PLANT CLASSIFICATION:

USING MACHINE LEARNING TO CLASSIFY POTATO AND SUGAR BEET PLANTS

The algorithm has to decide which image is a sugar beet plant and which a potato plant. Can you tell?

David Swinkels (920714820090)

Ping Zhou (900405987130)

Course: Machine Learning

Wageningen University

10 March 2017

INTRODUCTION

Farmers need weed control on their acres. Weed control helps them to prevent diseases from spreading and to prevent nutritional resources to be taken from their crops. Weeds are undesired plants in a particular situation. In our situation unwanted volunteer potatoes are growing between sugar beets. These volunteer potatoes can spread late blight and facilitate harmful soil nematodes. The potato plants also compete with the sugar beets for the soils' resources. Nowadays farmers still need to walk the entire field twice per season with pesticides to get rid of volunteer potatoes (figure 01). Weed control is currently a costly and inefficient operation.

Figure 01: *Overview of volunteer potatoes in sugar beet fields. From left to right: Volunteer Potato Plants growing in field of sugar beet, machine to remove potato plants and farmer performing manual weeding of volunteer potatoes*

An automated weed control system can make the removal of weeds more efficient, cheaper and more eco-friendly. To detect which plants should get a small dose of pesticide, the system needs to be able to identify crops and weeds. Sugar beets should not be harmed at all and potatoes should be removed. This report is going to investigate the prediction accuracy of classifying volunteer potato plants and sugar beet plants. Images of potato and sugar beet plants are the input for the model (figure 02). The output should be a class of either potato or sugar beet. Classification of sugar beet plants and potatoes is imperative to create a weed control system for sugar beet fields.

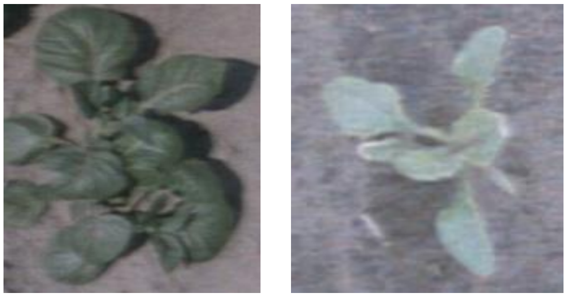


Figure 02: *Potato (46) and Sugar Beet Plant (3)*

How is this report structured? First, the applied classification methods are explained in the methodology. Second, the classification results are shown in the results. Third, these results will be interpreted in the discussion. Lastly, a conclusion will be given on the best classification approach and its practical applicability.

METHODOLOGY

In this chapter classification methods and the experimental setup will be discussed.

Classification Methods

Several statistical learning approaches will be used to classify the weeds and plants: logistic regression (GLM), linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), K-nearest neighbours (KNN), classification tree, pruned tree, random forest, boosted tree, and support vector machines (SVM).

Variable selection was performed for logistic regression. One model used all variables, one used a stepwise forward selection of variables and one model used a stepwise backward selection of variables. The selection of variables by the logistic regression model - all, forward and backward - are used in the GLM, LDA, QDA and KNN. The variable selection gives insight into which visual words are significant predictors of classification.

Parameter selection was performed on KNN (k), Random Forest (n.trees) and SVM Linear (cost), SVM Polynomial (cost, degree) and SVM Radial (cost, gamma). Tuning the models on their parameters improves the prediction accuracy of classes.

Experimental Setup

Identification of sugar beet and potato plants needs several steps: plant identification, feature extraction and classification. First, photos need to be made of the sugar beet field by a sensor and images of individual plants need to be extracted. Second, feature extraction is done on these images of the plants by using the bag of visual words approach. By doing feature extraction you can classify a plant based on their features. Grayscale images of plants are divided into a dense grid of overlapping cells. For each grid cell a SURF feature is computed that describes the image content in that cell as a vector (Bay et al, 2008).

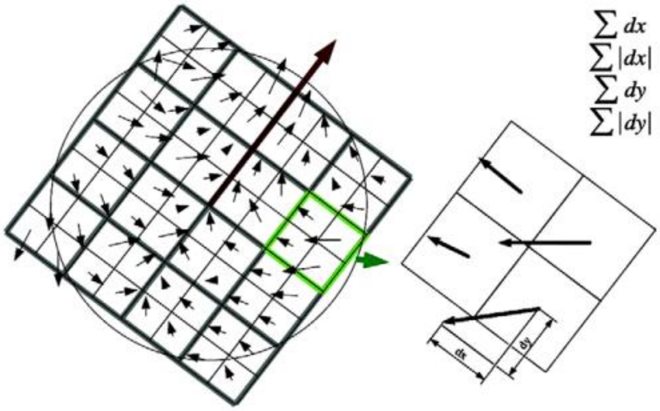


Figure 03: *SURF Feature: Representation of vectorized grid cells based on the gradient of the raster (Bay et al, 2008)*

The weakest SURF features, which are too homogeneous, are not used (alpha<0.2). The remainder of the features is clustered into K clusters with K-mean clustering. K is a cluster that refers to a type of visual word. A visual word can in this case be a nerve of a leaf, gradient colouring of a leaf or a tip of a leaf. The visual words are trained on the training dataset. Visual words are only derived from the training dataset, because the model cannot be trained on the test or validation data. Looking for visual words in test or validation data is training the model! Secondly only one set of visual words should be used for the training, test and validation datasets to do uniform classification of plants.

The frequency of visual words occurring will be different in each image. A potato plant image will perhaps have eight pointy leaf tips and a sugar beet plant only two. The pointy tips of the potato leaves and the rounded tips of sugar beet leaves can be seen in figure 04. The amount of features belonging to one visual word gives the value for each predicting variable. The frequency in which certain visual words occur is used as predictor for classification response of potato or sugar beet plant.

A training, test and validation dataset of each 135 observations have been provided for five visual word values. These values are 10,20,40,80 and 160. Out of these 135 observations 55 observations were potato plants and 80 observations were sugar beet plants. The response is a nominal variable: either potato (p) or sugar beet (p). The predictors are a continuous variable: the frequency of K visual words.

The test and training dataset are combined to get 270 total observations and then randomly split to get 135 training observations and 135 test observations. Model assessment is done by randomly setting test and training data. The test observations generate a model. The validation dataset tunes the parameters of the model and helps to prevent the model from overfitting. It can provide an estimate of the test error. The model is used to predict the class of the test observations based on its predictors. The main result is the test error rate. This value shows the efficiency of classifying plants correctly, where 0.0 is no mistakes and 1.0 is nothing but mistakes. Varied classification methods are applied to the five datasets.

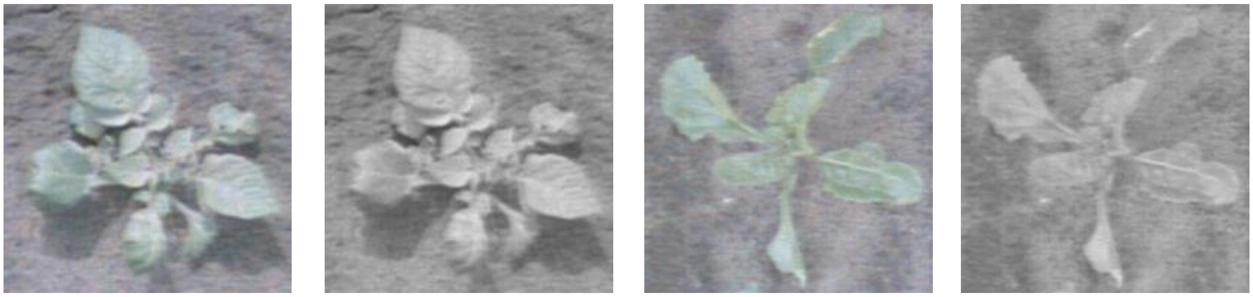


Figure 04: *Colour and Grayscale Figures of Potato Plant (39) on the left and Sugar Beet Plant (5) on the right*

Results

The outcomes of the variable selection, parameter selection and classification will be shown.

Variable selection

The Logistic Regression was used to select significant predictors. In this case significant predictors mean significant visual words. Forwards and backwards stepwise selection have been used to select variables. The stepwise variables selection methods look for best model fit (Adj. R2). Especially when the dataset has a lot of variables it is important to remove unimportant variables to improve the model fit (see table 01). The dataset with 160 variables had an AIC of 270. With forwards and backwards selection it went down to an AIC of 54 and 42. This is a big improvement. For the dataset with 10 variables there was only a minor bonus when some variables were removed. Logistic regression shows that the model can improve if variables are selected by forwards or backwards method.

Variable Selection with Logistic Regression: Full, Forwards and Backwards Selection

	Full Selection Model		Forwards Selection Model		Backwards Selection Model	
Dataset	Amount of Variables	AIC	Amount of Variables	AIC	Amount of Variables	AIC
K010	10	164.8	7	160.1	4	161.9
K020	20	174.9	10	159.4	4	159.3
K040	40	145.6	17	114.7	20	117.7
K080	80	162	26	54	24	50
K160	160	270	26	54	20	42

Table 01: *K-Nearest Neighbor Training- and Test Errors*

Parameter selection

The parameter selection of K-Nearest Neighbors is important to get a good fit. When calculating K-Nearest Neighbor with low K-values the training errors are low and test errors are high (overfitting). K = 1 gives a training error of 0.00 and a test error of 0.34. For high K-values training and test errors are both high. The optimal K for test errors is around K equals 18 with a test error of 0.26. It is important to tune the parameters to a correct value.

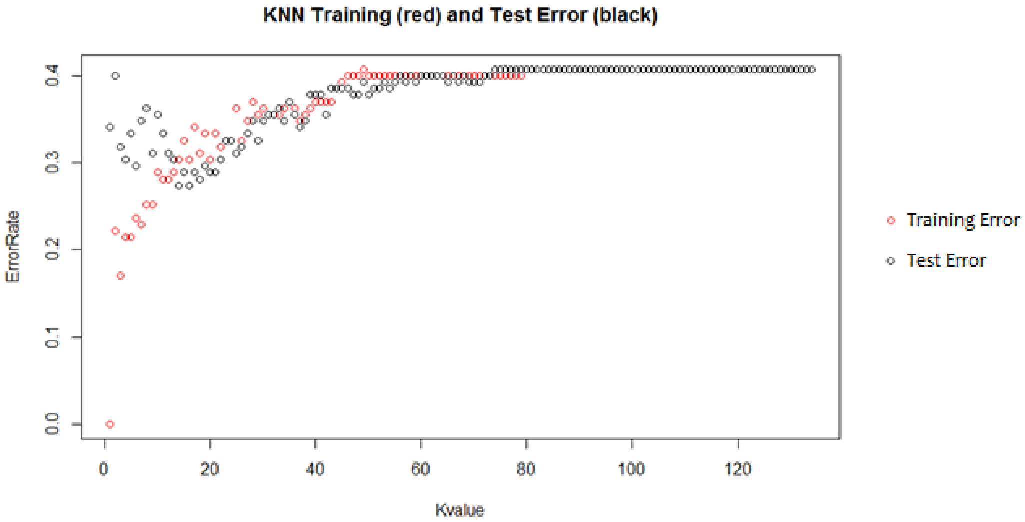


Figure 05: *K-Nearest Neighbor Training- and Test Errors*

Classification Outcome

The index (Train Error Rate – Test Error Rate) indicates whether the model is overfitting(> 0). The trees are overfitting a little bit, but generally most models are not overfitting (figure 06). In conclusion, LDA and SVMRad methods give the best fits for the trained model and predicted test classification. This does not mean that these classification methods give the best prediction accuracies.

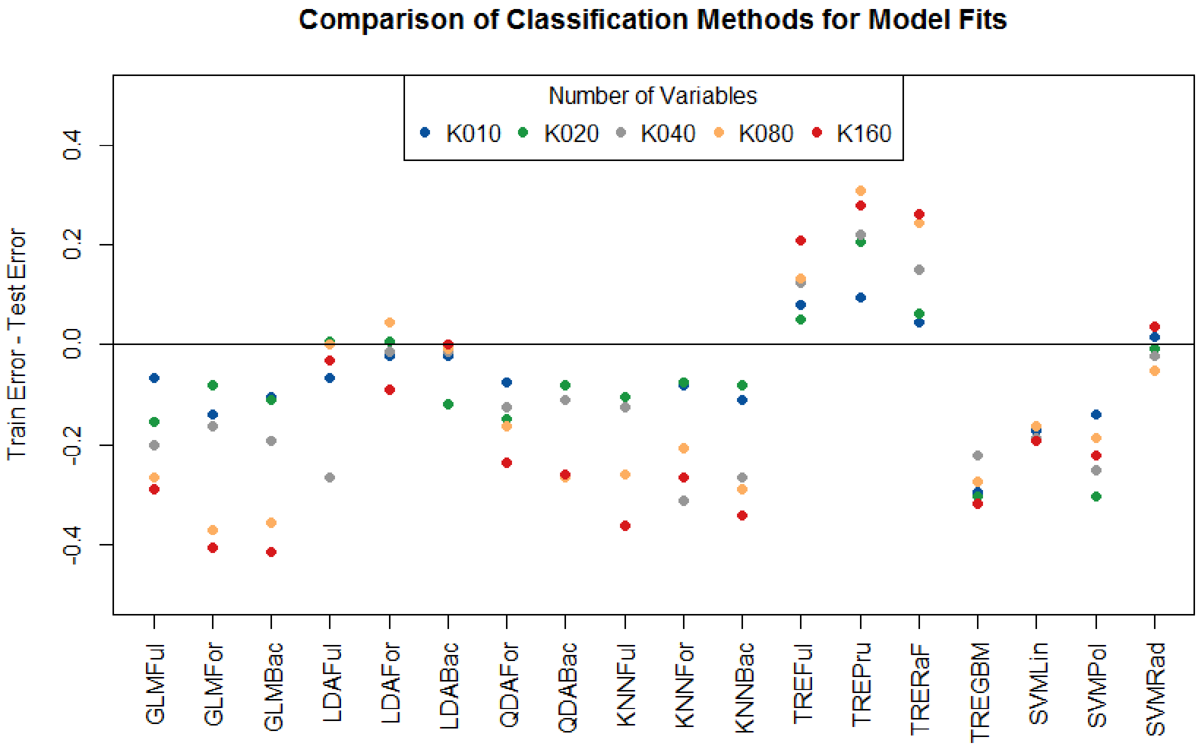
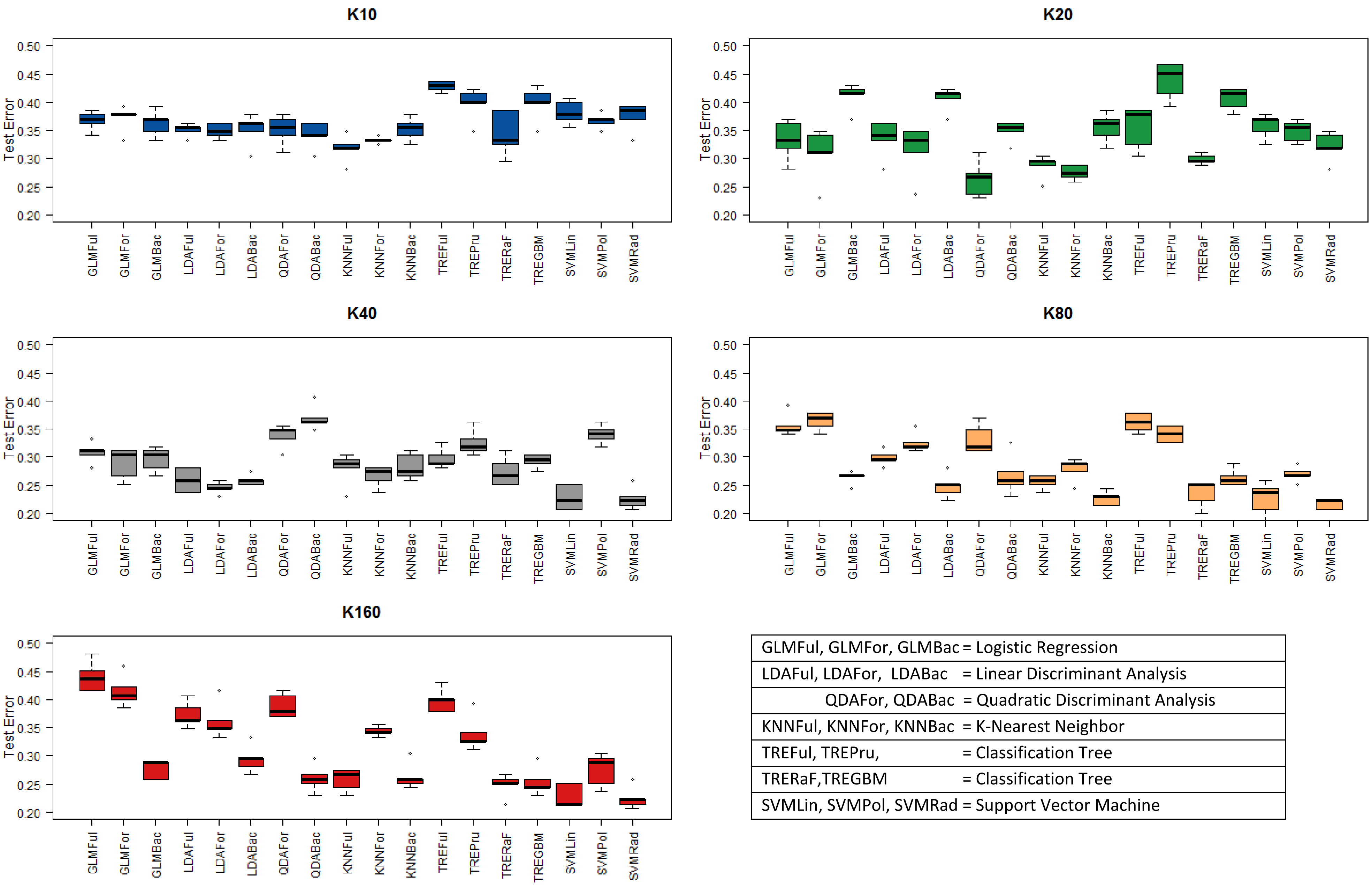


Figure 06: *Comparison of Model Fits with Training and Test Error*

The best classification methods have the lowest test errors. First the general trends for the five datasets with different amount of visual words is reviewed (figure 07). For the dataset with only 10 visual words the test errors are around 0.37. K20 has test errors around 0.35. K40 has test errors around 0.30. K80 has test errors around 0.28 and K160 has test errors around 0.28 too. There seems to be a trend that for higher K-values test errors are lower.
Second, the best classification methods are listed. Respectively for the five datasets from K10 to K160 these are the best classification methods: KNNFul, QDAFor, SVMRad,SVMRad and SVMRad. These five classification methods have the lowest mean test error in the model assessment.
Third, the overall best performing classification methods are listed. For various amounts of visual words KNNFul, Random Tree and SVM's are performing good with low test errors.

Figure 07: *Model Assessment of Classification Methods for Various Datasets (K10, K20, K40, K80, K160) with five random samples*



GLMFul, GLMFor, GLMBac	= Logistic Regression
LDAFul, LDAFor, LDABac	= Linear Discriminant Analysis
QDAFor, QDABac	= Quadratic Discriminant Analysis
KNNFul, KNNFor, KNNBac	= K-Nearest Neighbor
TREFul, TREPru,	= Classification Tree
TRERaF,TREGBM	= Classification Tree
SVMLin, SVMPol, SVMRad	= Support Vector Machine

DISCUSSION

In this chapter we will interpret and discuss our results.

Which classification methods do work best? From figure 07 we can see that for different datasets(with different number of variables), different classification methods have different performance. The results are varied. However generally speaking, SVMRad method works best. KNNFul, other SVM's and Random Forest work quite good as well. This is because the decision boundary is more likely to be non-linear. Non-linear models such as SVM's, Random Forest and KNNFul perform well on all random samples and five datasets.

There is a trend visible in the number of variables for each dataset. A lower amount of variables leads to a lower prediction accuracy and a higher amount of variables leads to a higher prediction accuracy. This is because there are more variables or visual words, which can be used to predict the class correctly. The downside of many variables is computation time. More variables can help to get higher prediction accuracy.

Some supporting meta-techniques can help to improve prediction accuracy, computation efficiency or model assessment. Forward and backwards selection of variables is helpful for big datasets to reduce the amount of variables or to select a model with a good fit. Especially backwards selection improves the prediction accuracy for larger amounts of variables. The downside of forwards and backwards selection is a high computation time, when stepping through variable selection. The variables were also removed and added one by one. Thus the variable selection is a linear process, but it can be that some variables together had an effect on the prediction. Backwards selection can be helpful to reduce the amount of variables and increase the prediction accuracy of datasets with a large amount of variables.

Cross validation was used to randomly sample datasets to assess the model quality. Cross validation is very helpful to give insight into the quality of the model. If the test and training data are only run once, then there is no information whether the model has a high probability of predicting the same result again. A confidence interval can be calculated to assess each classification method. Therefore a for-loop of 100 iterations was setup to go through all classification methods with 100 random test samples (figure 08), but there was a code error which gave bad results for the SVM's. Therefore cross validation was repeated on 5 random samples (figure 07). Cross validation is helpful in selecting the best model by assessing each classification method.

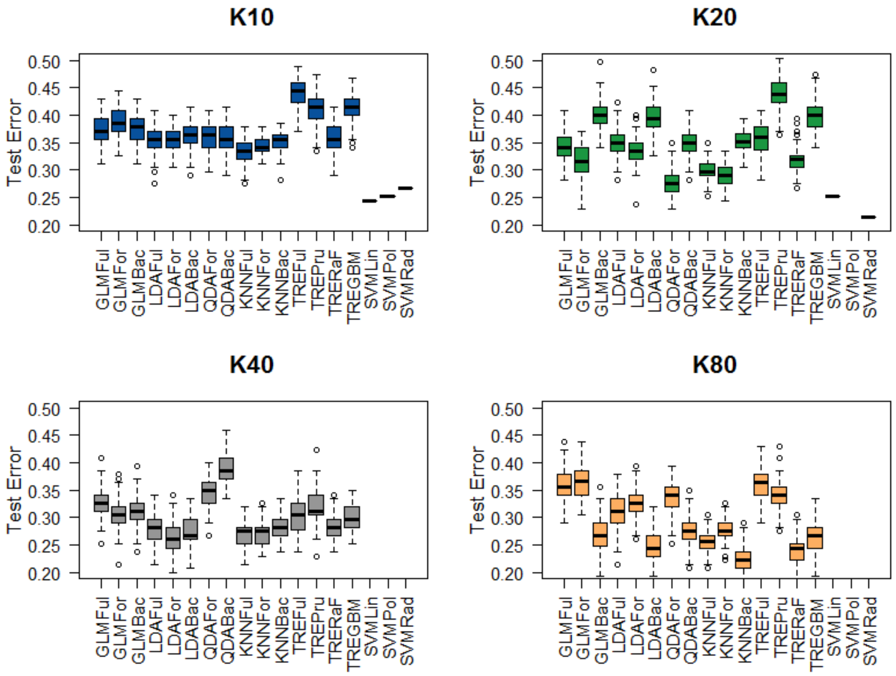


Figure 08: *Cross-Validation for Various Datasets with 100 random samples (Wrong Values for SVM's)*

In all the analyses the models were fitted to the training data. The fitted model is used to predict the outcome of the test data. If the model parameters make the model fit very flexible, the model can fit the training data really well. However the predicted test data can then have a high error rate, because it does not fit to this flexible line. This is called overfitting. The fit of the model to the training data can also be too simplistic. Then your test model is too simple and underfits the data. Overfitting can be seen in figure 06 for the classification trees. Overfitting did not affect our results.

The classification results of the support vector machine with radial kernel is used to check misclassifications (cost=10, gamma=0.01) of the test data. It had a high prediction accuracy of 83%. Three potato plants that were misclassified as sugar beet, and three sugar beet plants, that were misclassified as potato, will be reviewed (see pictures below). The three potato plants that were classified as sugar beet were visually less sharp and had less pointy tips. A reason can be that the blurry images do not create SURF features for the small nerves, because these are not visible. The three sugar beet plants that were classified as sugar beets had some abnormal features: two plants, a darker leaf colour with more shades and an extra stone in the image. More unexpected SURF features in the sugar beet plants can classify the plants wrongly as sugar beet.

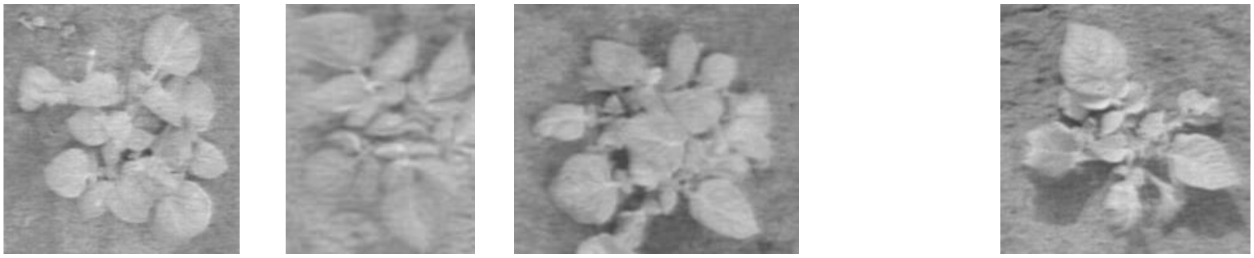


Figure 09: *Potato Plant Comparison - Left: Potato plants (test067 + test075 + test109) classified as sugar beet; right: Standard Potato Plant*

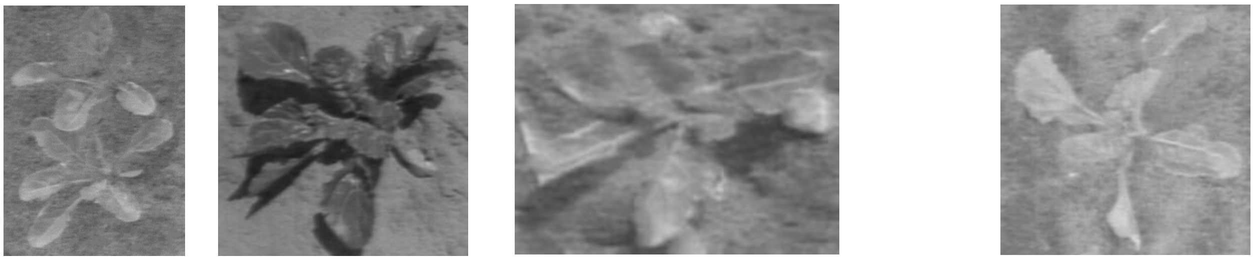


Figure 10: *Sugar Beet Plant Comparison - Left: Sugarbeet plants (test102 + test154 + test156) classified as potato plant; right: Standard Sugar Beet Plant*

Sometimes the farmer wants to keep all sugar beet plants (crops) and sometimes he want to destroy all volunteer potato plants (weeds). The prediction accuracy of all plants can be split into the prediction accuracy for crops (true sugar beet rate) and weeds (true potato rate). Using the highest prediction accuracy of potato plants can be useful when all potato plants need to be removed, because perhaps late blight is detected in the sugar beet field. A limited amount of crops can then be destroyed. In a normal case the true sugar beet rate can be used to let the farmer keep all his crops and profits. Then some weed can stay in the field. Sugar beet farmers need to make their agricultural business profitable and keep all their crops, but also want to remove most weeds . In some cases all weeds need to be destroyed. The best classification methods discussed in this paper can help to classify plants correctly to remove weeds.

Prediction accuracy of crops and weeds can be adjusted by setting a threshold. The threshold can be set on the probability rate of statistical tests. Normally the classification threshold is set on 0.5. Higher than 0.5 is classified as class “1” and lower than 0.5 is classified as class “0”. If the threshold is lowered to 0.2 more observations will be classified to class “1”, but the prediction accuracy of class “0” will be higher. Therefore lowering the threshold on the probability rate of being a crop can improve the prediction accuracy of crops and same works for weeds.
The classification performance can be improved by the location of plants. Sugar beet plants are planted on the acres in lines and have a certain distance between them. Basically sugar beet plants are points in a grid. Potato plants on the other hand grow on random locations and thus are not points on this grid. Performance can be improved by checking if the plants are on the grid. If plants are on the grid, a higher prediction accuracy is given to crops. So only potato plants that have a high probability rate to be a potato plant will be removed if they are on the grid. If the plants are off the grid, a higher prediction accuracy can be given to weeds. The location shows if the plants are on the grid, sugar beets are more likely to be on the grid and the prediction accuracy can be adjusted based on that information.

To end this chapter there is a discussion about the work process of this project. First, QDA gave an error on the full model for K80 and K160, because it could not handle more predictors than the amount of observations. It will stop the whole code `if (any(counts < parameters + 1))`. Second, it took much longer to calculate K40, K80 and K160 than K10 and K20. Computation time increases, because there are more predictors in each analysis to take into account. Thirdly there were some coding errors, which caused the support vector machines to get a very high result. It is important to critically check every result and to be aware what the code does.

Conclusion

This report set out to find the best classification method for potato and sugar beet plants based on the frequency of certain plant features. An answer will be given to this question and its applicability will be discussed.

The best classification methods obtained is a support vector machine with radial kernel. Generally non-linear models were performing better than linear models. A larger amount of predictors helps to get better prediction accuracy, but also increases computation time. In this case, with 160 visual words, the support vector machine with radial kernel reached the highest mean prediction accuracy of 78%.

K value	Methods	Mean Test Error
10	KNN Full	0.32
20	QDA For	0.27
40	SVM Rad	0.22
80	SVM Rad	0.23
160	SVM Rad	0.22

Support vector machine with radial kernel can be used in a weed control system to identify plants correctly with a prediction accuracy of 78%. For real-world applicability this prediction accuracy seems too low. Either potato plants would be left in the sugar beet field or some sugar beets would also get pesticide. For a real-world product you want better quality. Therefore this report proposes further fine tuning.

Further research can be done on the prediction rate of plants/crops, cross validation with larger samples and more classification methods. The true prediction rate of potato and sugar beet plants can be further researched, because this can help in the real-world applicability of the system to remove either all potatoes or keep crops intact. It can help to define a good threshold. The location of plants can also be taken into account more in further research.
Further research can also be done to do a cross validation with at least 50 random samples to assess the model better. Third, other classification methods such as neural networks can also be assessed.