Machine Learning Algorithms in Combination with Geographically Weighted Variables

Rick Venema

2018-11-16

Abstract

This paper extends the paper by Johnson et. al. 2012 [1], further examining different Machine Learning algorithms for ASTER image classifiction of forest types in a Japanese forest containing Sugi (*Cryptomeria japonica*) and Hinoki (*Chamaecyparis obtusa*) trees. This paper first looks into the dataset that was used in the original paper and tests different Machine Learning algorithms to create a model that can classify the different tree types quick and accurate. This has been done by using Weka, and the standard settings. Exept Support Vector Machines (SVMs), which has been set up with the settings used in the original paper.

Introduction

The use of machine learning in classifying different forest types has been a subject of research for a long time, however, no universal standard classification algorithm has been defined to classify forest types. An universal model can be quite difficult to create, this is because of the fact that a lot of different datacollecting methods exists and give different results. In the original paper by Johnson et al. [1], SVM was performed, however, other algorithms such as Logistic regression (LR), MLP, or C4.5 [6]. To determine the best classification algorithm, different algorithms needs to be tested on the data, with different settings. An unified model that can classify all the different forest types needs more refining and more data, such as soil texture, and is not something that can be done easily. A solution to the problem of an unified model, is creating different models that are optimized for the different forest types that exists. This however comes down to the same problem that exists today.

Classification of forest types can give a setup to a lot of different possibilities in agriculture and other geographical research areas. In agriculture the use of drones has increased significantly over the years, and is an active field of research where Machine Learning can have a lot of different uses, and can be improving the crop quality when used correctly.

However, before this all this is possible, there needs to be more research of the classification process. This study looks at the classificiation of forest types with different algorithms, and the general question that has been asked is, is what difference is there between different machine learnign algorithms in the classification of forest types, taking geographical weighted variables into account?

Materials and Methods

In this study the Forest type classfication set was used[2]. This dataset has 27 different columns, the first 9 are the measured spectral values of the images, the next 18 different columns are the Interpolated values of the different forest type, based on Moran's I. Which is defined as:

$$I = \frac{N}{W} \frac{\sum_i \sum_j w_{ij}(x_i - \bar{x}(x_j - \bar{x}))}{\sum_i (x_i - \bar{x})^2} \tag{1}$$

This interpolation was already done to the dataset that was used. Thus appending the measured spectral values with the 18 columns with the calculated interpolated values.

To determine which algorithm performs the best on the dataset used, different algorithms were used in Weka. The most standaard algorithms were used, all with the standard settings: ZeroR, OneR, IBk, C4.5, Logistic Regression, Mulitlayer Perceptron, Random Forest, Simple Logistic, and Support Vector Machine. Multilayer Perceptron was however chosen because of the fact that is a common algorithm used in the Geographical Information Science (GIS) community[6]. This algorithm performs better on multiclass classification problems, and can thus perform better or equal to svm[8]. SVM has been used in the classification of forest types by Johnson et al. 2012[1].

For the Exploratory data analysis different programs were used. Most of this was done in R in combination with R studio. A hierarchial cluster plot was made using the `hclust` method which is in the standard library of R. The Euclidian distance was used to calculate the distances, which is given by the next formula:

$$||a - b||_2 = \sqrt{\sum_i (a_i - b_i)^2} \tag{2}$$

The PCA was made with the use of the `ggplot` library which increases the readability of the plots by making them more clear with better backgrounds and better title and caption options. A PCA plot can give an insight at the different groups that can be made, and can give new and better insights at how the data can be clustered over multiple dimensions.

For batch running different algorithms the weka experimenter was used to compare the different algorithms easy and give clear insights which algorithms perform best. The experimenter used, was from Weka 3.8 which runs on Java. For the C4.5, J48 was used. This is due to the fact that J48 is a Java implementation of the C4.5 algorithm.

To create the roc curve, the knowledge flow of weka 3.8 was used to compare the roc curves of the algorithms that performed best. This knowledge flow can be setup via weka and has a gui to execute the different algorithms next to each other in parallel.

Results

The determination of the best algorithm to use on Spectral Values with geographically weighted variables has been done using an EDA first, then different algorithms were tested on speed and accuracy. After selecting the best 2 algorithms, these were examined further to determine the best algorithm for the dataset used in this research.

Exploratory Data Analysis

Layout of the data

After first downloading the raw dataset, a first impression of the data was needed. When trying to understand the data, the data needs to be examined to determine what the best strategy is for further actions. The results of this Exploratory Data Analysis (EDA) exist to get an impression on what the dataset is looking like. First the size of the dataset was determined. The dimensions are 9 attributes representing the measured spectral values, 18 attributes representing IDW interpolated values, these are further discussed later on, and 1 class attribute. The class attribute can be given 4 different values, the first two being "*H*", which stands for Hinoki *(Chamaecyparis obtusa)*, and "*S*", which stands for Sugi (*Cryptomeria japonica*). These 2 class labels represent the two different types of trees found in the forest that was used to create the dataset. Furthermore, there are 2 more labels that the class attribute can get. These 2 labels represent the Mixed label, displayed as "*D*", and the "*O*" label. These labels represent a mixed part and a non forest label respectively.

Descriptive Statistics

In table 1, the distribution of the class label can be seen. This table represents how the different labels are dispersed in the data set that was used. As we can see in the table, the distribution of the different labels is uneven, meaning that some classes are represented more than others. The Other label is the least frequent label, with 37 instances. What stands out, is the mixed label. This label is as frequent as the only one tree type labels, it be with a little difference.

Table 1: Distribution of the different labels of the Class attribute.

Class label	Amount
Hinoki	48
Sugi	59
Other	37
Mixed	54

The dataset was searched for any missing values, but because of the fact that the dataset is made up by measured spectral values of an ASTER image set. Meaning if there were any NA values, the measurement went wrong. Without any missing values, there is no need to remove any instances.

Clustering of the data

To get a view of the dispersion of the data, a hierachial clustering was performed. This gives a good view on how the data is layout, if the data is clustered correctly, the algorihtm will have more chance of being accurate, and thus performing much better. In this case, the data is clustered mostly correct, with just a few different misclustered datapoints. These are most frequent in the "Other" and "Mixed" class, thus representing a significant simularity in these two class labels. The cluster dendogram can be viewed in figure 1.

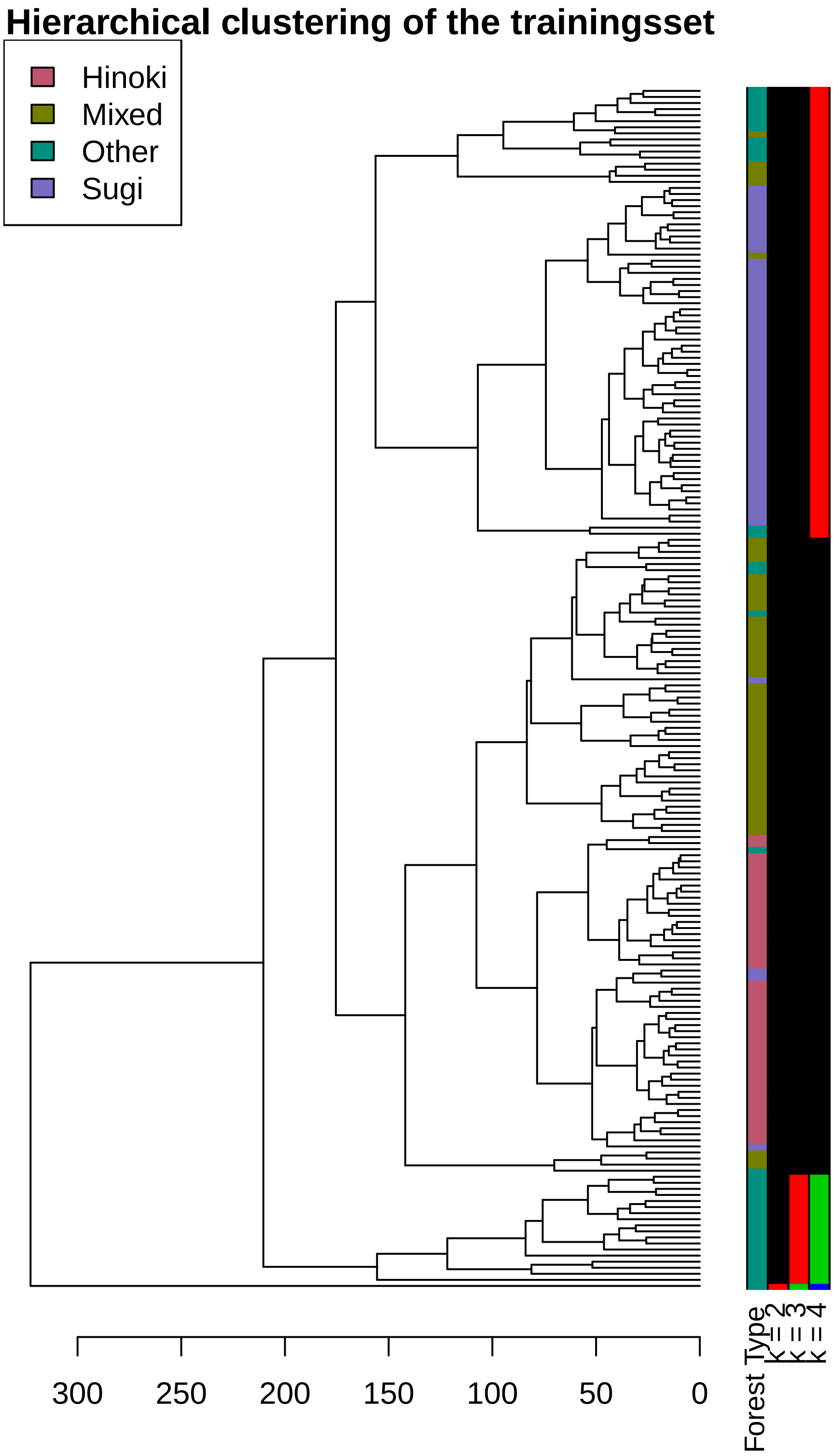


Figure 1: Hierarchical clustering of the trainings data used to train ML models. The different colors represent different class labels.

PCA

To get see how the data is dispersed, a PCA was performed to reduce the dimensionality of the dataset. The three different dates were clustered together to see if there was a significant difference in dates. First a a summary of the data is shown in table 2. This gives a first impresion on the dispersion of the data. In this case significant differences can be seen, the data from 26 September 2010 are much lower, exept for the max value, meaning that the datapoints are mostly lower than the other dates, but it has more significant outliners.

Table 2: Summary of the PCA data

26 September 2010	19 March 2011	8 May 2011
Min. :112.0	Min. :194.0	Min. :131.0
1st Qu.:143.0	1st Qu.:239.2	1st Qu.:165.0
Median :164.0	Median :258.0	Median :180.0
Mean :167.6	Mean :260.8	Mean :180.4
3rd Qu.:179.0	3rd Qu.:277.8	3rd Qu.:193.0
Max. :461.0	Max. :340.0	Max. :287.0

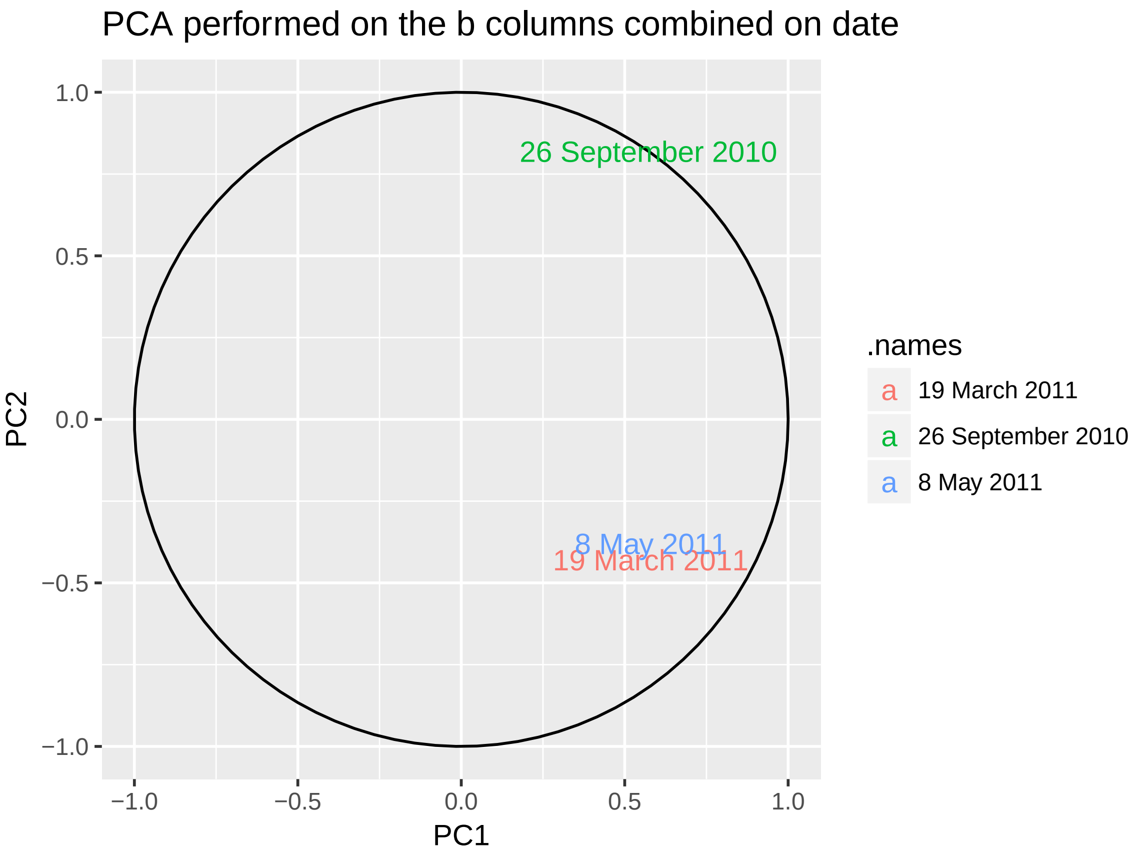


Figure 2: A PCA performed on the b columns data combined by date, september, march, may

In figure 2 the PCA can been seen, this PCA represents the dimensionality of the data. As we can see, the the 26 September date is on a different position than the 2 other dates. This can be explained by the simple fact that seasons exist. The 26 September date is taken in the fall, and the 2 other dates are taken in spring and summer.

Algorithm performance

Batch run

To determine which algorithm that performs best on this dataset, a batch run was executed with all the standard algorithms. These were executed in Weka and the standard settings were used, except for the SVM which used the setting in the original paper.

The accuracy of these standard algorithms can be seen in table 3. The table shows significant differences in the different algorithms, however, the more advanced algorithms perform much better. MLP and SVM perform best on this dataset, meaning that these 2 algorithms need further examining to get the best algorithm for the dataset used.

Table 3: Accuracy of the different algorithms with the trainings data used.

Algorithm	Average Percentage Correct
ZeroR	29.81
OneR	83.48
IBk	95.46
C4.5	96.13
Logistic Regression	95.96
MultiLayerPerceptron	97.28
RandomForest	95.77
Simple Logistics	95.96
Support Vector Machines	97.33

C4.5 does also perform quite well, but has not the accuracy that SVM or MLP have, meaning that this algorithm is not the best for accuracy in this specific dataset. But to incorporate speed into the decision, a speed test was performed. These results were gathered on a Intel Core i7 7700k and 16GB of RAM memory.

This results can be seen in table 4, which represent the time it took to train the different models. One time stands out: MLP. This algorithm took a long time to train with little data. This is due to the fact that the algorithm uses an Artificial Neural Network (ANN).

Table 4: Time used for each algorithm to train the algorithm

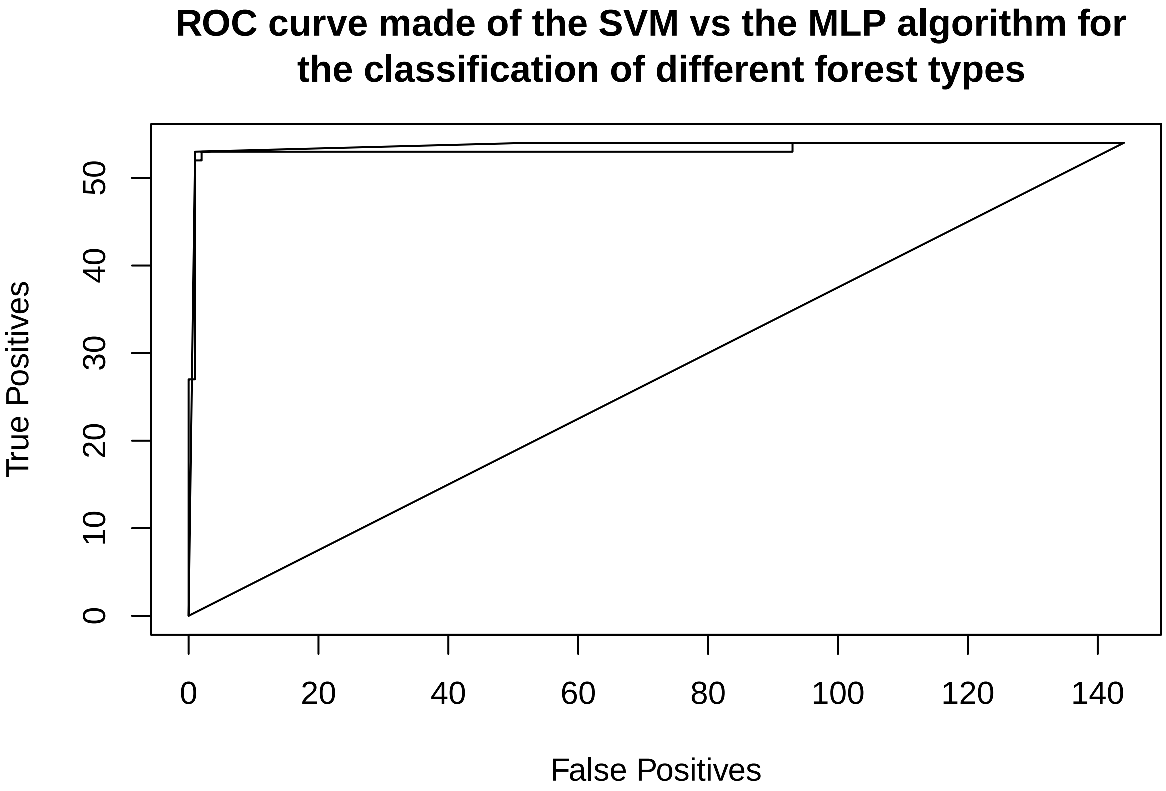
Algorithm	Average Time Cost
ZeroR	0.00014
OneR	0.00313
IBk	8e-05
C4.5	0.00339
Logistic Regression	0.02406
MultiLayerPerceptron	1.232
RandomForest	0.02565
Simple Logistics	0.02115
Support Vector Machines	0.01473

These elapsed times give a view of the time elapsed for training, but there is more interest in classification time, this is because the classification algorithm needs to be fast as well. This is because of the fact that the algorithm needs to classify on the go. Here, we can see that SVM and MLP are not the quickest algorithms, but perform quite quick.

Table 5: Time used for each algorithm to classify new instances

Algorithm	Average Time Cost
ZeroR	2e-05
OneR	8e-05
IBk	0.00081
C4.5	4e-05
Logistic Regression	2e-05
MultiLayerPerceptron	0.00017
RandomForest	3e-04
Simple Logistics	5e-05
Support Vector Machines	0.00012

To view if there is a significant difference between SVM and MLP, a ROC curve was made of the 2 different algorithms. This roc curve can be found below.Between the different roc curves there is no significant difference.



Discussion

The results show that there are significant differences in the accuracy and speed of different algorithms. We can see, that SVM and MLP performed best compared to other algorithms. SVM has been used in the original paper, but MLP performs as good. MLP uses a neural network and can handle more input, and can generate more output, which makes classification more accurate when there are related attributes. SVM is a much simpler algorithm, which only takes one input. Looking at the difference in classification time, SVM and MLP perform significantly slower than other algorithms, but the time it takes to classify is close to 0 seconds, meaning that these algorithms are quick enough to quickly classify the different forest types on the fly. The time training, however takes much longer with MLP. This is due to the fact that MLP has hidden layers with nodes, which take a long time to calculate and create a model. In the case of this dataset, training time does not matter, when the model is trained, it never needs to be trained again, making training time not an important metric to choose for MLP or SVM. SVM is an algorithm that has been used a lot in image classification, the original paper uses SVM as well, but the results shown in the table with accuracies show that MLP has also a great accuracy based on the trainingsset. The time training are run on a normal PC, and thus the time used for training and testing can be improved by using more advanced methods like servers and GPU algorithms. MLP has not much great implementations to run on a GPU, which can decrease the time needed to train and classify the different forest types.

As we can see in the ROC curve that was made for the SVM and MLP algorithms, there is little difference in the accuracy, meaning that both algorithms perform equal. The date can give different results, the dimensionality that has been shown in the PCA, represents this. The dataset only included 3 dates, which can give not quite accurate results, because each season, and for that matter, dates, give different spectral values. Meaning that more dates can be added to get an algorithm perform better on new datasets.

Conclusion

SVM and MLP are both great algorithms to use with the dataset in this paper. The final choice was made on MLP, because the setting can be modified more to create more accurate results. However, to use MLP or SVM on other datasets can give different results. Even on same datasets different results can be shown with the same algorithms that have been used, in the original paper, SVM resulted the best algorithm, but this paper, shows that MLP gives accurate results as well, and thus not a lot is clear about the best algorithm for each different dataset, which can be further explored to create a model that will determine the best algorithm that can be used with the different kinds of datasets. Also more advanced computer techniques could also improve the time for training and testing ML algorithms. Simple algorithms like SVM can be used, but also more advanced algorithms with more layers can be used, which can be easily modified to run faster on multiple cores, but that will need further research on improving existing algorithms to work more efficient.

Adding more dates, and including more variables can also give more accurate results. These additional variables can be texture variation of the soil, or biomass. The classification with different datatypes can also further improve the results, high-resolution imagery or hypersprectral imagery for example.

Project proposal

There are some possibilities to extend this research into different projects, the first being to write a complete wrapper that can take live drone images that are sent to the program and run them thru the trained model. The purpose of that program would be to be able to connect a pc to it, and fly with a plane to map the entire forest in one go. The deliverable of this project would be to have a functional program that can take information from a camera to classify different forest types. This can be tested to run it on a raspberry pi with a camera for testing purposes. This program can be very useful to foresters that need to keep an eye on the forest and this program can help visualizing problems that may not be visible to the naked eye. The other option is to improve the speed of the algorithms to write a program that can process the images on a GPU and then send it to a program on a cpu to perform the classification, this can be done due to the fact that GPU's are meant to process on a highly multithreaded basis and can thus quickly handle high quality images, thus improving the classification of the algorithm.

References

[1] Johnson, B., Tateishi, R., Xie, Z., 2012. *Using geographically-weighted variables for image classification. Remote Sensing Letters*, 3 (6), 491-499.

[2] *Forest type mapping Data Set* Retrieved September 11, 2018, from https://archive.ics.uci.edu/ml/datasets/Forest+type+mapping

[3] *Introduction to Principal Component Analysis (PCA)* Viewed September 24 2018, from https://tgmstat.wordpress.com/2013/11/21/introduction-to-principal-component-analysis-pca/

[4] Russell, G., Congaltol., 1988 *A Comparison of Sampling Schemes Used in Generating Error Matrices for Assessing the Accuracy of Maps Generated from Remotely Sensed Data*

[5] Mountrakis, G.,Im, J., Ogole, C., 2011 *A Support vector machines in remote sensing: A review*

[6] Jose M. Pena, Pedro A. Gutierrez, Cesar Herves-Martinez, Johan Six, Richard E. Plant and Francisca Lopez-Granados, 2014 *Based Image Classification of Summer Crops with Machine Learning Methods*

[7] Raczko, E., Zagajewski, B. 2017 *Comparison of support vector machine, random forest and neural network classifiers for tree species classification on airborne hyperspectral APEX images*

[8] Zanaty, E.A., 2012 *Support Vector Machines (SVMs) versus Multilayer Perception (MLP) in data classification*