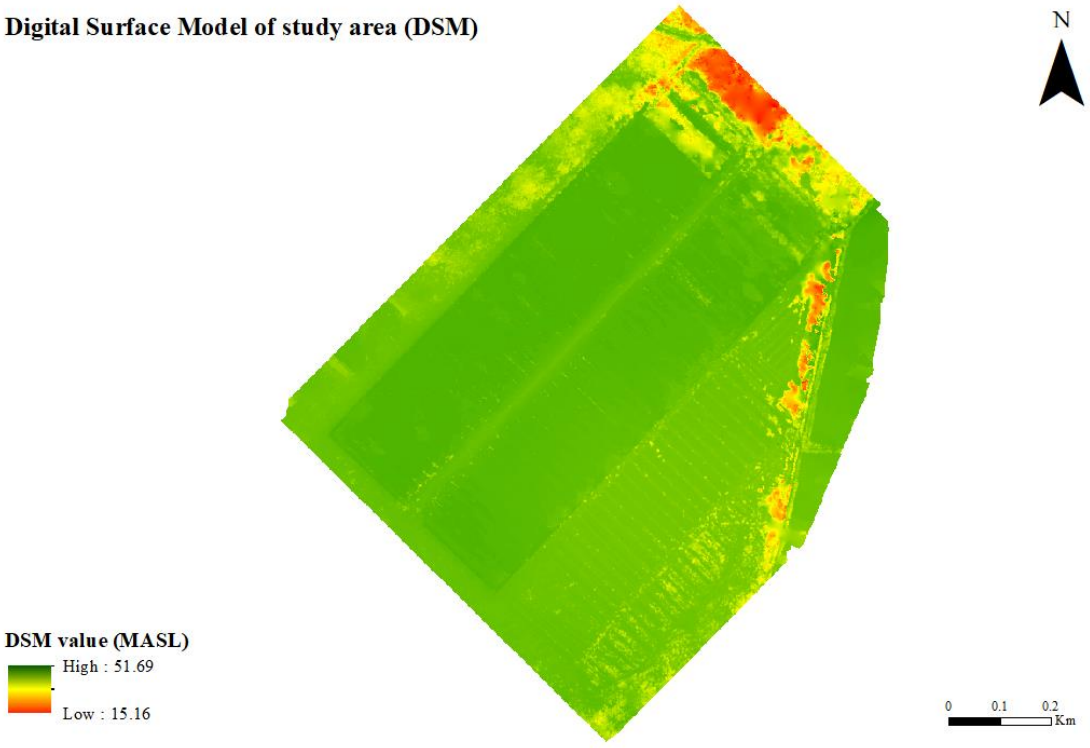


Figure 4. Digital Surface Model of the study area

Land cover classes

The second important dataset provided by Ain kull is landcover classes of the study area (Figure5). Ain Kull is a senior research fellow in physical geography at the Department of Geography. He used the same UAV orthophoto (Figure3) and combined them with additional orthophotos from Estonian Land Board and his knowledge about the area visited several times. Altogether, eight different landcover classes were defined. The final map of landcover samples on orthophoto is illustrated in figure 5. The information about each landcover type is presented in table 3.

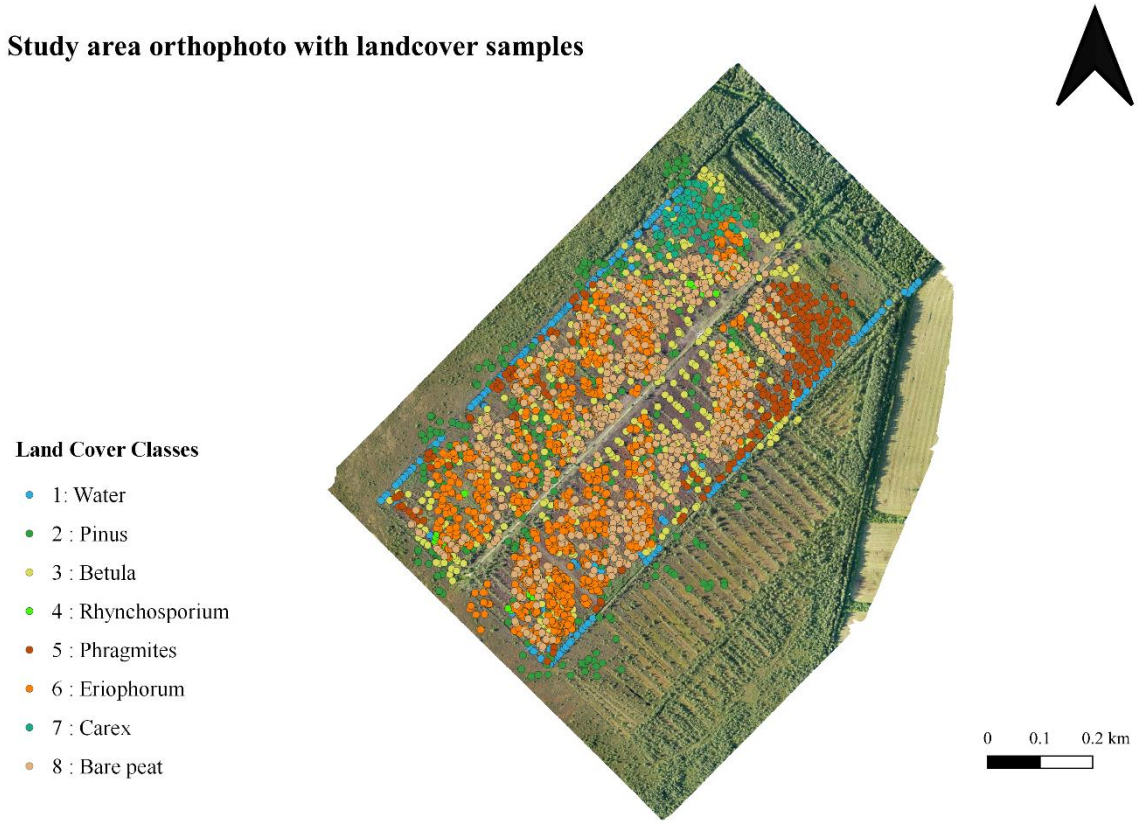


Figure 5.Distribution of landcover classes of the study area

Table 3. Land cover types description (provided by Ain Kull)

Land cover classes	Description
Bare peat	It is an unvegetated area of the abandoned milled peat extraction site. This class includes bare sphagnum peat, fen peat and heavily mineralized dark peat in depressions transported by wind and water erosion.
Betula	Mainly downy birch (Betula pubescens) growing in a row along the drainage ditches and as standalone trees on peat fields. Class includes also standing stems of dead birches along the ditches.
Carex	Six species of true sedges class are distributed mainly along wet margins of ditches and flat peat fields where fen peat is exposed.
Eriophorum	It is a class of species of flowering plant in the sedge family. The Eriophorum vaginatum (the hare's-tail cottongrass, tussock cottongrass) is a 30–60 cm high tussock-forming plant with solitary spikes and more abundant than the Eriophorum angustifolium, commonly known as common cottongrass or common cotton sedge.
Phragmites	Phragmites australis (common reed) grows along ditches where groundwater influx is more robust and forms extensive reed beds on peat fields where fen peat is exposed.
Pinus	Scots pines grow mainly as separate trees on dryer peat fields or smaller stands in more fertile parts of the abandoned peat extraction fields. They are numerous along main drainage ditches (collector ditches) on unexcavated side of the ditches and on blocks between dredged extraction strips.
Rhynchospora	Rhynchospora alba, the white beak-sedge, is a plant in the sedge family and forms smaller patches in smaller wet depressions. The plant is a perennial herb between 10 and 50 cm in height. Often the plant grows in tight clumps.
Water	Open water is in narrow ditches (1 m wide) and occasionally in depressions.

A total of 4184 of samples were taken from 8 land cover classes. It is apparent from figure 6 that the highest number of samples (1748) belongs to Eriophorum (Class 6). The lowest number of samples (28) belongs to Rhynchospora (Class 4). Images of landcover samples in the study area are illustrated in Figure 7.

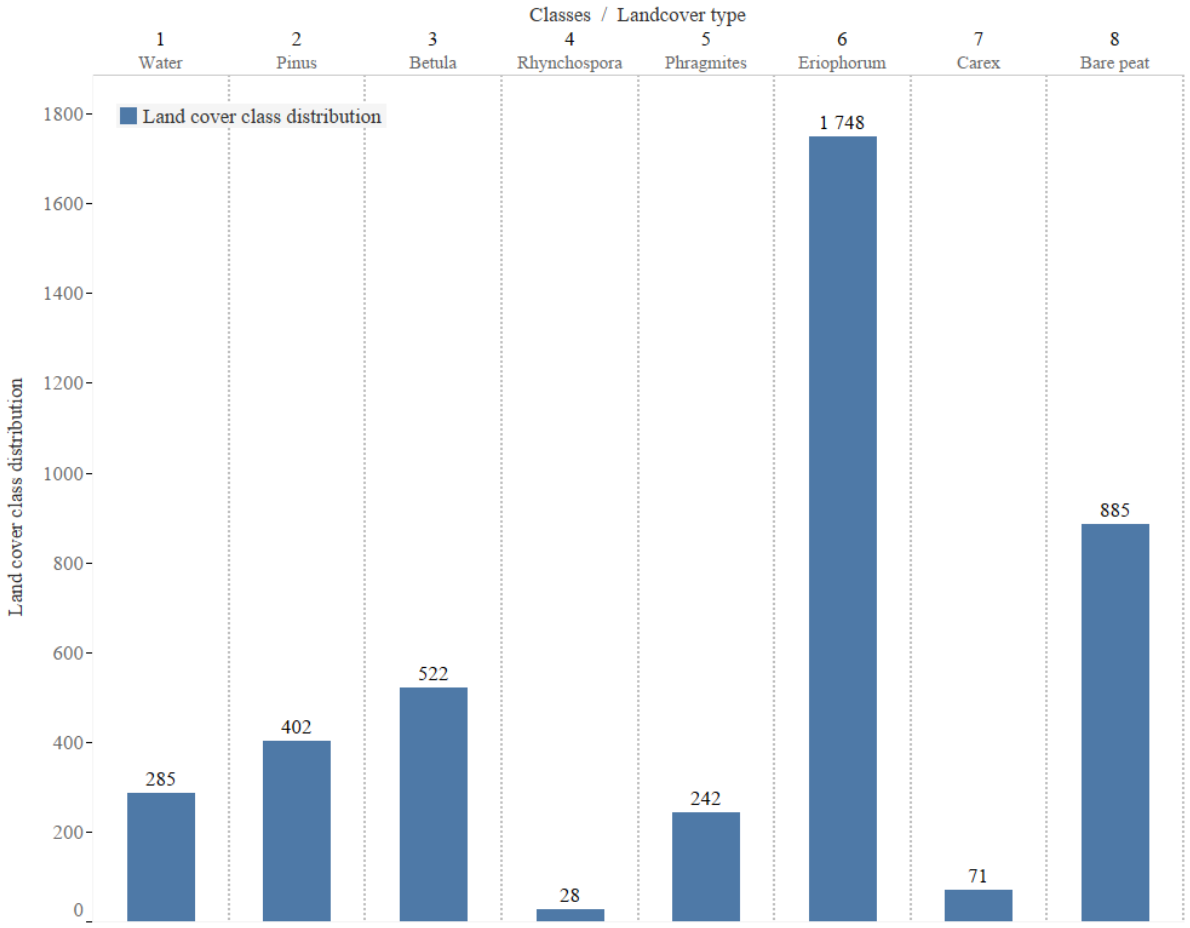


Figure 6. Bar chart of distribution of land cover classes

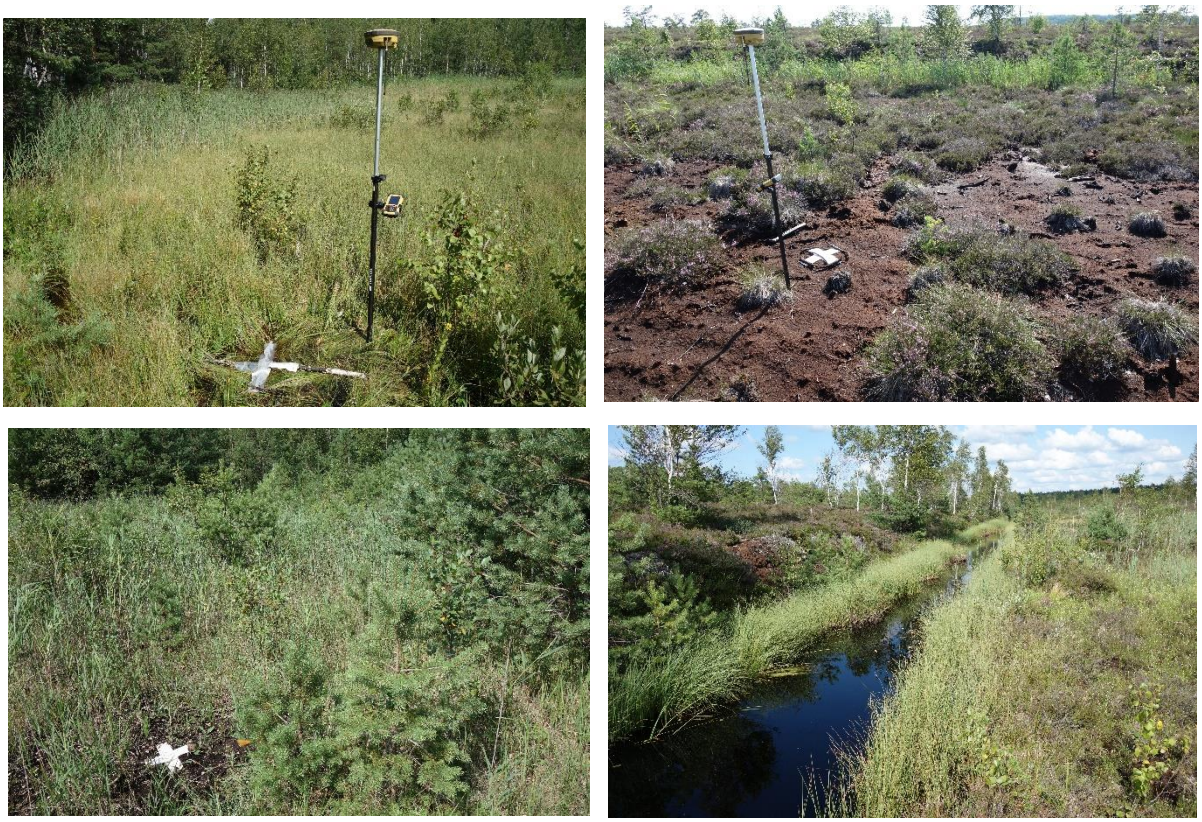


Figure 7. General view of four different land cover samples in study area. Provided by the expert (Ain Kull)

3.3 Additional constructed features

Two vegetation indices are calculated as additional constructed features to RGB and DSM values to enhance classification results. Bannari et al. (1995) reviewed the vegetation indices' application in remote sensing data and claimed that they have an essential role in the accuracy assessment of UAV image classification. The Visible Atmospherically Resistant Index (VARI) and Green Leaf Index (GLI) are two leading RBG based vegetation indices used in this study to evaluate their impact on each classifier accuracy score. These two RGB based vegetation indices in a survey by (Bannari et al., 1995), have been used without considering transformation into reflectance, atmospheric correction and sensor calibration for calibrated satellite images. The equations (6) and (7) are conducting the same approach, using RGB orthophoto bands' raw values to calculate additional features and not including any calibration and transformation.

Visible atmospherically resistant index (VARI)

This index is designed ideally for RGB images, and it is computed from digital numbers using three bands in orthophoto by equation (6). In a study by (Lussem et al., 2018) calibrated orthomosaic for estimating the surface reflectance with calibrated digital numbers for vegetation classification have been used. The vegetation indices are calculated only to find out their effect on the evaluation of classification performance. In equation 6, Green, Red and blue represent the digital number of RGB orthophoto values. (Figure 8) is illustrating the VARI as an additional constructed feature of the study area.

.

$$\mathrm{VARI} = (\mathrm{Green} - \mathrm{Red})/(\mathrm{Green} + \mathrm{Red} - \mathrm{Blue}) \quad (6)$$

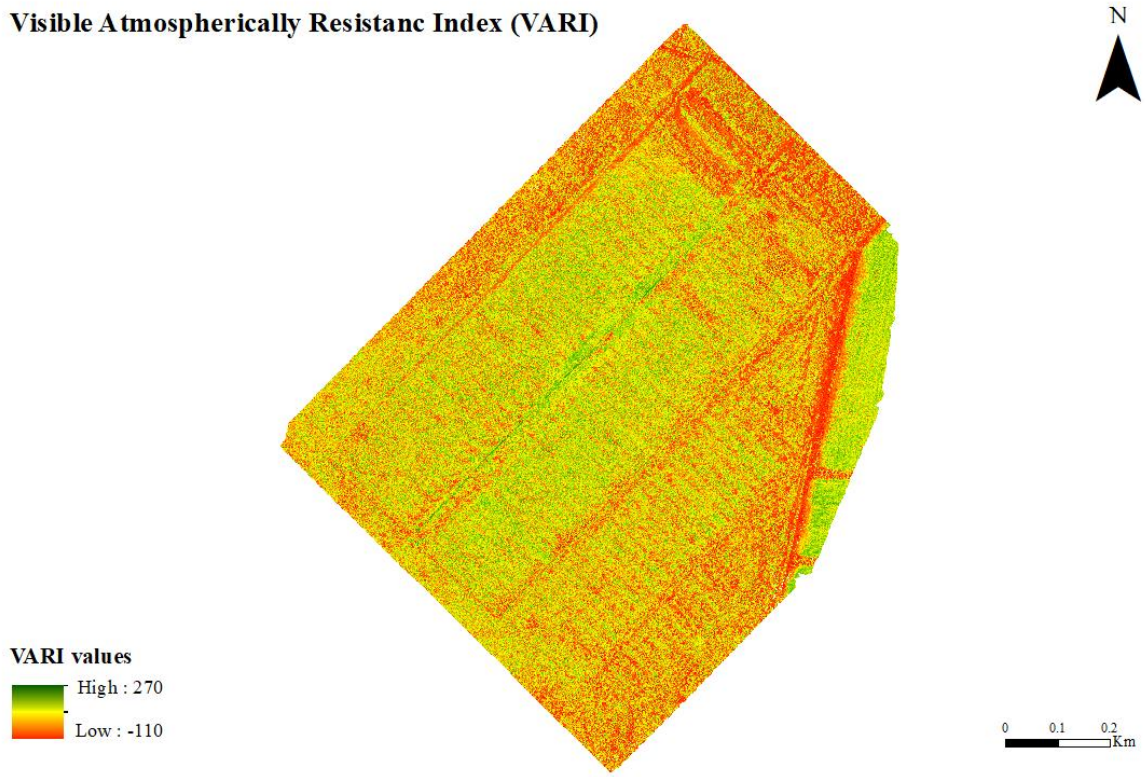


Figure 8. Visible atmospherically resistant index

Green Leaf Index(GLI)

The second additional feature is the Green Leaf Index which was made by (Louhaichi et al., 2001). GLI negative values represent soil and a non-living feature, and the positive values represent either green leaves or stems. But in this study, the mentioned description is not considered. The point for conducting the equation (7) is to calculate the GLI index based on RGB values and analyze its effect on the classification results. Figure 9 is illustrating the GLI index values made from RGB UAV orthophoto.

$$GLI = (2 * Green - Red - Blue)/(2 * Green + Red + Blue)$$

(7)

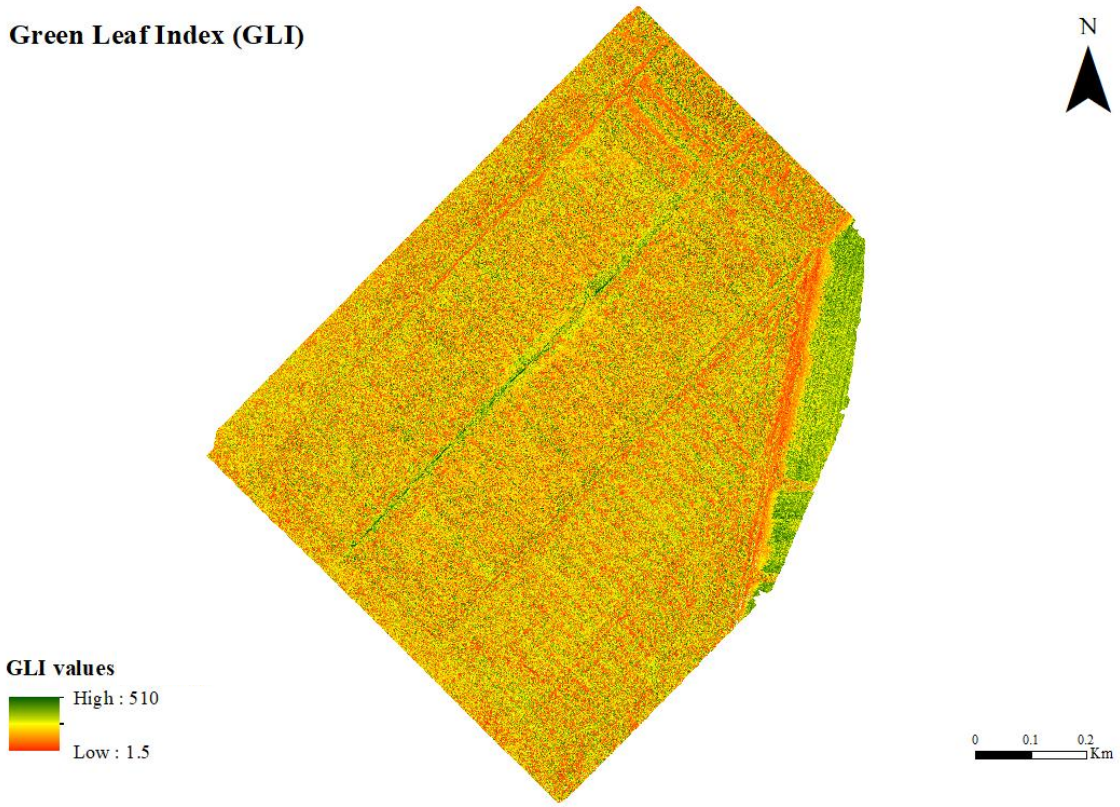


Figure 9. Green Leaf Index values

3.4 Data preparation and exploration

Data processing and classification were performed in Jupyter notebook 6.0.3 a, web-based, interactive computing notebook with python 3.8.0. Jupyter notebook was used to import original UAV orthophoto, DSM, Landcover classes and additional constructed features (GLI, VARI). Imported data was read by "*rasterio"* library. In this study, all data are projected in Estonian coordinate system of 1997, EPSG: 3301.
All imported data is gathered in a dataframe (Figure 10). The data frame values are extracted from raster images original data (UAV, DSM) and additional constructed features (GLI, VARI) using different spatial resolutions (pixel per sample) starting from 1pixel, 5x5 pixels, 11x11 pixels per sample. Figure 10 presents the main data frame for the classification process. Subsequently, a pair plot "*seaborn.pairplot"* of this dataframe was created to demonstrate the correlation between dataset features together. Figure11, highlights the correlation of 1 pixel per sample of each feature in the dataframe.

	Class	geometry	red	blue	green	DSM	GLI	VARI
0	1	POINT (521970.701 6495859.127)	3.0	22.0	46.0	18.148476	68.260872	26.934782
181	1	POINT (521795.614 6495034.264)	173.0	181.0	176.0	18.597954	356.985809	167.017044
182	1	POINT (521804.716 6495043.841)	22.0	32.0	41.0	18.640741	73.109756	30.463415
183	1	POINT (521775.146 6495143.273)	68.0	68.0	82.0	18.862532	150.085373	81.170731
184	1	POINT (521772.884 6495145.588)	77.0	75.0	66.0	18.874565	140.931824	66.833336
...	...	...	...	...	...	...	...	...
3595	8	POINT (521818.167 6495426.005)	108.0	99.0	96.0	18.817059	194.984375	103.875000
3596	8	POINT (521797.000 6495427.328)	143.0	142.0	129.0	18.359423	270.949615	128.891479
3597	8	POINT (521791.775 6495426.468)	132.0	117.0	99.0	18.522949	215.909088	112.666664
3586	8	POINT (521819.490 6495406.756)	97.0	77.0	80.0	19.583212	157.018753	98.787498
4182	8	POINT (521709.568 6494990.215)	122.0	85.0	82.0	18.731564	166.981705	117.512192

Figure 10. Data frame indicates the extracted pixel values of land cover sample points from orthophoto, DSM, VARI and GLI.

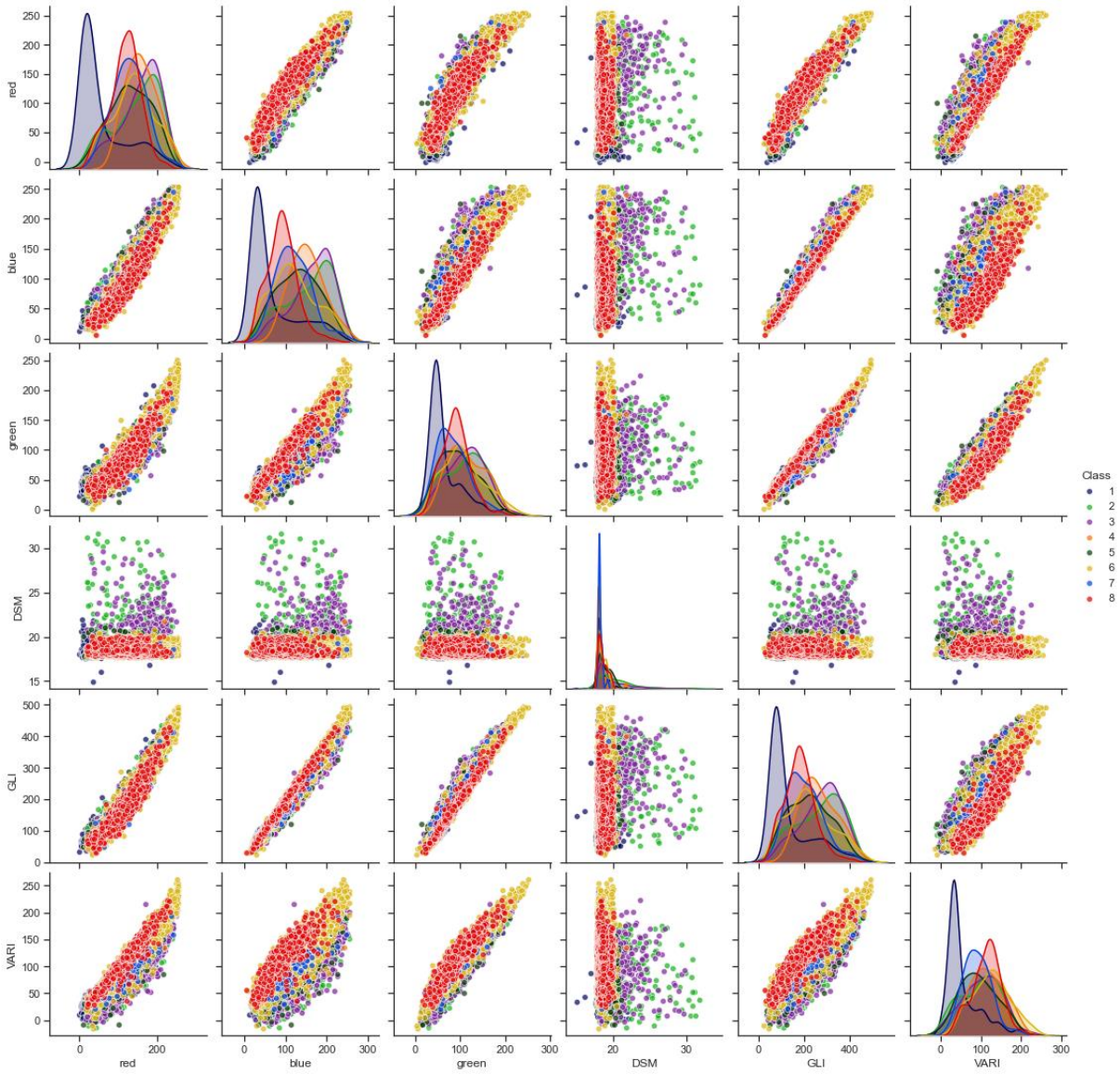


Figure 11. Pair plot visualization of the main data frame for 1- pixel data per sample

Test and Train split

The dataset made in the previous phase had to be divided into two-part training and test for initiation of the machine learning classification process. For data training, the dataframe generated in the last section is separated by 70 % for training and 30 % is assigned to validate the classifier performance. For splitting the dataframe "*train_test_split, Scikit learn"* library was used. The train and test data output is exported as CSV format. Figure 12 displays the distribution of the train and test datasets of each landcover type sample.

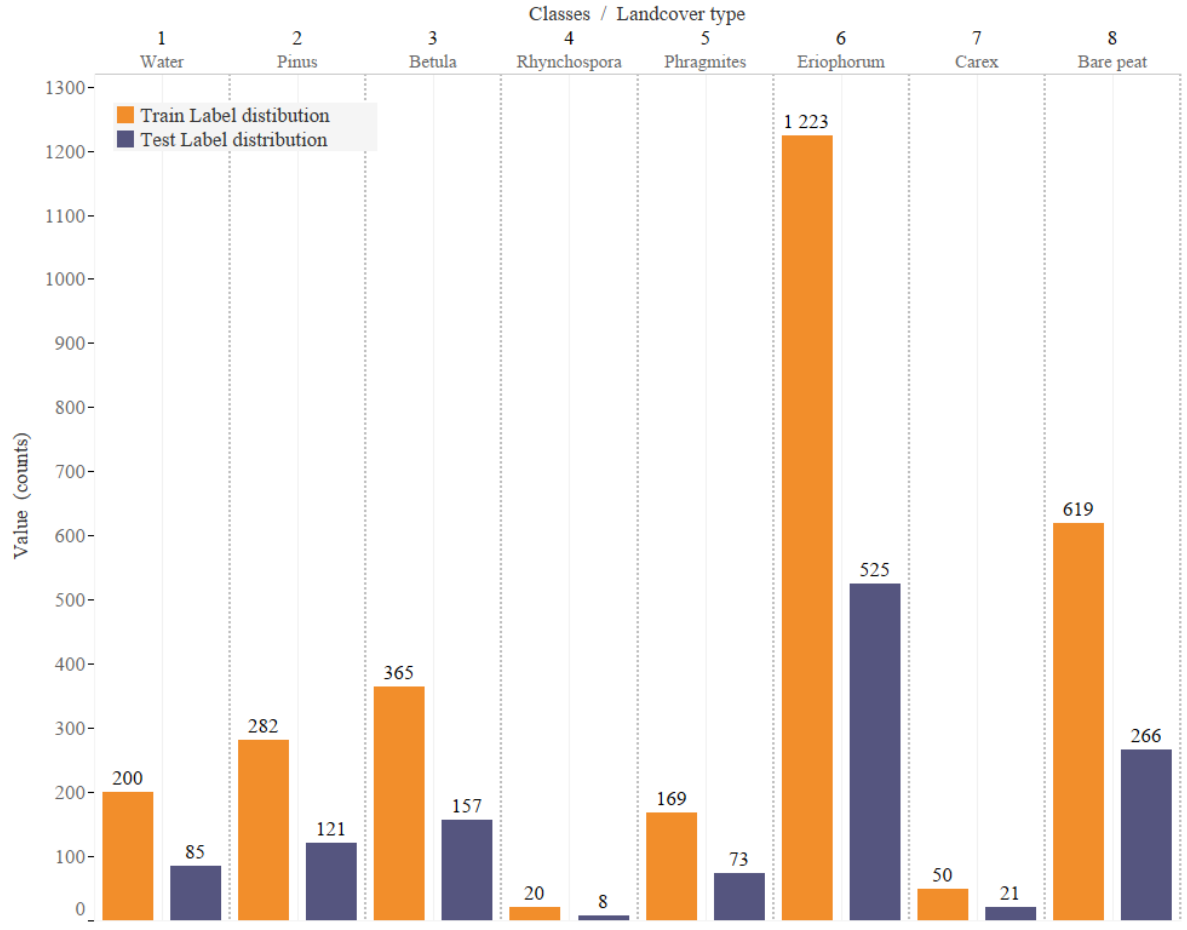


Figure 12. Distribution of classes after train and test split process.

Three different spatial samplings of train and test data have been processed in Qgis 3.14.15 desktop software.

1. Splitting data into half along the road (Train data: 2012, Test data: 2168)
2. Randomly selection of polygons (Train data: 2500, Test data: 1674)
3. Biased selection, imitating (70/30) split but with the random selection of polygons (Train data:3161, Test data: 1019)

Tools used for making these spatial distributions were as follows: vector tools ⟶ Research tools ⟶ Creating grid ⟶ Select by polygons and Select by location (Inside, within, intersect). Figure 13 represents the selection of test data by different polygons in three different distribution patterns. The number of train and test values distributed in each spatial distribution type is described in table 4.

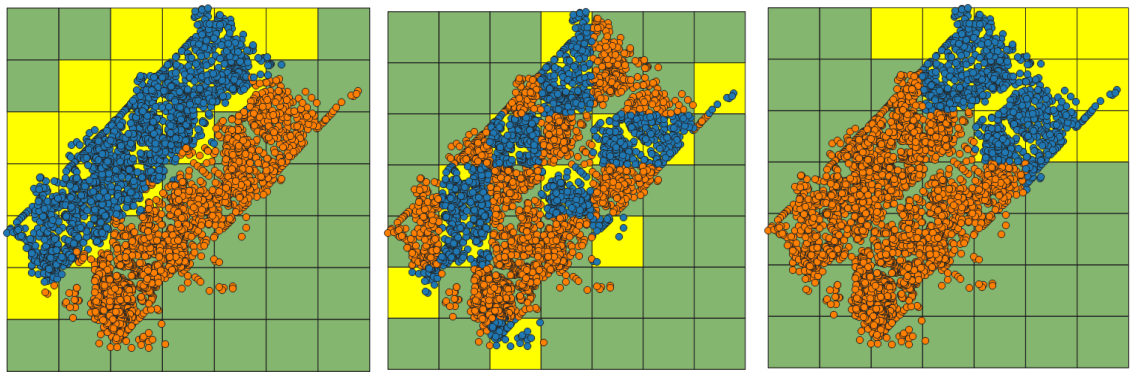


Figure 13. Different spatial sampling visualizations. Test data (blue points) inside the yellow polygons selected randomly.

Table 4: Distribution of land cover classes in 3 different spatial distribution

Classes	Spatial Distribution	1: Water	2: Pinus	3: Betula	4: Rhynchospora	5: Phragmites	6: Eriophorum	7: Carex	8: Bare peat
Train	Equal distribution	113	162	247	8	183	876	0	423
Test		171	238	275	20	59	872	71	462
Train	Random distribution	162	217	322	25	127	1127	37	489
Test		122	183	200	3	115	621	34	398
Train	Biased distribution	195	293	370	24	105	1538	0	636
Test		89	107	152	4	137	210	71	249

Training ML models

The main steps in jupytor notebook's workbooks are outlined in Figure 14.
In total, there are two qualified models for each of the three classifiers (RF, KNN, SVM): default models (default parameter' values of classifier) and models with Hyperparameter tuning. The features are manipulated with additional constructed features and ML concepts. Transforming the existing ones (scaling) or manipulating their distribution (oversampling) has different combinations in both default and hyperparameter models.
Altogether, RF, SVM and KNN have 40 different models added by three spatial sampling methods. As a result, 120 models and data sampling combinations were generated in this study.

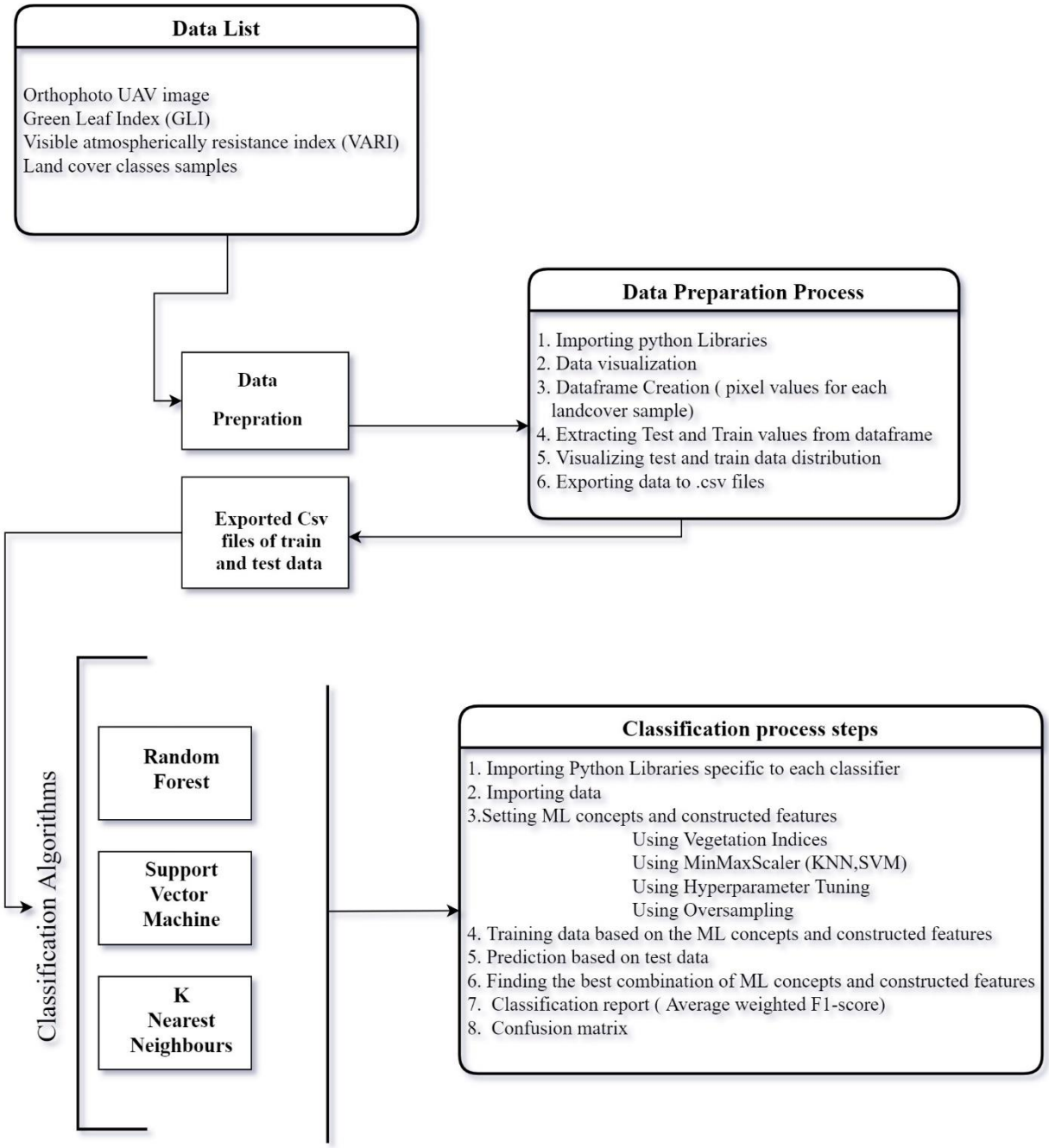


Figure 14. General description of classifier's workbooks

Model groups

In default models, the default values of each classifier parameter values are used. The values are from the library documentation of each classifier.

Table 5. Default parameters of each model classifier

Default models	Default models parameters
RF	N_estimators=100, criterion=gini, max_depth=None, min_samples_split=2, min_sample_leaf: 1, max_features = auto, random_state=42, max_samples=none
SVM	C=1, kernel = rbf, degree=3, gamma-scale, verbose=False, random_state=none
KNN	N_neighbours =5, weights=uniform, metrics=Minkowski

In the hyperparameter tuning model, a series of values is selected based on each classifier's library.

Table 6. Hyperparameter tuning parameters of each classifier's model

Hyperparameters models	Hyperparameters models parameters
RF	"n_estimators": [100, 200], "max_depth":[3, 5, 6, 9, 15], 'min_samples_split': [11, 15, 19], 'min_samples_leaf': [3,5,7], "bootstrap":[True,False],"max_features":['auto']
SVM	"kernel": ['rbf'], 'C': [1,10],'gamma': [0.001,0.1], 'degree': [1, 2]
KNN	"n_neighbors": [9,11,19,35,51], "weights":["uniform","distance"], "metric": ["euclidean","manhattan"]

Additional constructed features/ Vegetation indices: Adding two columns of GLI and VARI which were calculated from the UAV orthophoto to analyse their effect on classification accuracy.
MinMaxScaler: By applying this scaler, existing features are transformed into a number between (0 to 1). This step is not included in the random forest model.
Oversampling: The instances of each class were equally distributed by conducting random oversampling.
Spatial resolution: Data for landcover classes is obtained as points. When spatial sampling of 1 pixel is used, then the feature values are extracted from the one pixel where the point is located. In the spatial sampling with 25 pixels, a buffer for 2 pixels in each side results in 5x5 pixels. The feature value is calculated as the average of all these pixels. Similarly, the 5pixel buffer will result in size 11x11 pixels with 121 pixels, and the feature value is the average of all those pixels.
Spatial sampling: Three different spatial samplings (distribution of train/ test data) have been conducted to evaluate classifier's performance.
Table 7 represents ML concepts' combination with additional constructed data for both default and hyperparameter tuning models for three classifiers.

Table 7. ML classifiers with additional constructed data and ML concepts feature's combination

Classifiers	Model groups	ML concepts and features	Spatial resolution
RF	**Default models** (default parameters)	Base features (RGB+DSM) Vegetation indices (GLI+VARI) MinMaxScaler Oversampling Vegetation Indices+ MinMaxScaler Vegetation Indices+ Oversampling MinMaxScaler +Oversampling Vegetation Indices+MinMaxScaler+ Oversampling	1px 25px 121px
KNN SVM	**Models with Hyperparameter tuning**		

After analysing ML models group based on table7, the classifier with better performance than others are selected for UAV orthophoto classification. Figure 15 describes the steps for calculating the distance between the train and predicted labels for each landcover types. In this process, two raster files are created, one based on a constant value (value = 1) and the other for calculating the distance between training data and predicted labels. Sampling raster tool extracted the distance values (meter) and added a new column into the predicted points shapefile.

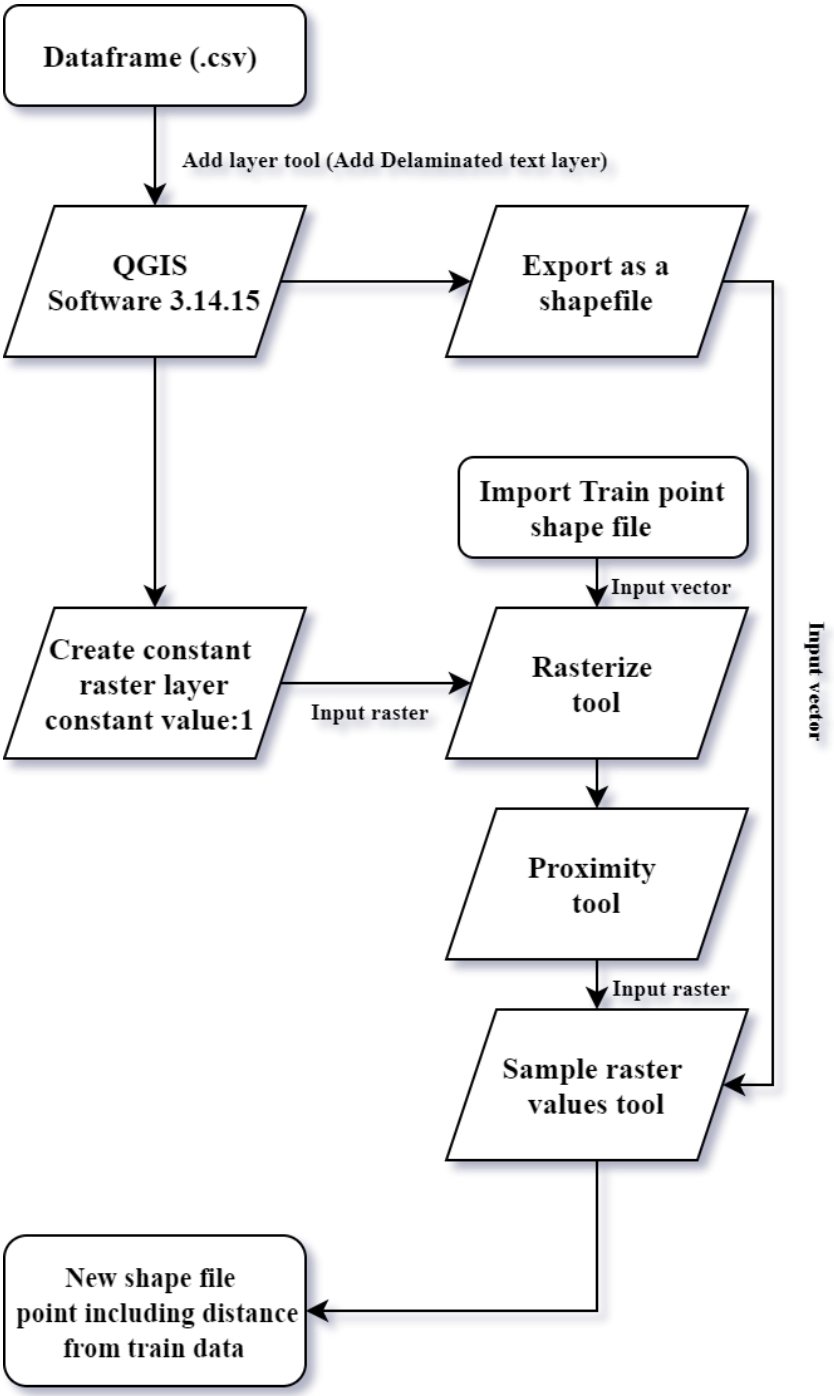


Figure 15. Flow chart of calculating distances between the train and test data with three different datasets.

4 Results

The overall accuracy of RF, SVM and KNN for classification of 8 land cover types, including different spatial samplings and resolutions with different ML concepts, is shown in Table 8.

Table 8. Overall Accuracies (Average Weighted F1-score) of all models including feature variation with different spatial sampling

Classifiers	Models	ML concepts features	1pix	25pix	121pix
RF	Default models	Basic	0.52	0.55	0.58
		Vegetation indices	0.53	0.57	0.59
		Oversampling	0.51	0.54	0.56
		Vegetation indices + Oversampling	0.52	0.55	0.58
	Models with Hyperparameter Tuning	Basic	0.52	0.56	0.56
		Vegetation indices	0.50	0.58	0.59
		Oversampling	0.51	0.53	0.57
		Vegetation indices + Oversampling	0.50	0.53	0.57
KNN	Default models	Basic	0.53	0.54	0.56
		vegetation indices	0.50	0.54	0.57
		MinMaxScaler	0.53	0.54	0.57
		Oversampling	0.43	0.46	0.48
		Vegetation indices + MinMaxScaler	0.53	0.54	0.58
		Vegetation indices + Oversampling	0.41	0.46	0.49
		MinMaxScaler + oversampling	0.42	0.46	0.49
		Vegetation indices +MinMaxScaler + Oversampling	0.43	0.45	0.50
	Models with Hyperparameter Tuning	Basic	0.51	0.53	0.55
		vegetation indices	0.51	0.52	0.55
		MinMaxScaler	0.50	0.52	0.56
		Oversampling	0.44	0.47	0.51
		Vegetation indices + MinMaxScaler	0.51	0.54	0.57
		vegetation indices + Oversampling	0.43	0.47	0.51
		MinMaxScaler + Oversampling	0.44	0.46	0.50
		Vegetation indices + MinMaxScaler+ Oversampling	0.45	0.48	0.51
SVM	Default models	Basic	0.40	0.47	0.52
		vegetation indices	0.40	0.46	0.53
		MinMaxScaler	0.48	0.52	0.55
		Oversampling	0.39	0.44	0.50
		Vegetation indices + MinMaxScaler	0.49	0.54	0.57
		Vegetation indices + Oversampling	0.39	0.44	0.49
		MinMaxScaler + Oversampling	0.42	0.48	0.51
		Vegetation indices + MinMaxScaler+ Oversampling	0.42	0.49	0.52
	Models with Hyperparameter Tuning	Basic	0.42	0.47	0.54
		vegetation indices	0.39	0.52	0.54
		MinMaxScaler	0.35	0.41	0.42
		Oversampling	0.39	0.44	0.51
		Vegetation indices + MinMaxScaler	0.33	0.40	0.42
		Vegetation indices + Oversampling	0.30	0.55	0.56
		MinMaxScaler + Oversampling	0.41	0.41	0.48
		Vegetation indices + MinMaxScaler +Oversampling	0.37	0.42	0.44

Overall, the conditional formatting in table 8 presents that model groups with 121 pixels performed better than 1 pixel and 25 pixels' models. All three classifiers always followed this trend in both default and models with Hyperparameters tuning. For each classifier (RF, KNN and SVM), the optimal combination of features leading to the highest accuracy is as follow:

Table 9. The highest accuracy (average weighted f1-score) in Random Forest

Classifier	Models	ML concepts and additional features	1pix	25pix	121pix
RF	Default	Vegetation Indices (GLI+VARI)	0.53	0.57	0.59
	Hyperparameter Tuning		0.50	0.58	0.59

In table 9, the best performance of RF classifier models with ML concept features is represented. The RF classifier achieved 0.59 accuracy in both default and Hyperparameter models by applying the same combination of vegetation indices (GLI+ VARI) to the training model with 121 pixels per sample. The best hyperparameter values provided by hyperparameter tuning are:
Best parameters = {'Bootstrap': True, 'max_depth': 15, 'max_features': 'Auto', 'min_sample_leaf': 3, 'min_samples_split':19, 'n_estimator' :200}
The accuracy of cross-validation with the best parameter's on train data is (0.63) and (0.60) on test data.

Table 10. The highest accuracy (Average weighted score) in K-nearest neighbours

Classifier	Models	ML concepts and additional features	1pix	25pix	121pix
KNN	Default	Vegetation indices + MinMaxScaler	0.53	0.54	0.58
	Hyperparameter Tuning		0.51	0.54	0.57

KNN best performance in both default and hyperparameter tuning models is shown in Table 10. The combination of vegetation indices and MinMaxScaler with 121-pixel spatial resolution is the main factor for the highest accuracy in these models. By looking at table 10, it is apparent that the default model classifier's of KNN performed better than hyperparameter tuning model. The best hyperparameters achieved by cross-validation are as follow:
Best parameters= {'metric': 'Euclidean', 'n_neighbours': 51, 'weight': 'distance'}.

The hyperparameter tuning model's accuracy with cross-validation and best parameters is 0.62 on train data and 0.58 on test data.

Table 11. The highest accuracy (average weighted f1-score) in Support Vector Machines

Classifier	Models	ML concepts and additional features	1pix	25pix	121pix
SVM	Default	Vegetation indices + MinMaxScaler	0.49	0.54	0.57
	Hyperparameter Tuning	Vegetation indices + Oversampling	0.30	0.55	0.56

SVM classifier reached its lowest accuracy in both default and hyperparameter tuning models with 1 pixel per sample same as the two other classifiers. The combination of vegetation indices and oversampling in the hyperparameter tuning model achieved better performance (0.56) with 121 pixels. In comparison, the default model gained a better result with 121 pixels with vegetation indices and MinMaxScaler (0.57).
The best hyperparameter values which were used in the classification process are:
Best parameters = {'C':10, 'degree': 1, 'gamma': 0.1, 'kernel': 'rbf'}.
The hyperparameter model's accuracy with cross-validation on train data is 0.94, and on the test data is 0.47.
Overall, each classifier's result in both default and hyperparameter tuning models was not satisfying, and the highest accuracy was approximately 60%.

Three highest accuracies achieved by RF, SVM and KNN are:
Default model RF + Vegetation Indices = 0.59
Default model KNN + Vegetation Indices + MinMaxScaler = 0.58
Default model SVM + Vegetation Indices + MinMaxScaler = 0.57

4.1 Classified UAV Orthophoto

The UAV orthophoto's original data was resampled into a lower spatial resolution of 31.46 cm (121 pixels). RF default model with vegetation indices was used to classify the orthophoto, and the result can be seen in Figure 16.

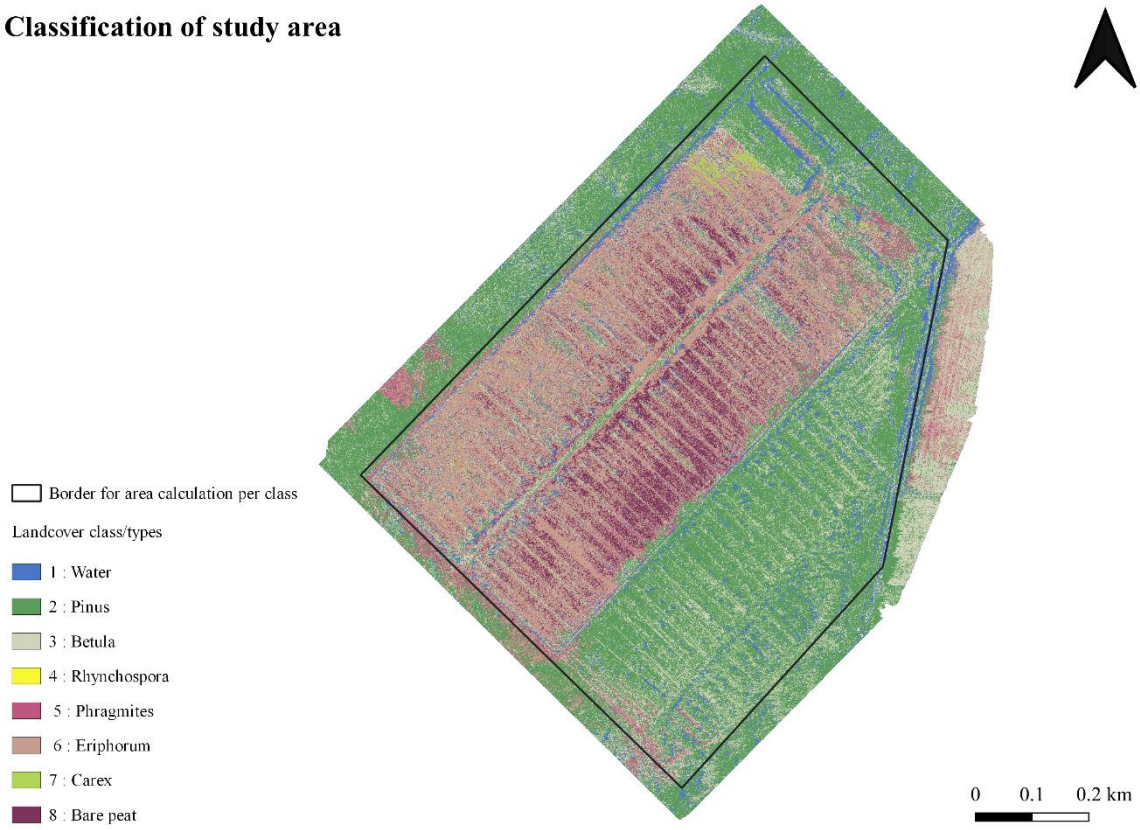


Figure 16. Classified UAV orthophoto of study area

In table 12, the area of each landcover class after classification is represented. Rhychospora (Class 4) has the lowest area and Pinus (Class 2) with 31.5% has the highest area.

Table 12. Area of landcover classes after classification

Landcover type/classes	**Area (m^2)**	**Area (ha)**	**Area (%)**
1: Water	53085	5.3	7.4
2:Pinus	226595	22.7	31.5
3: Betula	125334	12.5	17.4
4: Rhynchospora	937	0.1	0.1
5: Phragmites	39782	4.0	5.5
6: Eriophorum	207224	20.7	28.8
7: Carex	6322	0.6	0.9
8: Bare peat	59230	5.9	8.2
Total	718509	71.9	100.0

5 Discussion and Conclusion

5.1 Applying machine learning concepts

Based on the machine learning concepts and features in table 7, the results of all three classifiers showed approximately similar behaviour. Table 8, represented that the accuracy results for all three classifiers by having hyperparameter model either remained stable or reduced compared to the default model of each classifier. In RF classifier, the highest accuracy results (0.59) was achieved in both default and hyperparameters tuning models with vegetation indices. In two other classifiers (KNN, SVM) apart from some exceptions such as the KNN default model (1pixel), SVM default model (25pixels), and SVM hyperparameter model (1 pixel) with default values, the vegetation indices improved the performance of the classifier. MinMaxScaler was not as useful as vegetation indices, and the results were fluctuating in both SVM and KNN models. In the SVM hyperparameter model with all three spatial resolutions, the MinMaxScaler improved the accuracy significantly from (0.40, 0.47, 0.52) to (0.48, 0.52, 0.55).
On the other hand, all three classifiers' accuracy in both default and hyperparameter tuning models after adding oversampling to the training model reduced considerably.

5.1.1 Additional constructed features

The two constructed features Green Leaf Index (GLI) and Visible Atmospherically Resistant Index (VARI) have been calculated in this study to evaluate their effect on the accuracy of each model classifiers. This study does not guarantee that GLI and VARI are authentic and interpretable as their definition in the theory section.
Constructed features have contributed significantly to the increase of ML concepts' accuracy (Tables7, 8 and 9); however, the SVM hyperparameter tuning model's accuracy decreased from 0,42 to 0,39 after adding vegetation indices. KNN default model accuracy with 1pixel spatial resolution model decreased from 0.53 to 0.50 after applying vegetation indices. A study by (Jónsson, 2019) proved that the constructed additional features such as vegetation indices had improved the model's accuracy.

5.1.2 Data scaling

MinMaxScaler has normalized the values between (0 to 1) in both SVM and KNN classifiers. The transformation of values has changed the accuracy in default and hyperparameter models. For example, in KNN, the default model's accuracy remained stable after MinMaxScaler, but it decreased in the hyperparameter tuning model. The same pattern followed for SVM classifier but with lower accuracies. The combination of MinMaxScaler and vegetation indices in SVM and KNN provided the model's highest accuracy results. A study by (Borkin et al., 2019) has investigated the effect of MinMaxScaling on the classification after having the raw data results. It confirms that the performance was better after the data normalization with supervised classification.

5.1.3 Oversampling

Except in certain combinations with other ML concepts, the oversampling concept decreased the classifiers' accuracy in different spatial resolution (Table 8). Random oversampling did not affect the accuracy the way it was expected because the data sets for this classification were too imbalanced. Random oversampling was skewed towards lower-instances classes such as (4: Rhynchosporium and 7: Carex) with only 28 and 71 samples respectively. Different studies have tried to develop a solution to establish the new artificial instances for the classes with lower samples (Castellanos et al., 2018) to overcome the imbalanced dataset issue. However, another study (Bunke & Riesen, 2012) stated that although the different solutions for balancing the dataset could work, they often decreased the classifier's output. In the theory part, several studies related to oversampling proposed positive results, while this research's accuracy results contradict them. There is no definitive solution to conclude that oversampling is not beneficial for image classification. Still, it can be strongly verified that it depends on the quality and quantity of the dataset.

5.1.4 Hyperparameter tuning

In general, conducting hyperparameter tuning did not function as it was expected and defined in theory. Several studies confirmed the classifier accuracy enhancement by hyperparameter tuning (Claesen et al., 2014). It also emphasized making incorrect decisions for hyperparameters value will lead to the low performance of the classifiers.
The accuracy result of each classifier (RF, KNN, SVM) with hyperparameter tuning did not follow the same pattern, and it fluctuated with a different combination of ML concepts.
There are related studies that have investigated the performance of tuning and not tuning the ML classifiers. Weerts et al. (2020) applied methods to examine whether using hyperparameter or safely leaving the parameters to their default values. Authors concluded that having some hyperparameters at their default value can improve the model accuracy results. A research by (Probst et al., 2018) introduced the hyperparameters' tunability. The authors compared the model's performance with and without hyperparameter tuning. Consequently, most of the hyperparameters left at their default value. It has been mentioned that in these studies, the default parameters are determined by minimizing the average risk.

5.1.5 Combination of ML concepts

After applying ML Concepts on classifier’s models, the effect of their combination has been investigated as well. The combination of MinMaxScaler and Vegetation indices in the KNN and SVM default model provides their highest accuracy result (0.58, 0.57). On the other hand, the lowest accuracy was provided by the combination of vegetation indices with oversampling in both default and hyperparameter models (Table 6).
The same pattern followed in KNN, and the combination of ML concepts with oversampling was between the lowest accuracies in all three spatial resolutions. The only exception is a combination of oversampling and vegetation indices in RF classifier. It can be said that RF classifier was more flexible with oversampling than the two other classifiers.
One of ML concepts' most impressive combination was that all the classifiers accuracy results increased after lowering the spatial resolution except in one or two ML concepts combinations. For example, in SVM hyperparameter model with MinMaxScaler and Oversampling, the accuracy remained stable (0.41). The most significant improvement was in the SVM

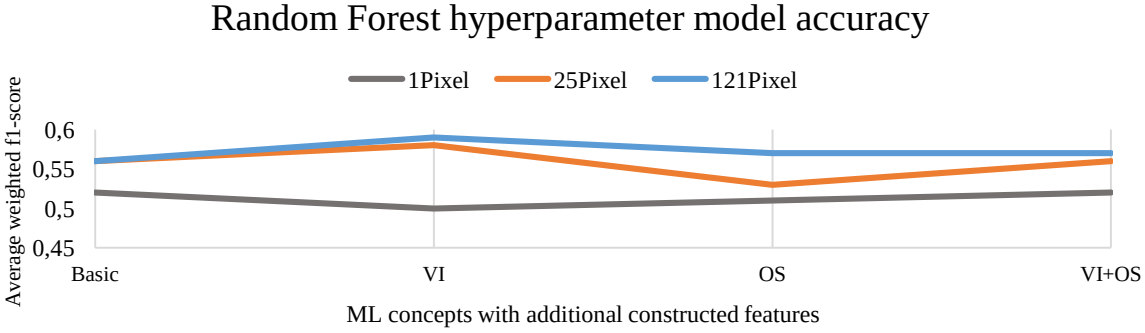
hyperparameter model with Vegetation indices and oversampling. The accuracy increased from 0.30 to 0.55 after lowering the spatial resolution (Table 8).

5.2 Comparison of RF, KNN and SVM Classifier

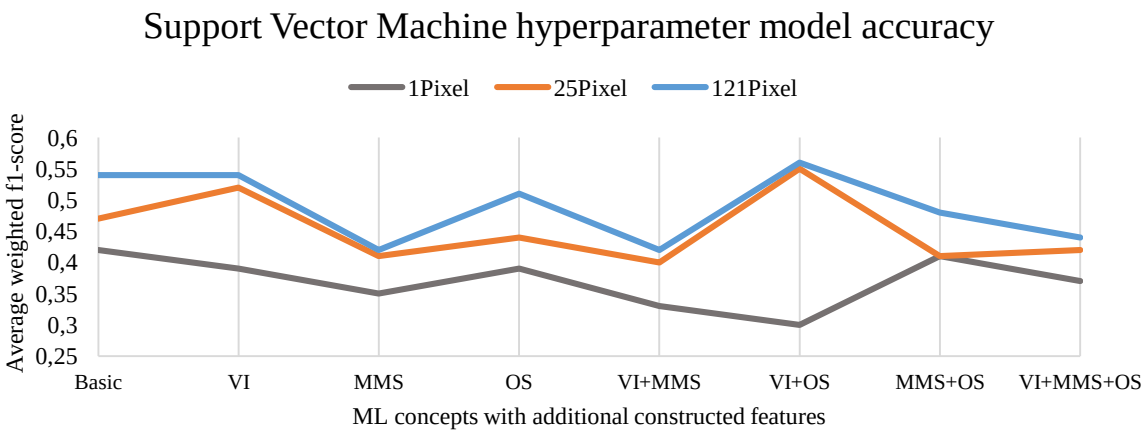
All three classifiers (RF, KNN, SVM) have approximately similar results after classification of landcover classes by similar ML concepts. Two different studies by (Adam et al., 2014) and (Ghosh & Joshi, 2014) have reached the similar results after conducting SVM, RF for classification of different landcover vegetation with satellite imagery data. Adam et al. (2014) by considering various bands of high spatial resolution satellite data received a high accuracy for RF and SVM, respectively (0.94, 0.92).

Applying Object-Based Segmentation on Bamboo species classification with SVM classifier, the accuracy increased from 0.82 to 0.94 (Ghosh & Joshi, 2014).
Thanh Noi & Kappas, (2018) compared three classifiers (RF, KNN, SVM) performance with sentinel-2 imagery for landcover classification with different training sample sizes. All three classifiers showed similar results around 93.85%, but SVM reached the highest accuracy and followed consecutively by RF and KNN. Another similar study has chosen the SVM the best classifier for multi-label classes (Pal & Mather, 2005). Contrary to the result of different comparison studies, RF has gained the highest accuracy in this study and is followed by KNN and SVM. The result of the SVM model with hyperparameter tuning was surprisingly lower than the other two classifiers. The most insufficient accuracy in all three classifiers was in SVM hyperparameter model with the combination of Oversampling and MinMaxScaler (0.30).
Also, it can be said that lowering the spatial resolution has effected the SVM model accuracy result greatly in comparison to KNN and RF. Wang & Xue (2014) investigated classifiers' behaviour with imbalanced data set, and SVM classifiers model was the most flexible one. In this study, the results of SVM classifier's model after applying oversampling decreased considerably as well. The following graphs (Figure 17) illustrate a better understanding of the classifiers' lowering spatial resolution effect with different ML concepts combinations.
The acronyms used in the x-axis of line graphs in Figure 17 are stand for algorithms parameters below:

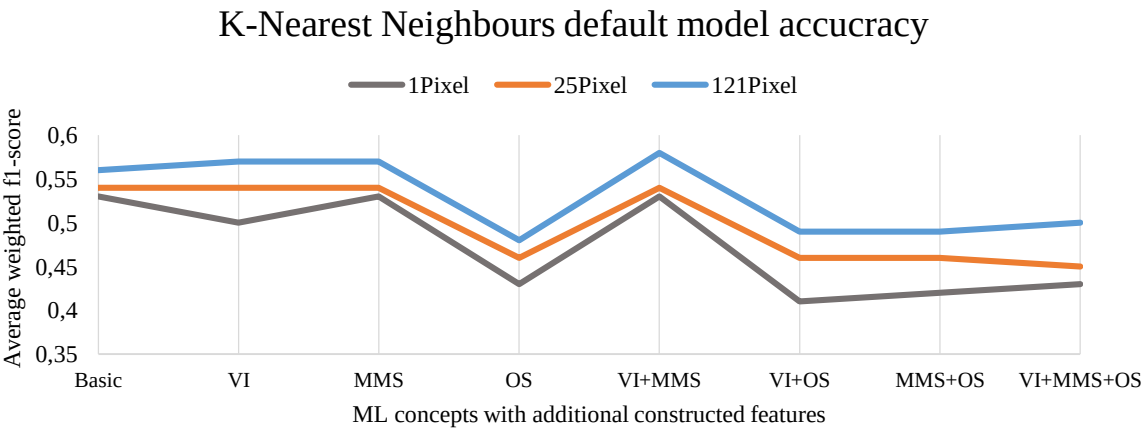
- VI: Vegetation Indices (GLI and VARI)
- MMS: MinMaxScaler
- OS: OverSampling



(1)



(2)



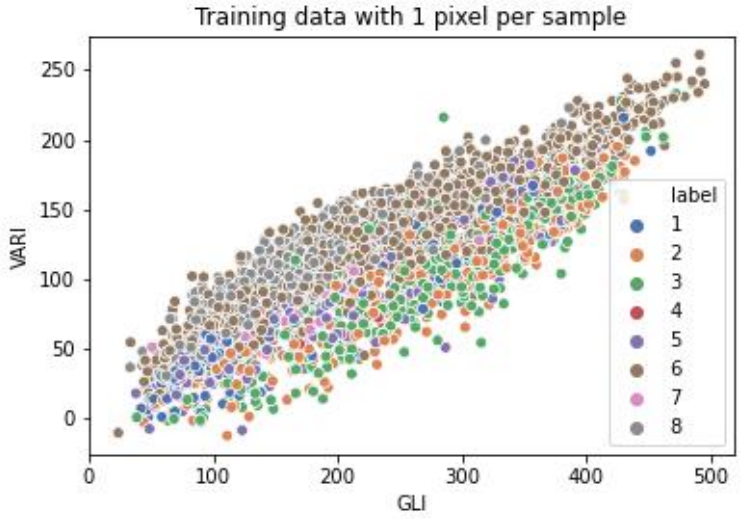
(3)

Figure 17. Three most conspicuous effects of lowering spatial resolution with different ML concepts and constructed features

5.2.1 Spatial resolution

From line graphs (1), (2) and (3) in figure 17, the interesting changes in the results of the classifier by lowering the spatial resolution is visible. In graph 1, RF hyperparameter model with 25 pixels' spatial resolution showed a significant drop after adding the OS. But in the same model with 121 pixels, this concept did not change the accuracy significantly. In the SVM hyperparameter model (graph 2) the accuracy results fluctuated in all three spatial resolutions, the lowest accuracy achieved by 25 pixels is with MMS and OS. There is no similar pattern followed in all three spatial resolutions in this model while in KNN default model, all spatial resolution accuracy result followed a similar pattern. KNN default model accuracy improved by VI and MMS, and it decreased significantly by applying OS to training model.

Lowering the spatial resolution (1 pixel, 25 pixels, 121 pixels) has significantly affected all classifiers' accuracy, as mentioned above. Figure 21 represented the effect of increasing the number of pixels per sample for extracting the data. In scatter plot (1), not so easy to differentiate the classes from each other. Still, after creating a buffer with 121 pixels for each sample (scatter plot (2)), the classes are more discernible from one another. With generalization (making sampling area bigger) the separation of classes is better. The limit for having generalization is based on the borders of the classes. Segmentation (grouping similar pixels together) would be a much smarter way of generalizing.



(1)

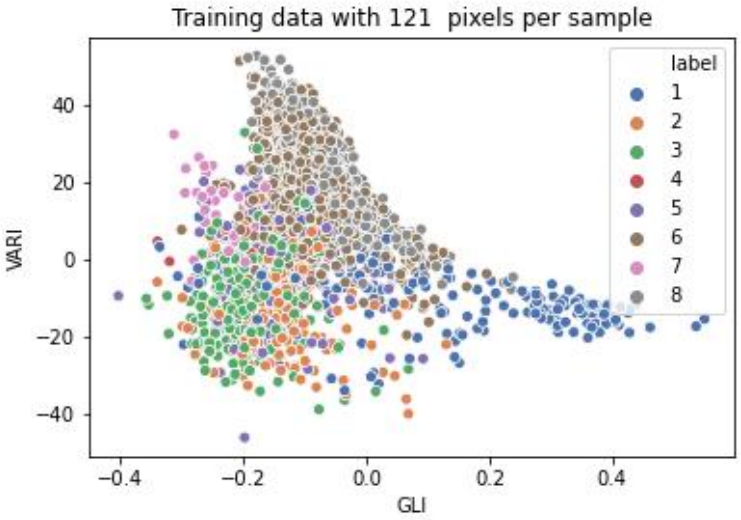


Figure 18. Comparison of class distribution after lowering spatial resolution

(2)

Having noise and uncalibrated images may be the key explanation for not having separating classes from one another. Image calibration is an essential factor for obtaining higher accuracy.

5.2.2 Random sampling vs spatial sampling

Three different spatial samplings (Figure 13) have been used to investigate different spatial sampling effect on each classifier's accuracy. As it was assumed, the random sampling (70/30-train/test) gained a higher accuracy in comparison to the other three spatial samplings.
Table 13 is representing the accuracy results of optimal classification models by three spatial samplings.

Table 13. Accuracy of weighted average f1-score of three different spatial sampling

	Different distribution of Train and Test data		
Optimal ML Concepts	Equal distribution	Random distribution	Biased distribution
RF default model + Vegetation indices	0,49	0,54	0,35
SVM default model + Vegetation indices + MinMax scaler	0,49	0,46	0,34
KNN default model + Vegetation indices + MinMax scaler	0,46	0,55	0,47

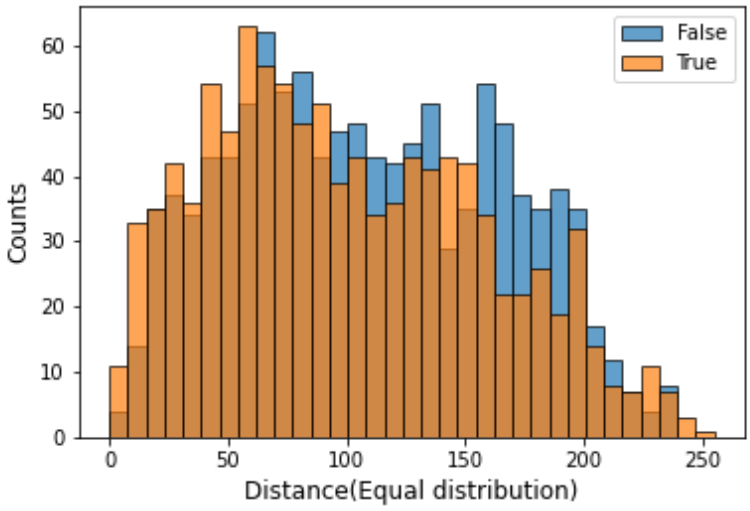
Table 14 represents each classifier's result based on different spatial sampling on predicting the class labels. The number of True labels in Equal distribution is higher than random and biased distribution. On the contrary, in Equal distribution, the number of false labelled data is significantly high. The same pattern is followed on biased distribution but with a lower number of true and false labelled data. That being said, random distribution has shown better results in predicting the true label than false labels in all three classifiers. Another study by (Colditz, 2015) investigates the result of classification with different training sizes, leading to a lower accuracy level than random sampling.

Table 14. Result of different distribution predicted labels by each classifier

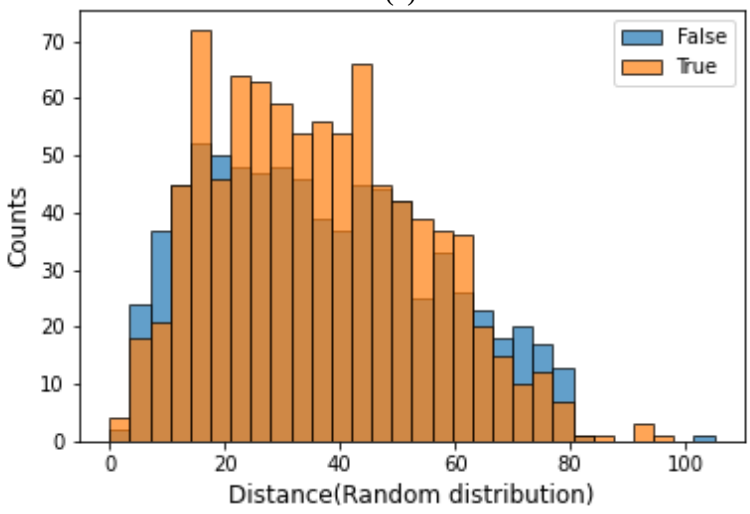
	ML classification algorithms					
Different Spatial Sampling	**RF**		**SVM**		**KNN**	
	Label	**Counts**	**Label**	**Counts**	**Label**	**Counts**
Equal Distribution	False	1111	False	1093	False	1163
	True	1057	True	1075	True	1005
Random Distribution	False	715	False	786	False	783
	True	959	True	888	True	891
Biased Distribution	False	583	False	598	False	582
	True	436	True	421	True	437

As it was expected, the highest accuracy has achieved by random distribution (70/30) in all three classifiers started from RF (0.59) and followed consecutively by KNN (0.58) and SVM (0.57). The classifiers showed similar behaviour for three spatial samplings except for SVM, random distribution (along the road) gained higher accuracy than equal and biased distribution. All three classifiers achieved the lowest accuracy in biased distribution.
The distance of True and False predicted data from the original training model is visualized in Figure 19.

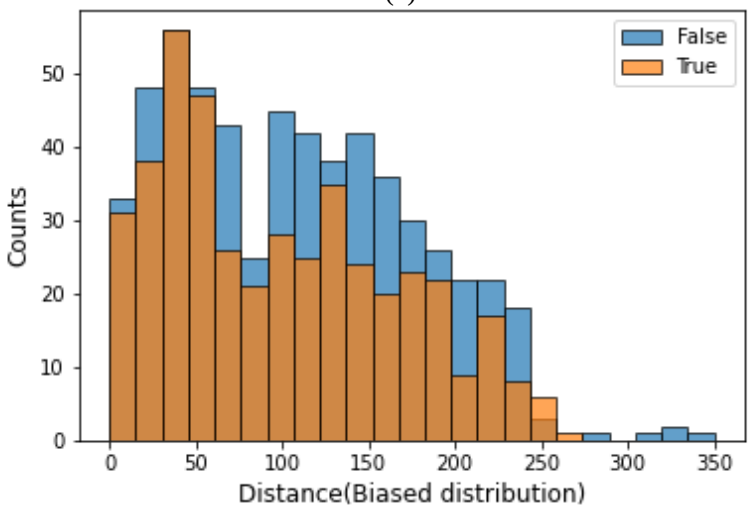
The aim here was evaluating the distance effect on the accuracy of classifiers (RF, KNN, SVM). After visualizing the results in figure 19, it is concluded that there is no definite pattern to confirm a correlation between correct and incorrect predicted labels based on distance from the original data.



(1)



(2)



(3)

Figure 19. Distance pattern between predicted labels (True, False) from original (trained) labels in 3 different distribution patterns (1,2,3)

5.3 Classification accuracy

Average weighted f1-score accuracy has been chosen among the other averaged accuracies such as precision and recall because some researchers such as (Pereira & Nunes, 2017) has claimed weighted f1-score is more sensible to imbalanced datasets. On the contrary, micro-average or accuracy is more appropriate for balanced datasets. Average weighted f1-score includes both positive and negative predicted labels per class which is an important factor for imbalanced dataset. Other researchers (Hu, 2017; Shahzad et al., 2015) have conducted the same evaluation parameter average weighted f1-Score and recall and precision and f1-score. The average weighted f1-score is the weighted average of recall and precision, and make the classifying process with an imbalanced dataset easier (Zhu et al., 2018).
The number of training samples per each class has manipulated the classifier's accuracy (Figure 20). The only exception is class "1: water" that it gained higher accuracy than (2: Pinus, 3: Betula, 8: Bare peat). The highest accuracy belongs to 6: Eriophorum with 1223 number of training data. The training samples per class is provided in Figure 12.

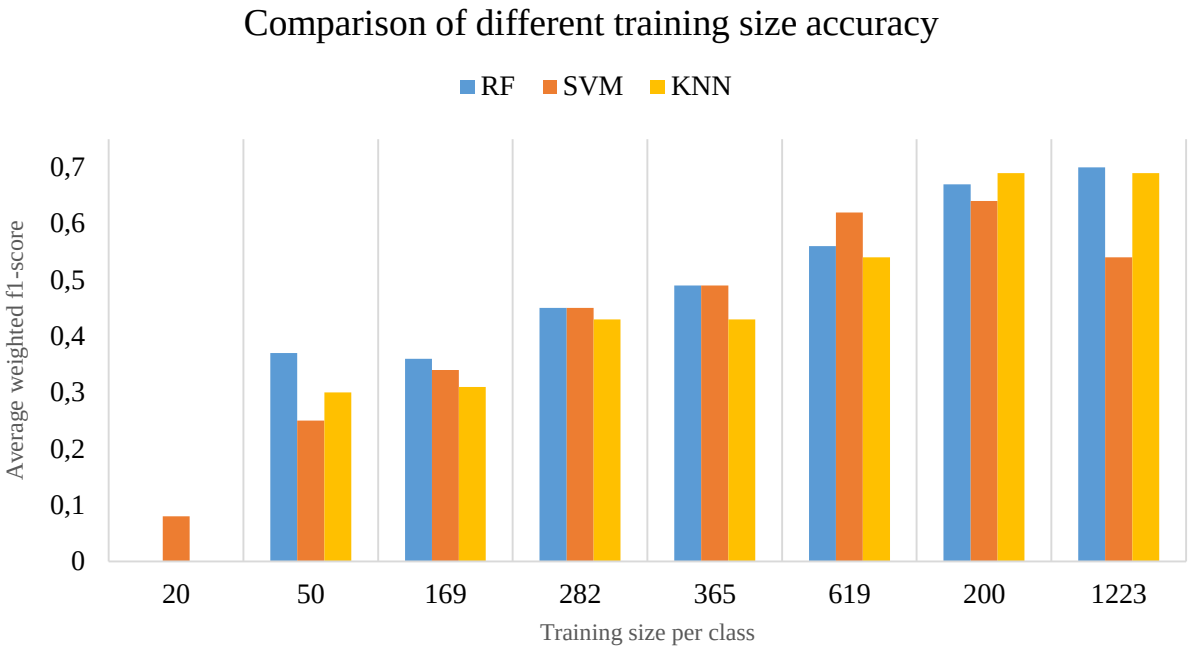


Figure 20. Comparison of three classifiers accuracy per class based on different training sizes

Several confusion matrices of each classifier based on test data by "*confusion_matrix Scikit-learn*" library have been conducted to analyze each classifier's performance by predicting labels per class. Altogether, three confusion matrices (tables 15,16,17) represent the confusion matrix of best models of each classifier (RF, SVM, KNN) results. Since all three classifiers behaved similarly for predicting labels, the confusion matrices results look similar for all three classifiers. "Class 6: Eriophorum" has the highest number of samples that have been predicted correctly by RF, SVM and KNN respectively (396, 418, 373) out of 525 test data of this class. SVM has the highest number of correctly predicted.
On the contrary, none of the classifiers could predict "class 4: Rhynchospora" correctly. The total number of class 4 samples are 28, which 8 of these samples were used as test data. That confirms an earlier discussion about the effect of the number of training samples. A class with the higher number of training samples ("Class 6: Eriophorum") have gained more correctly predicted labels in comparison to ("class 4: Rhynchospora") with the lower number of training samples.

Table 15. Confusion matrix of RF Hyperparameter classifier(121-pixels) with Vegetation indices parameter

Classes	1: Water	2: Pinus	3: Betula	4: Rhyncho-spora	5: Phragmit-es	6: Eriophor-um	7: Carex	8: Bare peat
1: Water	58	5	6	0	2	12	2	0
2: Pinus	8	52	48	0	1	11	0	1
3: Betula	3	30	89	0	6	28	0	1
4: Rhyncho-spora	0	0	2	0	0	5	1	0
5: Phragmit-es	2	14	8	0	21	25	3	0
6: Eriophorum	11	8	16	0	2	396	4	87
7: Carex	1	4	2	0	1	7	6	0
8: Bare peat	3	0	2	0	1	123	0	137

Table 16. Confusion Matrix of SVM default model with vegetation indices and MinMaxScaler (121pixel)

Classes	1: Water	2: Pinus	3: Betula	4: Rhyncho-spora	5: Phragmit-es	6: Eriophor-um	7: Carex	8: Bare peat
1: Water	54	6	4	0	1	18	2	0
2: Pinus	5	53	47	0	2	13	0	1
3: Betula	2	28	91	0	4	31	0	1
4: Rhyncho-spora	0	0	2	0	0	6	0	0
5: Phragmit-es	2	14	8	0	11	38	0	0
6: Eriophor-um	11	2	16	0	4	418	2	71
7: Carex	2	3	0	0	1	13	2	0
8: Bare peat	4	0	2	0	0	143	0	117

Table 17. Confusion matrix of KNN default model (121pixel) with Vegetation indices and MinMaxScaler

Classes	1: Water	2: Pinus	3: Betula	4: Rhyncho-spora	5: Phragmit-es	6: Eriophor-um	7: Carex	8: Bare peat
1: Water	59	8	4	0	3	7	1	2
2: Pinus	5	56	45	0	4	8	4	3
3: Betula	5	43	66	0	14	27	2	1
4: Rhyncho-spora	1	0	3	0	0	3	1	0
5: Phragmit-es	5	12	13	0	21	18	5	0
6: Eriophor-um	6	13	13	0	13	373	3	101
7: Carex	1	4	2	0	5	3	6	0
8: Bare peat	5	2	1	0	1	120	0	137

X. Zhang et al. (2017) has also confirmed the critical role of training sizes on the evaluation of classification performance by increasing the number of the sample from 1000 to 8000. The result of this change was a vast improvement in the classifier accuracy. After having the confusion matrices, TP, FP, TN, and TP are calculated and based on these information classification reports of all three classifiers are created (Table 18).

Table 18. Classification report of all three classifiers with best combination of ML concepts and features for each landcover class

	Random Forest				Support Vector Machine				K-Nearest Neighbours			
Classes/ Accuracy	**Precision**	**Recall**	**F1-Score**	**Support**	**Precision**	**Recall**	**F1-Score**	**Support**	**Precision**	**Recall**	**F1-Score**	**Support**
1: Water	0.67	0.68	0.67	85	0.68	0.64	0.65	85	0.69	0.68	0.69	87
2: Pinus	0.47	0.44	0.45	121	0.50	0.44	0.47	121	0.46	0.41	0.43	138
3: Betula	0.48	0.50	0.49	157	0.54	0.58	0.56	157	0.42	0.45	0.43	147
4: Rhyncosphora	0	0	0	8	0	0	0	8	0	0	0	0
5: Phragmites	0.47	0.29	0.36	73	0.48	0.15	0.23	73	0.29	0.34	0.31	61
6: Eriphorium	0.65	0.75	0.7	524	0.61	0.8	0.69	524	0.71	0.67	0.69	559
7: Carex	0.41	0.33	0.37	21	0.33	0.10	0.15	21	0.29	0.32	0.30	19
8: Bare Peat	0.60	0.52	0.56	266	0.62	0.44	0.51	266	0.52	0.56	0.54	244
Accuracy			0.60	1255			0.59	1255			0.57	1255
Macro AVG	0.47	0.44	0.45	1255	0.47	0.39	0.41	1255	0.42	0.43	0.42	1255
Weighted AVG	0.59	0.60	0.59	1255	0.58	0.59	0.57	1255	0.58	0.57	0.58	1255

All codes of this study are provided in 4 different workbooks in this link: https://github.com/MBarekaty/Thesis_Code-.git

Conclusion

This study's result brings out the accuracy of three ML classifiers (RF, KNN, SVM) for VHR UAV orthophoto classification with additional constructed features. Meanwhile, different ML concepts and feature combination has been implied in this study as well. Looking through the results, it is apparent that some features such as the Vegetation indices, MinMaxScaler have positively influenced the results, while Oversampling and Hyperparameter tuning parameters decreased the results considerably. Pixel generalization (1pixel, 25pixels, 121 pixels) reducing the spatial resolution has increased the accuracy through all classifiers with all feature combination apart from some exceptions.

The number of training samples is another factor influencing classification accuracy. Class 6 Eriophorum and Class 4: Rhynchospora has the highest and lowest training samples and has gained the highest and lowest accuracy per class, respectively. The results showed that random selection (70/30) would provide higher accuracy than three different spatial samplings. The distance of the train data from (True, False) classified data, did not indicate a concise pattern to conclude any specific effect from different distances between the train and test data on the accuracy results.

After considering all these parameters and features, ML classifiers algorithms' result indicated that the RF default model with Vegetation indices (GLI and VARI) had reached 0.59, and KNN default model with vegetation MinMaxScaler provided 0.58. Finally, SVM gained 0.57 with the combination of vegetation indices and MinMaxScaler. It is recommended to consider these indices in other studies with approximately the same classifiers parameters and datasets.

To achieve higher model accuracy, it is recommended to consider Multispectral calibrated images and OBIA segmentation techniques rather than reduce the spatial resolution.

Acknowledgements

I wish to express my deepest gratitude to my thesis supervisor Edgar Sepp for his unwavering support and profound belief in my work. At many stages of this work, I benefited from his brilliant advice and guidance. It would have been impossible to complete this thesis without his extensive knowledge. It was a great privilege and honour to work and study under his guidance. My appreciation also extends to my lecturers and committee in the University of Tartu, Geography department for their support and encouragement during this program.
I am extremely grateful to my parents for their love and sacrifices for educating and supporting me to reach my life goals. I cannot express how grateful I am for having them, without whom I would never have the chance to experience new things in my life.
Finally, but by no means least, my deepest heartfelt appreciation goes to my dear brother and amazing friends in Iran and Estonia, which were always there for me.

Masinõppe meetodite rakendamisel saadud taimkatte klassifikatsioonimudelite tulemuslikkuse hindamine kasutades kõrge ruumilise lahutusega UAV andmeid
Marjansadat Barekaty

Kokkuvõte

UAV ortofoto mosaiiki on kasutatud erinevate uurimisteemade juures, käesolevas töös kasutatakse seda uurimisala maakatte klassifitseerimiseks. UAV on kulutõhus seiretehnoloogia, kui uurimisala ei ole liiga suur. Primaarne maakatte klassifitseerimisel kasutatav andmestik on kõrge ruumilise lahutusega (VHR) UAV ortofoto mosaiik ja andmed maakatteklasside (8 erinevat klassi ja kokku 4148 näidist) asukohtadega..

Antud uurimustöö peamine eesmärk on võrrelda kolme enamkasutatud masinõppeklassifikaatori *Random Forest* (RF), *Support Vector Machine* (SVM) ja *K-nearest neighbours* (KNN) põhjal loodud maakatte klassifitseerimise mudelite täpsust. Nende mudelite loomisel kasutatakse järgmisi masinõppe kontseptsioone, nagu andmete skaleerimine (*MinMaxScaler* - KNNi ja SVMi korral), alaesindatud klasside võimendamist neid juhuslikult korduvalt valides (R*andom Oversampling*) ja mudeli parameetrite häälestamist koos täiendavate tunnuste lisamisega (vegetatsiooniindeksid): roheliste lehtede indeks (GLI) ja nähtav atmosfääritakistuse indeks (VARI).

Kõige suurem saavutatud täpsus oli 0,59, mis saadi RFi vaikemudeli ning vegetatsiooniindeksitega. KNNi ja SVMi vaikemudelite ning MinMaxScaleri ja vegetatsiooniindeksite kasutamisel saadi täpsuseks vastavalt 0,58 ja 0,57. Mudelite suhteliselt madala täpsuse põhjusteks on mitte tasakaalus olevad maakatteklasside andmed, nende omavaheline keeruline eristatavus ja VHR ortofoto müra.

ML-kontseptsioonide ja täiendavate lisatud tunnuste kõrval mõjutavad klassifitseerimise tulemusi veel mitmed erinevad tegurid. Ruumilise lahutuse vähendamine viis korda (5 x 5 = 25 piksli ühendamine üheks piksliks, kasutades kõigi ühendatud pikslite keskmist väärtust uue piksli väärtusena) ja üksteist korda (11 x 11 = 121) suurendas klassifikaatorimudeli täpsust. Pärast ruumilise lahutuse vähendamist on kõige mõttekam KNNi ja SVMi täpsuse suurendamiseks kasutada vegetatsiooniindekseid ja *MinMaxScalerit*. Täpsused viitasid sellele, et klassidel, millel oli teistest klassidest rohkem treeningandmeid , läks paremini. Kõige vähemtäpsed tulemused saadi nii vaike- kui mudeli parameetrite häälestamist kasutatavate mudelite korral teiste ML-kontseptsioonidega alaesindatud klasse juhuvalimiga ülevalides. Erinevalt teooriaosas mainitud uuringutulemustest ei parandanud juhuvalimiga ülevalik mudeli parameetrite häälestamine ühegi klassifikaatori tulemuslikkust.

Erinevad kasutatud ruumilised valikud maakatteklasside andmete jagamiseks õppe- ja valideerimisandmeteks ei andnud suuremat tõpsust juhusliku jaotusega (70/30) võrreldes. Üheks oluliseks asjaoluks ruumilise valiku halvemas täpsuses oli, et õpetusandmetes ei olnud tagatud kõigi maakatteklasside esinemine. Uurimistöös ei leidnud kinnitust seos et kaugus õpetusandmetest mängiks rolli tõpsuse juures..

Teistes uurimustes on leitud , et kalibreeritud multispektraalsed pildid on andnud täpsemad tulemused. Klassifitseerimise tõhustamiseks soovitatakse ruumilise resolutsiooni vähendamise asemel kasutada objektipõhist pildianalüüsi (OBIA).

Kokkuvõttes saavutati uurimistöös eespool kirjeldatud probleeme arvestades ja sarnaste maakattetüüpide korral ligikaudu täpsus 0,60.

References

Books

Breiman, L., & Cutler, A. (2005). *Random Forests. Berkeley*.

Raschka, S. (2018). *STAT 479: Machine Learning Lecture Notes*.

Journal articles, Conference paper, Thesis

Akbulut, Y., Sengur, A., Guo, Y., & Smarandache, F. (2017). NS-k-NN: Neutrosophic Set-Based k-Nearest Neighbors Classifier. *Symmetry*, *9*(9), 179. https://doi.org/10.3390/sym9090179

Ataei, M., & Osanloo, M. (2004). Using a combination of genetic algorithm and the grid search method to determine optimum cutoff grades of multiple metal deposits. *International Journal of Surface Mining, Reclamation and Environment*, *18*(1), 60–78. https://doi.org/10.1076/ijsm.18.1.60.23543

Bannari, A., Morin, D., Bonn, F., & Huete, A. R. (1995). A review of vegetation indices. *Remote Sensing Reviews*, *13*(1–2), 95–120. https://doi.org/10.1080/02757259509532298

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, *6*(1), 20–29. https://doi.org/10.1145/1007730.1007735

Bendig, J., Yu, K., Aasen, H., Bolten, A., Bennertz, S., Broscheit, J., Gnyp, M. L., & Bareth, G. (2015). Combining UAV-based plant height from crop surface models, visible, and near infrared vegetation indices for biomass monitoring in barley. *International Journal of Applied Earth Observation and Geoinformation*, *39*, 79–87. https://doi.org/10.1016/j.jag.2015.02.012

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, *13*(1), 281–305.

Böhler, J. E., Schaepman, M. E., & Kneubühler, M. (2018). Crop Classification in a Heterogeneous Arable Landscape Using Uncalibrated UAV Data. *Remote Sensing*, *10*(8), 1282. https://doi.org/10.3390/rs10081282

Borkin, D., Nemethova, A., Michalconok, G., & Maiorov, K. (2019). Impact of Data Normalization on Classification Model Accuracy. *Research Papers Faculty of Materials Science and Technology Slovak University of Technology*, *27*, 79–84. https://doi.org/10.2478/rput-2019-0029

Bosch, A., Zisserman, A., & Munoz, X. (2007). Image classification using random forests and ferns. *2007 IEEE 11th International Conference on Computer Vision*, 1–8. https://doi.org/10.1109/ICCV.2007.4409066

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Bremner, D., Demaine, E., Erickson, J., Iacono, J., Langerman, S., Morin, P., & Toussaint, G. (2005). Output-sensitive algorithms for computing nearest-neighbour decision boundaries. *Discrete and Computational Geometry*, *33*(4), 593–604. https://doi.org/10.1007/s00454-004-1152-0

Bunke, H., & Riesen, K. (2012). Towards the unification of structural and statistical pattern recognition. *Pattern Recognition Letters*, *33*(7), 811–825. https://doi.org/10.1016/j.patrec.2011.04.017

Castellanos, F. J., Valero-Mas, J. J., Calvo-Zaragoza, J., & Rico-Juan, J. R. (2018). Oversampling imbalanced data in the string space. *Pattern Recognition Letters*, *103*, 32–38. https://doi.org/10.1016/j.patrec.2018.01.003

Claesen, M., Simm, J., Popovic, D., & Moor, B. D. (2014). Hyperparameter tuning in Python using Optunity. *Proceedings of the International Workshop on Technical Computing for Machine Learning and Mathematical Engineering*, *1*, 3.

Colditz, R. R. (2015). An Evaluation of Different Training Sample Allocation Schemes for Discrete and Continuous Land Cover Classification Using Decision Tree-Based Algorithms. *Remote Sensing*, *7*(8), 9655–9681. https://doi.org/10.3390/rs70809655

Cortes, C., & Vapnik, V. (1995). Support-vector networks. 20. *Mach Learn*, *20*. https://doi.org/10.1007/BF00994018

Cover, T. M., & Hart, P. E. (1967). Nearest neighbour pattern classification. *IEEE Trans. Inf. Theory*. https://doi.org/10.1109/TIT.1967.1053964

Crammer, K., & Singer, Y. (2001). On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, *2*(Dec), 265–292.

Drummond, C., & Holte, R. C. (2003). *C4.5, Class Imbalance, and Cost Sensitivity: Why Under-sampling beats Over-sampling*. 1–8. https://doi.org/10.1145/1089827.1089843

Du, P., Samat, A., Waske, B., Liu, S., & Li, Z. (2015). Random Forest and Rotation Forest for fully polarized SAR image classification using polarimetric and spatial features. *ISPRS Journal of Photogrammetry and Remote Sensing*, *105*, 38–53. https://doi.org/10.1016/j.isprsjprs.2015.03.002

Duro, D. C., Franklin, S. E., & Dubé, M. G. (2012). A comparison of pixel-based and object-based image analysis with selected machine learning algorithms for the classification of agricultural landscapes using SPOT-5 HRG imagery. *Remote Sensing of Environment*, *118*, 259–272. https://doi.org/10.1016/j.rse.2011.11.020

El Garouani, A., Alobeid, A., & El Garouani, S. (2014). Digital surface model based on aerial image stereo pairs for 3D building. *International Journal of Sustainable Built Environment*, *3*(1), 119–126. https://doi.org/10.1016/j.ijsbe.2014.06.004

Feng, Q., Liu, J., & Gong, J. (2015). UAV Remote Sensing for Urban Vegetation Mapping Using Random Forest and Texture Analysis. *Remote Sensing*, *7*(1), 1074–1094. https://doi.org/10.3390/rs70101074

Foody, G. M., & Mathur, A. (2004). Toward intelligent training of supervised image classifications: Directing training data acquisition for SVM classification. *Remote Sensing of Environment*, *93*(1), 107–117. https://doi.org/10.1016/j.rse.2004.06.017

Franco-Lopez, H., Ek, A. R., & Bauer, M. E. (2001). Estimation and mapping of forest stand density, volume, and cover type using the k-nearest neighbors method. *Remote Sensing of Environment*, *77*(3), 251–274. https://doi.org/10.1016/S0034-4257(01)00209-7

Friedrichs, F. (2004). C. igel. Evolutionary tuning of multiple svm parameters. *Proceedings of the 12th European Symposium on Artificial Neural Networks*.

Ganganwar, V. (2012). An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, *2*, 42–47.

Garcia-Ruiz, F. J., Wulfsohn, D., & Rasmussen, J. (2015). Sugar beet (Beta vulgaris L.) and thistle (Cirsium arvensis L.) discrimination based on field spectral data. *Biosystems Engineering*, *139*, 1–15. https://doi.org/10.1016/j.biosystemseng.2015.07.012

Ghosh, A., & Joshi, P. K. (2014). A comparison of selected classification algorithms for mapping bamboo patches in lower Gangetic plains using very high resolution WorldView 2 imagery. *International Journal of Applied Earth Observation and Geoinformation*, *26*, 298–311. https://doi.org/10.1016/j.jag.2013.08.011

Guerrero, J. M., Pajares, G., Montalvo, M., Romeo, J., & Guijarro, M. (2012). Support Vector Machines for crop/weeds identification in maize fields. *Expert Systems with Applications*, *39*(12), 11149–11155. https://doi.org/10.1016/j.eswa.2012.03.040

Guo, W., Rage, U. K., & Ninomiya, S. (2013). Illumination invariant segmentation of vegetation for time series wheat images based on decision tree model. *Computers and Electronics in Agriculture*, *96*, 58–66. https://doi.org/10.1016/j.compag.2013.04.010

Hale, J. (2020, February 21). *Scale, Standardize, or Normalize with Scikit-Learn*. Medium. https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02

Hall, O., Dahlin, S., Marstorp, H., Archila Bustos, M. F., Öborn, I., & Jirström, M. (2018). Classification of Maize in Complex Smallholder Farming Systems Using UAV Imagery. *Drones*, *2*(3), 22. https://doi.org/10.3390/drones2030022

Hamuda, E., Glavin, M., & Jones, E. (2016). A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture*, *125*, 184–199. https://doi.org/10.1016/j.compag.2016.04.024

Heumann, B. W. (2011). An Object-Based Classification of Mangroves Using a Hybrid Decision Tree—Support Vector Machine Approach. *Remote Sensing*, *3*(11), 2440–2460. https://doi.org/10.3390/rs3112440

Ho, T. K. (1995). Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, *1*, 278–282. https://doi.org/10.1109/ICDAR.1995.598994

Horning, N. (2010). Random Forests: An algorithm for image classification and generation of continuous fields data sets. *Proceedings of the International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences, Osaka, Japan*, *911*.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). *A practical guide to support vector classification*. Taipei.

Hu, J. (2017). Automated Detection of Driver Fatigue Based on AdaBoost Classifier with EEG Signals. *Frontiers in Computational Neuroscience*, *11*. https://doi.org/10.3389/fncom.2017.00072

Huang, C., Davis, L. S., & Townshend, J. R. G. (2002). An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, *23*(4), 725–749.

Hung, C., Xu, Z., & Sukkarieh, S. (2014). Feature Learning Based Approach for Weed Classification Using High Resolution Aerial Images from a Digital Camera Mounted on a UAV. *Remote Sensing*, *6*(12), 12037–12054. https://doi.org/10.3390/rs61212037

Hunt, E. R., Cavigelli, M., Daughtry, C. S. T., Mcmurtrey, J. E., & Walthall, C. L. (2005). Evaluation of Digital Photography from Model Aircraft for Remote Sensing of Crop Biomass and Nitrogen Status. *Precision Agriculture*, *6*(4), 359–378. https://doi.org/10.1007/s11119-005-2324-5

Im, J., Jensen, J. R., & Tullis, J. A. (2008). Object-based change detection using correlation image analysis and image segmentation. *International Journal of Remote Sensing*, *29*(2), 399–423. https://doi.org/10.1080/01431160601075582

Jannoura, R., Brinkmann, K., Uteau, D., Bruns, C., & Joergensen, R. G. (2015). Monitoring of crop biomass using true colour aerial photographs taken from a remote controlled hexacopter. *Biosystems Engineering*, *Complete*(129), 341–351. https://doi.org/10.1016/j.biosystemseng.2014.11.007

Japkowicz, N. (2000). The Class Imbalance Problem: Significance and Strategies. *In Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI*, 111–117.

Jelínek, Z., Mašek, J., Starý, K., Lukáš, J., & Kumhálová, J. (2020). *Winter wheat, winter rape and poppy crop growth evaluation with the help of remote and proximal sensing measurements*. https://doi.org/10.15159/ar.20.176

Jónsson, S. (2019). RGB and Multispectral UAV image classification of agricultural fields using a machine learning algorithm. *Student Thesis Series INES*. http://lup.lub.lu.se/student-papers/record/8971989

Kelcey, J., & Lucieer, A. (2012). Sensor Correction of a 6-Band Multispectral Imaging Sensor for UAV Remote Sensing. *Remote Sensing*, *4*(5), 1462–1493. https://doi.org/10.3390/rs4051462

Khan, L., Awad, M., & Thuraisingham, B. (2007). A new intrusion detection system using support vector machines and hierarchical clustering. *The VLDB Journal*, *16*(4), 507–521. https://doi.org/10.1007/s00778-006-0002-5

Khatami, R., Mountrakis, G., & Stehman, S. V. (2016). A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: General guidelines for practitioners and future research. *Remote Sensing of Environment*, *177*, 89–100. https://doi.org/10.1016/j.rse.2016.02.028

Knorn, J., Rabe, A., Radeloff, V. C., Kuemmerle, T., Kozak, J., & Hostert, P. (2009). Land cover mapping of large areas using chain classification of neighboring Landsat satellite images. *Remote Sensing of Environment*, *113*(5), 957–964. https://doi.org/10.1016/j.rse.2009.01.010

Li, Manchun, Ma, L., Blaschke, T., Cheng, L., & Tiede, D. (2016). A systematic comparison of different object-based classification techniques using high spatial resolution imagery in agricultural environments. *International Journal of Applied Earth Observation and Geoinformation*, *49*, 87–98. https://doi.org/10.1016/j.jag.2016.01.011

Li, Miao, Zang, S., Zhang, B., Li, S., & Wu, C. (2014). A Review of Remote Sensing Image Classification Techniques: The Role of Spatio-contextual Information. *European Journal of Remote Sensing*, *47*(1), 389–411. https://doi.org/10.5721/EuJRS20144723

Liaw, A., & Wiener, M. (2001). Classification and Regression by RandomForest. *Forest*, *23*.

Ling, C. X., & Li, C. (n.d.). *Data Mining for Direct Marketing: Problems and Solutions*. 7.

Lottes, P., Khanna, R., Pfeifer, J., Siegwart, R., & Stachniss, C. (2017). UAV-based crop and weed classification for smart farming. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 3024–3031. https://doi.org/10.1109/ICRA.2017.7989347

Louhaichi, M., Borman, M., & Johnson, D. (2001). Spatially Located Platform and Aerial Photography for Documentation of Grazing Impacts on Wheat. *Geocarto International*, *16*. https://doi.org/10.1080/10106040108542184

Lussem, U., Bolten, A., Gnyp, M., Jasper, J., & Bareth, g. (2018). evaluation of rgb-based vegetation indices from uav imagery to estimate forage yield in grassland. *isprs - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII–3*, 1215–1219. https://doi.org/10.5194/isprs-archives-XLII-3-1215-2018

Ma, L., Cheng, L., Li, M., Liu, Y., & Ma, X. (2015). Training set size, scale, and features in Geographic Object-Based Image Analysis of very high resolution unmanned aerial vehicle imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, *102*, 14–27. https://doi.org/10.1016/j.isprsjprs.2014.12.026

Ma, L., Fu, T., Blaschke, T., Li, M., Tiede, D., Zhou, Z., Ma, X., & Chen, D. (2017). Evaluation of Feature Selection Methods for Object-Based Land Cover Mapping of Unmanned Aerial Vehicle Imagery Using Random Forest and Support Vector Machine Classifiers. *ISPRS International Journal of Geo-Information*, *6*(2), 51. https://doi.org/10.3390/ijgi6020051

Mantovani, R. G., Rossi, A. L. D., Vanschoren, J., Bischl, B., & Carvalho, A. C. P. L. F. (2015). To tune or not to tune: Recommending when to adjust SVM hyper-parameters

via meta-learning. *2015 International Joint Conference on Neural Networks (IJCNN)*, 1–8. https://doi.org/10.1109/IJCNN.2015.7280644

Melville, B., Fisher, A., & Lucieer, A. (2019). Ultra-high spatial resolution fractional vegetation cover from unmanned aerial multispectral imagery. *International Journal of Applied Earth Observation and Geoinformation*, *78*, 14–24. https://doi.org/10.1016/j.jag.2019.01.013

Mercier, G., & Lennon, M. (2003). Support vector machines for hyperspectral image classification with spectral-based kernels. *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477)*, *1*, 288–290. https://doi.org/10.1109/IGARSS.2003.1293752

Millard, K., & Richardson, M. (2015). On the Importance of Training Data Sample Selection in Random Forest Image Classification: A Case Study in Peatland Ecosystem Mapping. *Remote Sensing*, *7*(7), 8489–8515. https://doi.org/10.3390/rs70708489

Mitchell, T. (1997). Introduction to machine learning. *Machine Learning*, *7*, 2–5.

Mori, S., Suen, C. Y., & Yamamoto, K. (1992). Historical review of OCR research and development. *Proceedings of the IEEE*, *80*(7), 1029–1058. https://doi.org/10.1109/5.156468

Ozesmi, S. L., & Bauer, M. E. (2002). Satellite remote sensing of wetlands. *Wetlands Ecology and Management*, *10*(5), 381–402. https://doi.org/10.1023/A:1020908432489

Pajares, G. (2015). Overview and Current Status of Remote Sensing Applications Based on Unmanned Aerial Vehicles (UAVs). *Photogrammetric Engineering & Remote Sensing*, *81*(4), 281–330. https://doi.org/10.14358/PERS.81.4.281

Pal, M., & Mather, P. M. (2005). Support vector machines for classification in remote sensing. *International Journal of Remote Sensing*, *26*(5), 1007–1011. https://doi.org/10.1080/01431160512331314083

Pan, Y., Zhang, X., Sun, M., & Zhao, Q. (2017). object-based and supervised detection of potholes and cracks from the pavement images acquired by UAV. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, *42*. https://doi.org/10.5194/isprs-archives-XLII-4-W4-209-2017

Pereira, L., & Nunes, N. (2017). A comparison of performance metrics for event classification in Non-Intrusive Load Monitoring. *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 159–164. https://doi.org/10.1109/SmartGridComm.2017.8340682

Pérez-Ortiz, M., Peña, J. M., Gutiérrez, P. A., Torres-Sánchez, J., Hervás-Martínez, C., & López-Granados, F. (2015). A semi-supervised system for weed mapping in sunflower crops using unmanned aerial vehicles and a crop row detection method. *Applied Soft Computing*, *37*, 533–544. https://doi.org/10.1016/j.asoc.2015.08.027

Pérez-Ortiz, María, Peña, J. M., Gutiérrez, P. A., Torres-Sánchez, J., Hervás-Martínez, C., & López-Granados, F. (2016). Selecting patterns and features for between- and within-crop-row weed mapping using UAV-imagery. *Expert Systems with Applications*, *47*, 85–94. https://doi.org/10.1016/j.eswa.2015.10.043

Poblete-Echeverría, C., Olmedo, G. F., Ingram, B., & Bardeen, M. (2017). Detection and Segmentation of Vine Canopy in Ultra-High Spatial Resolution RGB Imagery Obtained from Unmanned Aerial Vehicle (UAV): A Case Study in a Commercial Vineyard. *Remote Sensing*, *9*(3), 268. https://doi.org/10.3390/rs9030268

Preuveneers, D., Tsingenopoulos, I., & Joosen, W. (2020). Resource usage and performance trade-offs for machine learning models in smart environments. *Sensors*, *20*(4), 1176. https://doi.org/10.3390/s20041176

Probst, P., Bischl, B., & Boulesteix, A.-L. (2018). Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *ArXiv:1802.09596 [Stat]*. http://arxiv.org/abs/1802.09596

Qian, Y., Zhou, W., Yan, J., Li, W., & Han, L. (2015). Comparing Machine Learning Classifiers for Object-Based Land Cover Classification Using Very High Resolution Imagery. *Remote Sensing*, *7*(1), 153–168. https://doi.org/10.3390/rs70100153

Raeder, T., Forman, G., & Chawla, N. V. (2012). Learning from Imbalanced Data: Evaluation Matters. In D. E. Holmes & L. C. Jain (Eds.), *Data Mining: Foundations and Intelligent Paradigms: Volume 1: Clustering, Association and Classification* (pp. 315–331). Springer. https://doi.org/10.1007/978-3-642-23166-7_12

Raju, V. N. G., Lakshmi, K. P., Jain, V. M., Kalidindi, A., & Padma, V. (2020). Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification. *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 729–735. https://doi.org/10.1109/ICSSIT48917.2020.9214160

RAMADHAN, M. M., SITANGGANG, I. S., NASUTION, F. R., & GHIFARI, A. (2017). Parameter Tuning in Random Forest Based on Grid Search Method for Gender Classification Based on Voice Frequency. *DEStech Transactions on Computer Science and Engineering*, *cece*. https://doi.org/10.12783/dtcse/cece2017/14611

Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-Validation. In L. LIU & M. T. ÖZSU (Eds.), *Encyclopedia of Database Systems* (pp. 532–538). Springer US. https://doi.org/10.1007/978-0-387-39940-9_565

Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., & Rigol-Sanchez, J. P. (2012). An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *67*, 93–104. https://doi.org/10.1016/j.isprsjprs.2011.11.002

Shahzad, R. K., Fatima, M., Lavesson, N., & Boldt, M. (2015). Consensus decision making in random forests. *International Workshop on Machine Learning, Optimization and Big Data*, 347–358.

Shi, D., & Yang, X. (2015). Support Vector Machines for Land Cover Mapping from Remote Sensor Imagery. In J. Li & X. Yang (Eds.), *Monitoring and Modeling of Global Changes: A Geomatics Perspective* (pp. 265–279). Springer Netherlands. https://doi.org/10.1007/978-94-017-9813-6_13

Song, Q., Xiang, M., Hovis, C., Zhou, Q., Lu, M., Tang, H., & Wu, W. (2019). Object-based feature selection for crop classification using multi-temporal high-resolution imagery. *International Journal of Remote Sensing*, *40*(5–6), 2053–2068. https://doi.org/10.1080/01431161.2018.1475779

Thanh Noi, P., & Kappas, M. (2018). Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. *Sensors*, *18*(1), 18. https://doi.org/10.3390/s18010018

Torres-Sánchez, J., López-Granados, F., & Peña, J. M. (2015). An automatic object-based method for optimal thresholding in UAV images: Application for vegetation detection in herbaceous crops. *Computers and Electronics in Agriculture*, *114*, 43–52. https://doi.org/10.1016/j.compag.2015.03.019

Tzotsos, A., & Argialas, D. (2008). Support Vector Machine Classification for Object-Based Image Analysis. In *Object-Based Image Analysis* (pp. 663–677). https://doi.org/10.1007/978-3-540-77058-9_36

Varma, M. K. S., Rao, N. K. K., Raju, K. K., & Varma, G. P. S. (2016). Pixel-Based Classification Using Support Vector Machine Classifier. *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, 51–55. https://doi.org/10.1109/IACC.2016.20

Watts, J. D., Powell, S. L., Lawrence, R. L., & Hilker, T. (2011). Improved classification of conservation tillage adoption using high temporal and synthetic satellite imagery. *Remote Sensing of Environment*, *115*(1), 66–75. https://doi.org/10.1016/j.rse.2010.08.005

Wei, C., Huang, J., Mansaray, L. R., Li, Z., Liu, W., & Han, J. (2017). Estimation and Mapping of Winter Oilseed Rape LAI from High Spatial Resolution Satellite Data Based on a Hybrid Method. *Remote Sensing*, *9*(5), 488. https://doi.org/10.3390/rs9050488

Wenwen, L., Xiaoxue, X., Fu, L., & Yu, Z. (2014). Application of improved grid search algorithm on SVM for classification of tumor gene. *International Journal of Multimedia and Ubiquitous Engineering*, *9*(11), 181–188. http://dx.doi.org/10.14257/ijmue.2014.9.11.18

Xie, F., Li, F., Lei, C., & Ke, L. (2018). Representative Band Selection for Hyperspectral Image Classification. *ISPRS International Journal of Geo-Information*, *7*(9), 338. https://doi.org/10.3390/ijgi7090338

Ying, X. (2019). An Overview of Overfitting and its Solutions. *Journal of Physics: Conference Series*, *1168*, 022022. https://doi.org/10.1088/1742-6596/1168/2/022022

Young, N. E., Anderson, R. S., Chignell, S. M., Vorster, A. G., Lawrence, R., & Evangelista, P. H. (2017). A survival guide to Landsat preprocessing. *Ecology*, *98*(4), 920–932. https://doi.org/10.1002/ecy.1730

Yuan, Y., Lin, J., & Wang, Q. (2016). Hyperspectral Image Classification via Multitask Joint Sparse Representation and Stepwise MRF Optimization. *IEEE Transactions on Cybernetics*, *46*(12), 2966–2977. https://doi.org/10.1109/TCYB.2015.2484324

Zhang, H. K., & Roy, D. P. (2017). Using the 500m MODIS land cover product to derive a consistent continental scale 30m Landsat land cover classification. *Remote Sensing of Environment*, *197*, 15–34. https://doi.org/10.1016/j.rse.2017.05.024

Zhang, X., Chen, G., Wang, W., Wang, Q., & Dai, F. (2017). Object-Based Land-Cover Supervised Classification for Very-High-Resolution UAV Images Using Stacked Denoising Autoencoders. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(7), 3373–3385. https://doi.org/10.1109/JSTARS.2017.2672736

Zheng, Z., Cai, Y., & Li, Y. (2016). Oversampling method for imbalanced classification. *Computing and Informatics*, *34*(5), 1017–1037.

Zhu, M., Xia, J., Jin, X., Yan, M., Cai, G., Yan, J., & Ning, G. (2018). Class Weights Random Forest Algorithm for Processing Class Imbalanced Medical Data. *IEEE Access*, *6*, 4641–4652. https://doi.org/10.1109/ACCESS.2018.2789428

Web Reference

3.2.4.3.1. Sklearn.ensemble.RandomForestClassifier—Scikit-learn 0.23.2 documentation. (n.d.). Retrieved November 1, 2020, from https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Brownlee, J. (2016, November 17). What is a Confusion Matrix in Machine Learning. *Machine Learning Mastery*. https://machinelearningmastery.com/confusion-matrix-machine-learning/

Dataman, D. (2020, July 20). *Using Over-Sampling Techniques for Extremely Imbalanced Data*. Medium. https://towardsdatascience.com/sampling-techniques-for-extremely-imbalanced-data-part-ii-over-sampling-d61b43bc4879

Dorpe, S. V. (2018, December 13). *Preprocessing with sklearn: A complete and comprehensive guide*. Medium. https://towardsdatascience.com/preprocessing-with-sklearn-a-complete-and-comprehensive-guide-670cb98fcfb9

Dwivedi, D. (2019, March 27). *Machine Learning For Beginners*. Medium. https://towardsdatascience.com/machine-learning-for-beginners-d247a9420dab

Feature (machine learning). (2020). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Feature_(machine_learning)&oldid=960189864

Feature Scaling | Standardization Vs Normalization. (2020, April 3). *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/

KNN Classification. (2020, July 9). https://www.saedsayad.com/k_nearest_neighbors.htm

Mohajon, J. (2020, September 9). *Confusion Matrix for Your Multi-Class Machine Learning Model*. Medium. https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826

Narkhede, S. (2019, August 29). *Understanding Confusion Matrix*. Medium. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Pant, A. (2019, January 23). *Workflow of a Machine Learning Project*. Medium. https://towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94

Sharma, P. (2019, August 25). *Why is scaling required in KNN and K-Means?* Medium. https://medium.com/analytics-vidhya/why-is-scaling-required-in-knn-and-k-means-8129e4d88ed7

Sklearn.model_selection.GridSearchCV — scikit-learn 0.23.2 documentation. (n.d.). Retrieved November 1, 2020, from https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

sklearn.neighbors.KNeighborsClassifier—Scikit-learn 0.23.2 documentation. (n.d.). Retrieved November 2, 2020, from https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

Sklearn.svm.SVC — scikit-learn 0.23.2 documentation. (n.d.). Retrieved November 2, 2020, from https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

Vickery, R. (2020, April 16). 6 Machine Learning Concepts for Beginners. Medium. https://towardsdatascience

Non-exclusive licence to reproduce thesis and make thesis public

I, Marjansadat Barekaty,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

"Compare the performance of applying Machine Learning concepts to landcover classification models using very high-resolution UAV data", supervised by Edgar Sepp,

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Marjansadat Barekaty

Tartu,18/01/2021