Ludwig-Maximilians-Universität München

Bachelor Thesis

An optimized stopping rule for statistical boosting algorithms

by Veronika Huber

supervised by
Prof. Dr. Andreas Mayr

June 24, 2018

Abstract

Boosting algorithms are a relatively new way to fit models on high-dimensional data sets, which can be interpreted in the same way as classically fitted models. The algorithm offers the possibility to select variables already during modeling as well as an automatic control of the effect estimates, only regulated by one main tuning parameter: the stop point of the algorithm. A pre-selection of the possible variables or other preparations are no longer needed. This thesis deals with the stop point of the algorithm and the optimization of the stopping rule by using the one standard error rule.

Both classical and newly developed methods are applied on simulated data for two types of models: linear and logistic regression. Comparing the resulting models of both methods to choose the stop point, the newly developed method led to an improvement of prediction accuracy in both types of regression.

Finally, the two stopping rules were applied on two high-dimensional genomic data, each for one type of regression. Comparing the logistic boosted models, the prediction accuracy improved again when using the new method. In the case of linear regression, however, no improvement could be achieved. Overall, both methods yield similar results, but in some cases, the new method produces better results.

Contents

1 Introduction

One of the most important tasks in statistics probably is to describe the relationship of a variable Y and several possible influencing variables $X = (X_1, X_2, ...X_i)$ as precisely as possible. To describe the information that X provides about Y, a function $f(X)$ is searched, which explains the connection in the best possible way. The difficulty is to find a function, based on a sample of data, that is generally applicable (cf. James et al., 2013). A lot of options already exist to create the best possible regression model out of many potential predictors. One of the biggest challenges is to find the right trade-off between the bias and the variance of a model. The more complex the model, the lower the bias but the greater the variance; whereas the simpler the model, the larger the bias but the lower variance (cf. Fahrmeir et al., 2013). Methods such as stepwise selection or shrinkage control the variance by variable selection. Another way is to reduce the dimensions by linear combinations or projections e.g. using a principal component analysis (PCA) (cf. James et al., 2013). For high-dimensional data records, the task of finding the right model proves to be even more difficult as classical regression approaches can no longer be used. Many methods of treating this problem have already been proposed. These include various ways to pre-select the possible variables (Saeys et al., 2007). An alternative are model fitting processes where the variables are selected during the modeling process. The best known method so far is probably is the lasso operator (least absolute shrinkage and selection operator) (cf. Tibshirani, 1996). An innovative new approach, which uses a similar method to select the variables during modeling, is the boosting algorithm, which can be regulated by one main tuning parameter: the stop point of the algorithm (Mayr and Hofner, 2018).

The outline of this thesis is as follows: In Section 2, the boosting algorithm and the referring stop point are explained. For further improvement of this approach, a method is presented to optimize the choice of the stop point by using the one standard error rule, described in Section 3. To evaluate the results, the two methods are first applied to simulated data in Section 4 and finally tested for true data in Section 5.

2 Boosting Algorithm

Boosting is a relatively new approach, to select the variables of a model, and originates from machine learning (Freund, 1995). If the number of possible predictor variables is much larger than the number of observations, it is not longer possible to fit a classical regression model. This problem can be solved by using the boosting algorithm (Mayr et al., 2012). Boosting is suitable for high-dimensional data as it has the advantage of variables selecting while modeling (Mayr and Hofner, 2018).
In addition, the resulting models contain an implicit penalization and smaller effect estimates. Also, boosting offers a high degree of flexibility with regard to the various types of effects (Mayr and Hofner, 2018). All these features can be controlled by a parameter, namely the point at which the algorithm stops. The following concentration on the stop point will precede a detailed explanation of the algorithm.

2.1 Algorithm

The idea of boosting algorithms is to minimize the error residuals, by adding the variable that can best describe the residuals in each step. These residuals, i.e. the distance between the values estimated by the model and the observed data, are described by the negative gradient of a specific loss function. The objective is therefore to minimize this loss function.
In order to achieve this aim the kind of effect the variables have on the outcome (e.g. linear or smooth effects) has to be determined. They can be specified by the so-called base-learner. These base-learners describe what type of function is used for modeling the variables; i.e. if the variable has a linear effect, then a linear base-learner would be used leading to a simple linear regression model. Then the form of the loss function has to be defined which depends on the species of the outcome. Finally, the stop point m_{stop} of the algorithm must be set (cf. Mayr and Hofner, 2018).
The boosting algorithm will now include in each step the base learner that best describes the error residuals. For this iterative process, in each step, the loss function and the corresponding negative gradient are calculated. Then, all possible base-learners will be fitted on the negative gradient, i.e. the residuals, and only that one that suits best will be selected. This selected variable is then included in the model. After this update of the model, the negative gradient is recalculated and again all base learners are matched, including the variable taken in the previous step. Once more the best fitting base learner will be resumed. This procedure is repeated until the final iteration m_{stop} is reached (cf. Mayr and Hofner, 2018). The following graphic illustrates the procedure:

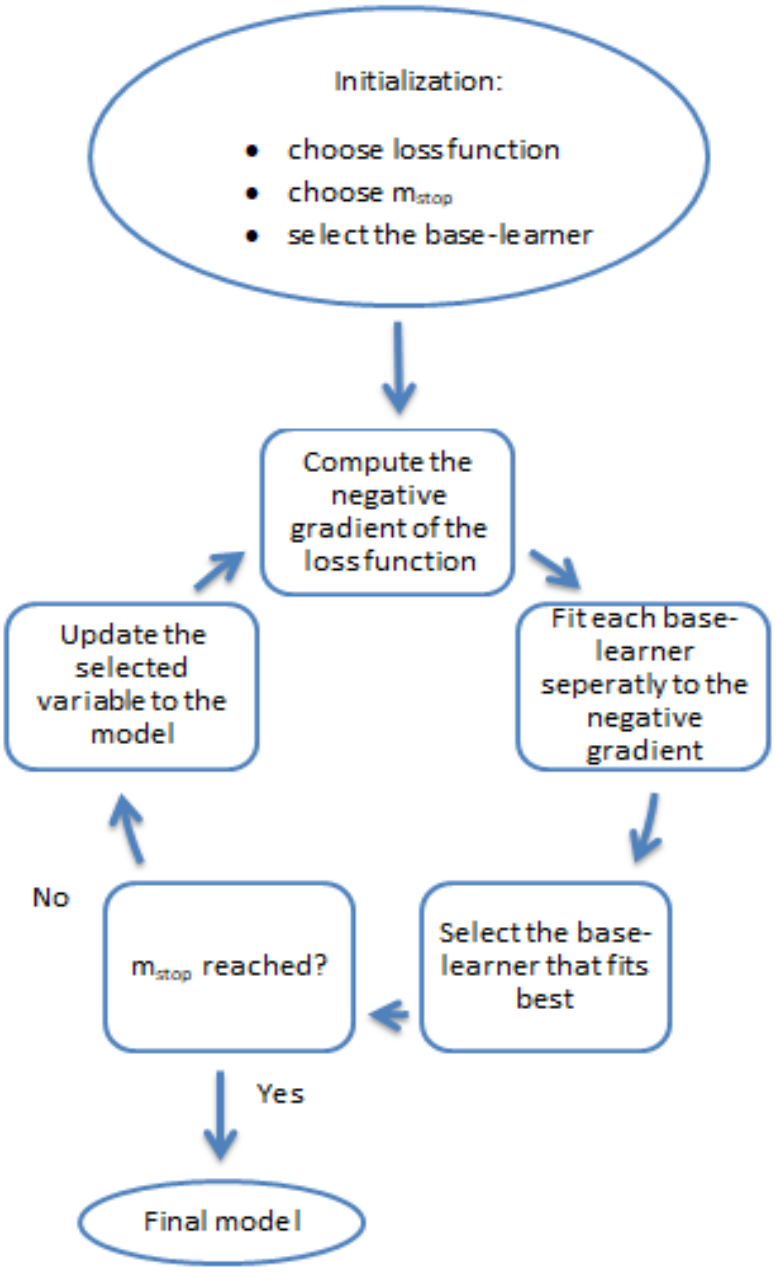


Figure 1: Shown is the basic process of the boosting algorithm. On the basis of the available data, a loss function must first be selected, the stop point of the algorithm must be set, and the base learners of the possible influencing variables must be defined. Thereafter, the negative gradient of the loss function is calculated and each base learner separately fitted to the negative gradient. Then selecting the base-learner who best explains the residuals and update the negative gradient. This procedure is repeated until the stop point is reached.

In this work, two types of outcome are examined. A metric target with $Y \in \mathbb{R}$, which uses the squared error loss. If this is scaled by the factor 1/2, the negative gradient of the function corresponds to the residuals. The loss function is composed as follows.

$$p(y, f) = \frac{1}{2}|y - f|^2 \tag{1}$$

Where y are the true values and f the estimated ones. With the associated population minimizer (cf. Bühlmann and Hothorn, 2007):

$$f^*(x) = \mathbb{E}[Y|X = x] \tag{2}$$

The other outcome is binary with $Y \in \{0, 1\}$. The target size is recoded here to $\tilde{Y} \in \{-1, 1\}$ where $\tilde{Y} = 2Y - 1$, so the used loss function is the following negative binomial log-likelihood:

$$p(y, f) = -(y\log(\pi(f)) + (1 - y)\log(1 - \pi(f))) = \log(1 + \exp(-2\tilde{y}f)) \tag{3}$$

Where $\pi(f) = \mathbb{P}(Y = 1|x)$ is the probability of success in each trial (cf. Hofner et al., 2014). The related population minimizer can be shown as follows (cf. Bühlmann and Hothorn, 2007):

$$f^*(x) = \frac{1}{2}\log\left(\frac{\pi(f)}{1 - \pi(f)}\right) \tag{4}$$

To prevent overfitting and to move slowly onto the minimal loss, the selected variables are multiplied by a constant step length ν before being included in the model (Mayr and Hofner, 2018). It has been demonstrated that the value of ν does not have much influence as long as it is small (Bühlmann and Hothorn, 2007). In this work $\nu = 0.1$ is set. More details can be read in the article of Friedman (2001).

2.2 The Choice of the Stopping Iteration

The choice of the stop point m_{stop} is decisive for the shape of the resulting model. The smaller m_{stop} the fewer variables are included in the model, given that only one variable is updated in each step of the algorithm. Stopping too early implies that not all important variables are recorded, stopping too late means too many unimportant variables are in the model. Therefore the choice of m_{stop} is very important. Stopping the boosting algorithm before it converges not only results in variable selection during the process, it also reduces the risk of overfitting and improves the prediction accuracy. In addition, it also leads to the fact that the effect estimates are reduced.

This, in turn, leads to a smaller variance and thus to a more stable model. The former stop can additionally reduce the complexity of the model. Fewer variables in a model lead to a lower complexity which can improve the prediction accuracy. By choosing the m_{stop}, the resulting model can be mainly controlled (cf. Mayr et al., 2012).
A common method is to run the boosting algorithm until a certain iteration and then determine a measure of the prediction quality for each iteration step. Followed by a selection of the iteration step that brings the best results. To determine the quality, the record is subdivided into a test and a training record, for example, by cross-validation, bootstrapping or the like. On the training data sets, a model is fitted which is then applied to the test data sets. Next, the error is calculated via the chosen loss function. After all, the iteration step is picked, which achieves the smallest error (cf. Mayr and Hofner, 2018).
When selecting the squared error loss function, the smallest possible quadratic error is selected. When using the negative binomial log-likelihood as the loss function, one minimizes the upper limit of the mean misclassification error (cf. Bühlmann and Hothorn, 2007).

3 One Standard Error Rule

A main point in statistical modeling is the choice of the best model. In a collected data set there are usually several possible variables which could be used to explain the outcome of interest. The aim is then to choose the variables that give the best prediction models on the test data. There are several possibilities to find this model such as cross-validation or subdivision of the data into a training and test dataset. The models are then fitted on the training data and tested on the remaining data (Fahrmeir et al., 2013). The one standard error rule (one SE rule) is an approach to improve the model quality, firstly introduced by Breiman et al. (1984) for the selection of decision trees. Decision trees are a series of binary decisions. The estimated value is determined by following the branches of the tree. It is now desirable to find the subtree with the binary choices that will give the best results. There are a variety of subtrees $\hat{T}_k$ possible. If $\hat{R}_{T_k}$ is an estimation of the misclassification cost, the chosen tree would be (c.f. Breiman et al., 1984) :

$$\hat{R}(T_{k0}) = \min_k \hat{R}(T_k) \tag{5}$$

The binary separation property of the decision trees results in the fact that even small changes in the values or the uses of another part of the data set as training

data lead to a completely different optimal subtree $\hat{R}(T_{k0})$ (c.f. Breiman et al., 1984). The one standard error rule (one SE rule) was created to reduce this instability. This should be realised by choosing the simplest tree whose accuracy is within the range of one standard error around the point with the lowest misclassification cost. If $\hat{T}_{k0}$ is the tree with the lowest estimator of the misclassification rate, the one SE rule would look as follows:

$$\hat{T}_{k1} \leq \hat{T}_{k0} + SE(\hat{T}_{k0}) \tag{6}$$

where $\hat{T}_{k1}$ would be the new chosen tree (c.f. Breiman et al., 1984).
This method can not only be used for regression trees, it can also be applied to the area of statistical modeling. An example for that would be the lasso estimate, a method for variable selection (LeBlanc and Tibshirani, 1998).
Lasso stands for "Least Absolute Shrinkage Selection Operator" and at the same time, this operator performs a variable selection while minimizing the coefficients. Again, one tuning parameter λ determines the strength of the shrinkage applied to the estimates (cf. Tibshirani, 1996). Just like the regression trees, the value for λ, where the mean error of the cross-validation is the lowest, would be chosen. If now the one SE rule is used, a larger value would be selected for λ, which is just in the one standard error interval (cf. James et al., 2013). Figure 2 shows an example for an application.
It shows the mean squared error, which was determined by a 20-fold cross-validation, depending on λ using some example data of the 'glmnet' package in R. The shaded area around the curve represents the interval of a standard error for each λ:

$$CV(\lambda) \leq CV(\hat{\lambda}) + SE(\hat{\lambda}) \tag{7}$$

where $\hat{\lambda}$ is the λ with the smallest error:

$$\hat{\lambda} = \min_{\lambda} CV(\lambda) \tag{8}$$

By this procedure, a much larger λ is now selected which means that the model gets smaller.

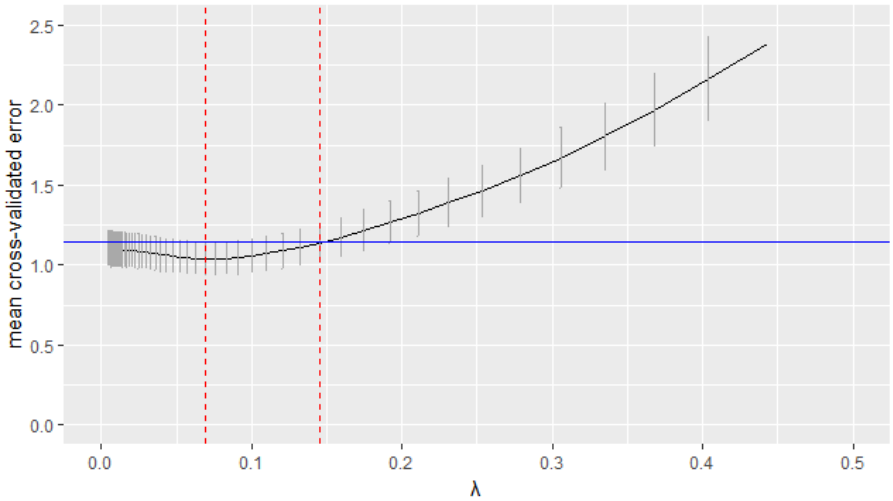


Figure 2: An example of using the one SE rule in combination with the lasso, a 20-fold cross-validation was performed and the mean squared error where calculated. The black solid line shows the mean squared error for each λ and the gray shaded area around marks the interval of one standard error. The left dashed line marks the λ that would classically be chosen. The blue solid line marks the upper limit of the allowed error after the one SE rule, following it to the point to the right where it hits the black curve marks the newly chosen λ (right red dashed line).

The boosting algorithm has many similarities to the lasso estimator and both methods usually lead to similar models (Hepp et al., 2016). The aim is now to apply the one SE rule to the boosting algorithm.

3.1 Function for the One Standard Error Rule

To implement the one SE rule the statistic program R version 3.4.3. is used. First the stop point according to the previously used rule was calculated. Therefore, the functions available in the 'mboost' package are extended. The existing `cvrisk ()` function performs a 25-fold bootstrap and calculates the mean errors ϵ :

$$\epsilon = \frac{1}{k} \sum_{i=1}^{k} (p_i) \tag{9}$$

k stands for the number of sampled bootstraps and in this case it is 25 (cf. Hofner et al., 2014). p_i again depends on the choice of the loss function and corresponds to the errors of the k individual samples. Using the quadratic error loss, ϵ then equals the mean squared error, using the negative binomial log-likelihood, it corresponds to the mean misclassification error (cf. Bühlmann and Hothorn, 2007).
The mean errors are compared and the stop point is chosen where the error is the

smallest (Mayr and Hofner, 2018). The `mstop()` call returns the value for m_{stop} where the mean error is the smallest (Hofner et al., 2014). Typically, the model that results from this step of interpolation would now be used.

The one SE rule now starts at the selected m_{stop}. Therefore a function was created that operates as follows.
First, the standard error for the mean error ϵ is calculated. Formula 10 shows how it calculates in this case (cf. Wilcox, 2010):

$$\mathrm{sd}(\epsilon) = \sqrt{\mathrm{Var}\,(\epsilon)} = \sqrt{\mathrm{Var}\left(\frac{1}{k}\sum_{i=1}^{k}(p_i)\right)} = \sqrt{\frac{1}{k^2}k\mathrm{Var}(p_i)} = \frac{1}{\sqrt{k}}\mathrm{sd}(p_i) \tag{10}$$

The calculated standard error is now added to the error at the position of the previously selected m_{stop}. This value forms the upper limit of the allowed error. That means the allowed error after the one SE rule is:

$$\epsilon_{max} \leq \epsilon_{m_{stop}} + sd(\epsilon_{m_{stop}}) \tag{11}$$

The third step chooses now exactly that stop point, where the mean error is just within the allowed range. Figure 3 shows an example of the application of the one SE rule. The black dashed line is the selected m_{stop} according to the previous method. The blue dotted line marks e^2_{max}. Following the blue line as far to the left until it hits the error curve the stop point after the one SE rule is reached, marked here in red.
By this rule, therefore, a stop point is selected which is smaller than that by the previous method.

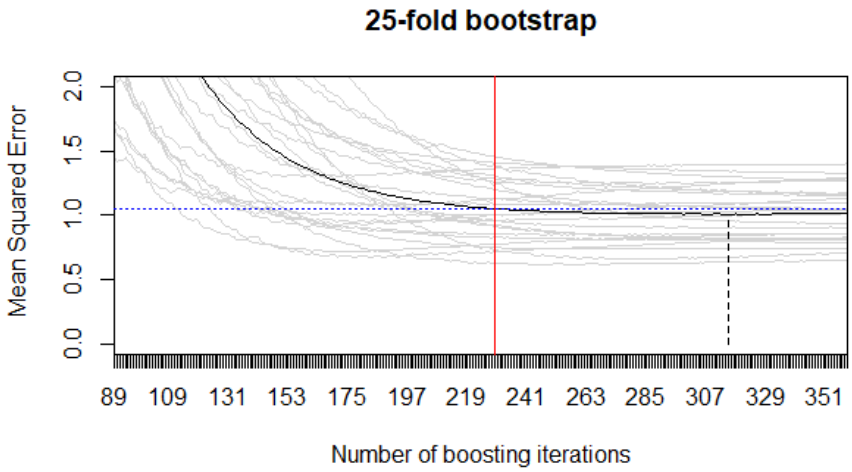


Figure 3: An example of using the one SE rule for a boosted linear model that uses the quadratic error loss to find the optimal stop point. The black dashed line shows the stop point where the square error is the smallest. At this point, the standard error of the point was calculated and used as the upper allowed error limit, here shown with the blue dotted line The stop-point according to the one SE rule results from following the blue dotted line to the point to the left where it meets the curve of the loss function. This point is marked here with a red line.

4 Simulation

To analyse and compare the results of the two methods, we use simulated data. The methods are applied to two types of settings. A classical boosted linear regression model with a metric dependent variable and a logistic boosted regression model with a binary outcome. For both cases, data are simulated and the results of the previously used method compared with those of the new approach, the one SE rule. Simulation studies are very convenient for comparing variable selection as the variables that have an impact on the outcome are already known. In addition, to the selected variables and their estimated coefficients, the prediction accuracy of the models will be considered.

4.1 Data

For the linear regression model the following dataset was constructed: first, a data set with 1100 observations and 100 possible normal distributed predictors was created.

Of these 100 variables, six were simulated with influence on the dependent variable,

$$y = \beta_0 - 1\beta_1 - 3\beta_2 - 5\beta_3 + 1\beta_4 + 3\beta_5 + 5\beta_6 + \epsilon \tag{12}$$

ϵ describes the standard error and is normally distributed. 100 of the observations are used as training data since this is the size of a typical sample, which is collected under real conditions. The remaining 1000 serve as test data. With such a large test record, the results of the review will become more accurate. Now, the boosting algorithm is executed with the training data and the stop point is determined once by the least squares error method used so far, and once by the one SE rule. For both methods, the selected variables are compared, as well as the estimated coefficients. In addition, the Mean Squared Error (MSE) on the training data and the Mean Squared Error of Prediction (MSEP) on the test data are calculated as a measure of the prediction accuracy.
The mean squared error MSE between the predicted values $\hat{y}$ and the exact value y is calculated as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{13}$$

where N stands for the number of observations. The calculation of the MSEP is analogue, with the only difference that the model is now applied to a test data set (c.f. James et al., 2013).
The procedure described above is now repeated 100 times. So 100 datasets are simulated and then modelled with the boosting algorithm. Finally, there are the results of 100 simulation steps.

Similarly, the data for the logistic regression are created. Again data set with 1100 observations and 100 possible binomial distributed predictors were created and the outcome depends on six of the possible predictors. The probability of success, on each step, π is calculated as follows:

$$\pi = \frac{\exp(\eta)}{1 + \exp(\eta)} \tag{14}$$

where η is the linear predictor:

$$\eta = \beta_0 + 1\beta_1 + 3\beta_2 + 5\beta_3 - 5\beta_4 - 3\beta_5 - 1\beta_6 \tag{15}$$

Once more, 100 of the observations are used as training data and the remaining as test data. Then the boosting algorithm is performed again and the selected variables

and estimated coefficients of the new and old methods are compared.
As a measure of the prediction accuracy, we use the receiver operating characteristic (ROC) curve and the associated area under the curve (AUC). The ROC curve reflects the 1-specificity on the x-axis against the sensitivity on the y-axis. Sensitivity describes the proportion of correctly classified events and the specificity describes the proportion of true negative events. The values of 1-specificity can be interpreted as the number of false positive classified events. The AUC value corresponds to the area under the ROC curve and can take values between 0 and 1. This measure makes it possible to compare the values of several models with each other. A model that doesn't explain the classification at all would have a AUC of 0.5. The higher the AUC value, the better the model. A perfect model would, therefore, have an AUC value of 1 (cf. James et al., 2013).
Finally, the whole procedure within the data simulation, boosting and calculation of the main parameter is repeated 100 times.

4.2 Results

In the following section, the results of the two methods are presented and compared. First, the results of the classical linear boosted regression model based on the simulated data are presented, subsequently, the results of the logistic boosted regression. In the results of the simulated data mainly the selected variables are compared since it is known which of the variables have an influence. In addition, the values of the estimated influence are considered. Finally, to measure the prediction accuracy, the MSE/MSEP or AUC are calculated for both methods, too.

4.2.1 Linear Regression

The aim of the one SE rule chosen stop point is that as many important variables are selected as by the conventional method, but less of the non-informative ones.

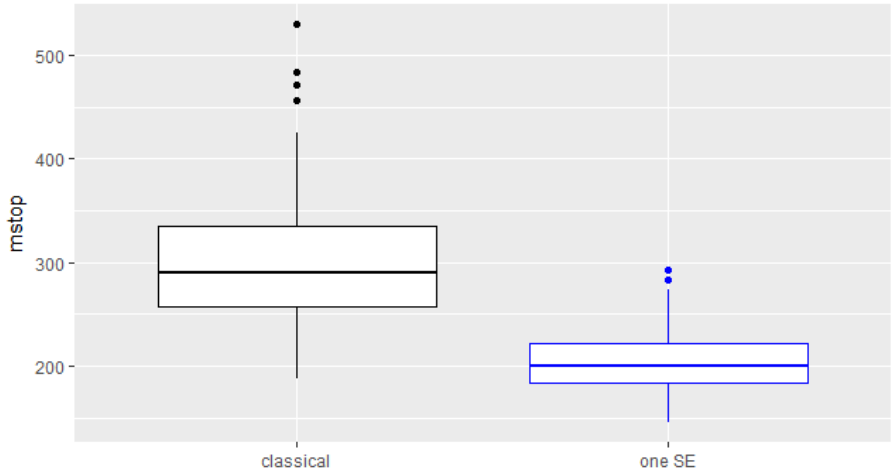


Figure 4: Comparison of the stop points for the boosting algorithm used for a linear regression, once selected via the classical method and once over the newly applied one SE rule applied on simulated data.

Overall, the application of the one SE usually selects a stop point which on average is earlier than that of the conventional method, shown in Figure 4. A small stop point causes fewer variables to be taken (Mayr and Hofner, 2018). It is important, that nevertheless all the important variables are identified and only the number of non-informative variables decreases. At the beginning the variables were compared with each other, that were simulated with an influence.
Both methods achieve the same result here. In each of the 100 cases, all of the six important variables are selected, by both chosen stop points. With regard to the important variables, the one SE rule achieves just as good results as the method used so far. If now the number of selected non-informative variables is considered, it is an advantage if it is as small as possible. In the data simulation, these were created without influence, so it is known that they should not be selected. The following graphic shows the number of selected unimportant variables of both methods.

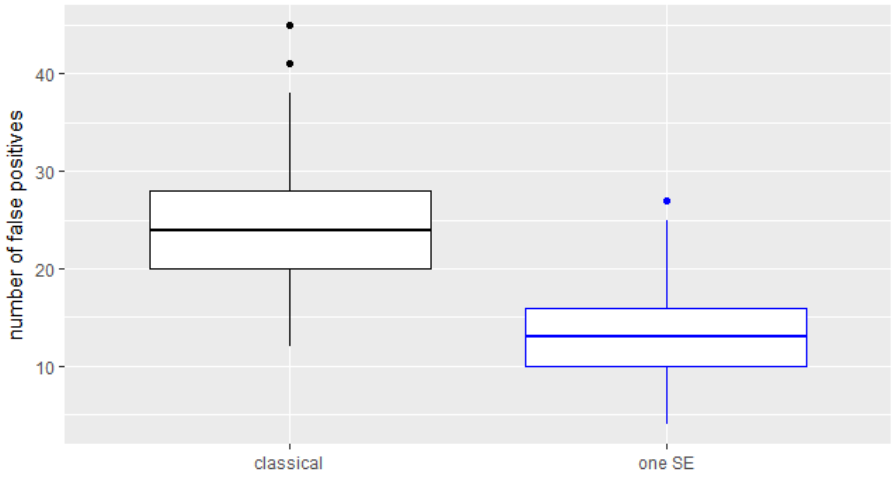


Figure 5: Compared here are two boosted linear models via simulated data and the stop point of the algorithms once chosen by the classical method and once over the newly developed one SE rule. Considering the number of selected non-informative variables, thus the number of selected variables that were simulated without influence (false positives).

It turns out that the stop point chosen by the one SE rule selects less of the non-informative variables. In 50% of cases, the model resulting by choosing the stop point over the one SE rule will only pick up 13 or less non-informative variables. By the method used so far, the median is 24, so nearly twice as many non-informative variables are included.

Using the new method, only ten out of 100 cases pick up more than 19 non-informative variables. In contrast, the method used so far records more than 19 of the non-informative variables in 80 of the 100 cases.

Overall, by choosing the stop point over the one SE rule, no important information are lost in any of the 100 cases and in addition, the number of non-informative variables recorded is lower. In this particular example, the use of the one SE rule seems to lead to considerably better results. In addition to the selection of the right variables, the estimated effect of these must also be taken into account. Now the estimated values of β are considered. As before, the simulation of the data specifies which value the β should ideally assume.

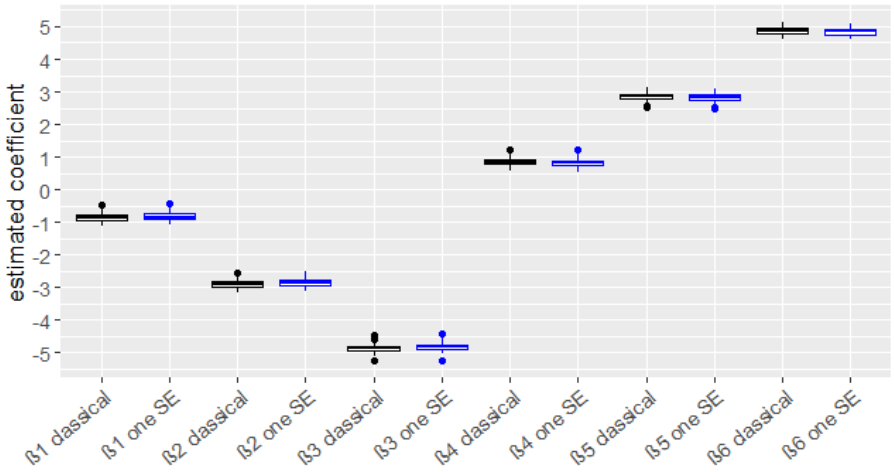


Figure 6: Comparing the estimated coefficients through the boosted linear model created by the classical choice of the stop point as well as through the choice of the one SE rule. The boosting algorithm was applied on simulated data. Contemplated are the variables that were created in the simulation with an impact on the outcome.

Both models, resulting by using the previous method and by using the one SE rule, almost reached the true values for β. Also, the variance is very low. However, it can be seen that the values are minimally underestimated when using the one SE rule when compared with the results of the previously used method. To compare the mean estimated effects for one non-informative variable, the absolute value of the coefficient for the selected non-informative variables are added up. Out of it the mean estimated effect of one variable was calculated. Figure 7 shows the results.

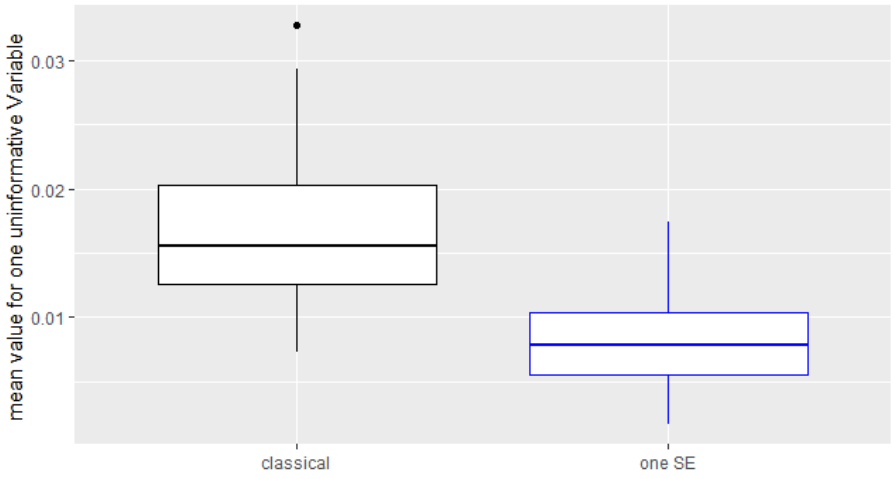


Figure 7: Comparison of two boosted linear models using simulated data. The stop point of the algorithm was once chosen by the classical method and once by the one SE rule. Here the mean estimated coefficients for one variable, that where created without influence on the outcome, are compared. Therefore the means of the added absolute values of the coefficients are calculated.

The use of the one SE rule leads to a substantially lower average estimated effect for non-informative variables. Using the one SE rule, the sum of the estimated effects of the non-informative variables is below approximately 0.01 in 75% of the cases. Using the previous method, only 25% of the cases show a number below 0.01.

After considering the selected variables and their estimated effects, the prediction accuracy of both models is contemplated. In order to compare the prediction quality of the models, both the model originated by the original method and the one SE rule are applied to the training dataset and the MSE is calculated.

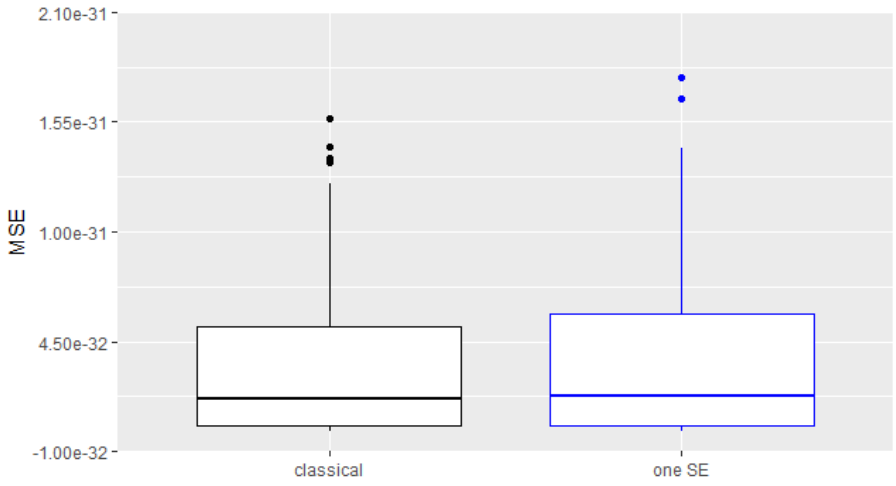


Figure 8: Comparing two boosted linear models, once selected by the classical method once by the one SE rule, using simulated data. The boxplots refer to the calculated MSE for both models when applying to the training data.

For both methods the calculated MSE is very low, also the median is very similar. When using the one SE rule, the median is only around $0.2 * 10^{-32}$ higher. Looking at the MSEP, which reflects the prediction accuracy of the models on the test data, more differences can be seen.

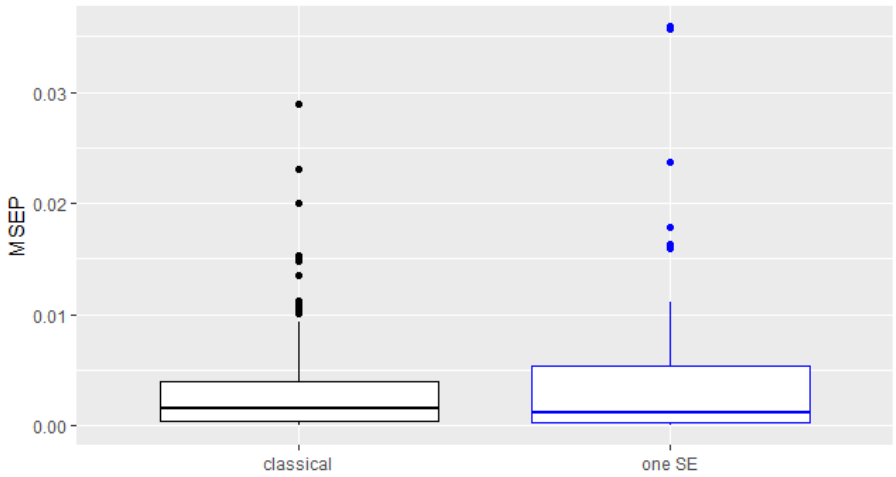


Figure 9: Comparison of the MSEP values of two boosted linear models. Therefore simulated data are used and the stop point of the algorithm was once chosen by the classical method and once over the newly applied one SE rule. The MSE was calculated for both models on simulated test data.

The values for the MSEP are also very low for both methods. The upper quantile of the MSEP is slightly higher when using the one SE rule, but the median is lower. Using the one SE rule the MSEP is lower than 0.0011 in 50% of the cases, as compared with the classical boosted model, where the MSEP is lower than 0.0015 in 50% of the cases.

4.2.2 Logistic Regression

As expected, we also chose a much lower stop point in logistic regression as a result of the one SE rule. So fewer variables are included in the model:

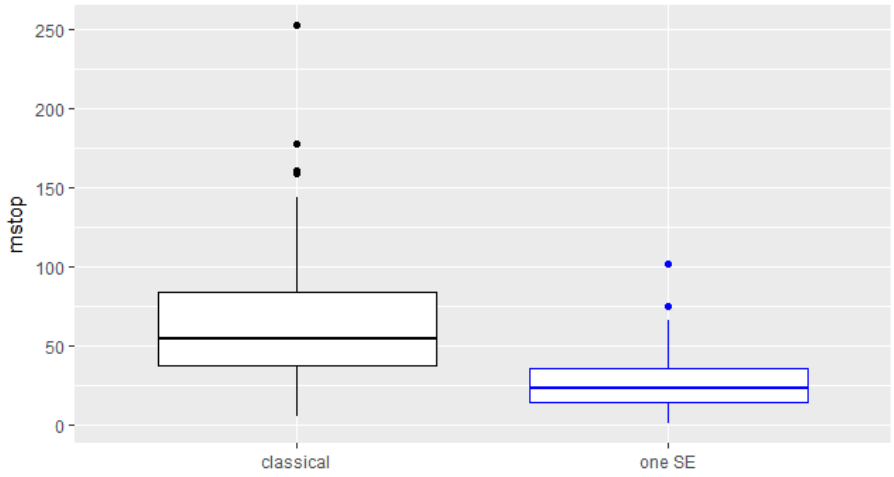


Figure 10: Comparison of the stop points for the boosting algorithm used for a logistic regression, once selected via the classical method and once over the newly applied one SE rule applied on simulated data.

As before it is important to reduce only the number of non-informative selected variables. When considering the six variables that were created with an influence on the outcome, it is striking that in most cases not all of the important variables are selected by both methods:

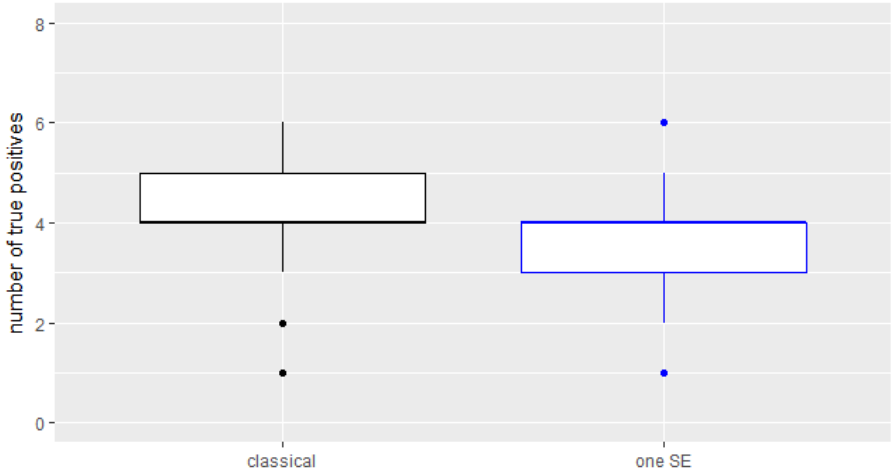


Figure 11: Compared are the selected variables of two boosted logistic models at simulated data. The stop point of the algorithms was once chosen by the classical method and once over the newly applied one SE rule. Boxplots refer to the number of selected variables that were created with influence on the outcome (true positives).

The median for both boosted models, created by the two methods to select the stop point, is 4. But the method used so far tends upwards and the one SE rule downwards. The upper quantile of the model resulting in order from the classical method is 5, for the model choosing the one SE rule it is only 4. Nevertheless, it can be said that the methods achieve comparable results considering absolute frequencies. Table 1 shows a table of the absolute frequencies.

Number of selected informative variables	absolute frequencies of the	
	classical method	one SE rule
1	1	4
2	2	13
3	19	32
4	48	39
5	26	11
6	4	1
Total	100	100

Table 1: Comparison of two boosted logistic models, once using the classical method, once the newly developed one SE rule. Underlying are simulated data. Contemplated are the absolute frequencies of the selected variables that were simulated with an influence on the outcome.

The one SE rule thus selects almost as many of the important variables as the classical method. When considering the selected non-informative variable, however, there are great differences. Figure 11 shows that noticeable fewer non-informative variables are selected by the one SE rule.

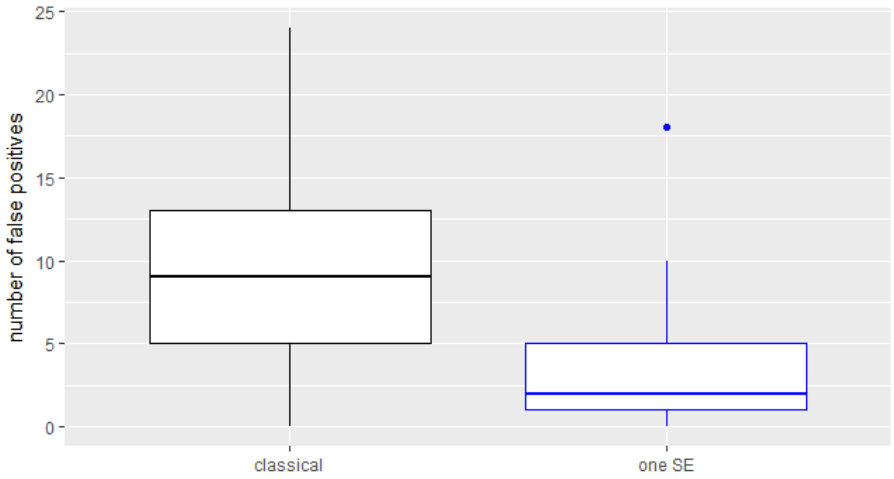


Figure 12: Compared here are two boosted logistic models. Therefore simulated data are used and the stop point of the algorithms was once chosen by the classical method and once over the newly developed one SE rule. The boxplots refer to the number of selected non-informative variables, this means the number of selected variables that were simulated without influence (false positives).

Nine non-informative variables are selected by the method used so far in the median, in contrast only two are selected by the one SE rule. The one SE rule selects five or fewer variables in 75 % of the cases, compared to only 25% of the cases by the method used so far.

Besides the selected variables, the estimated coefficients of the variables are also important. It starts again with the informative variables and their estimated effects. Figure 14 shows the estimated values of both models.

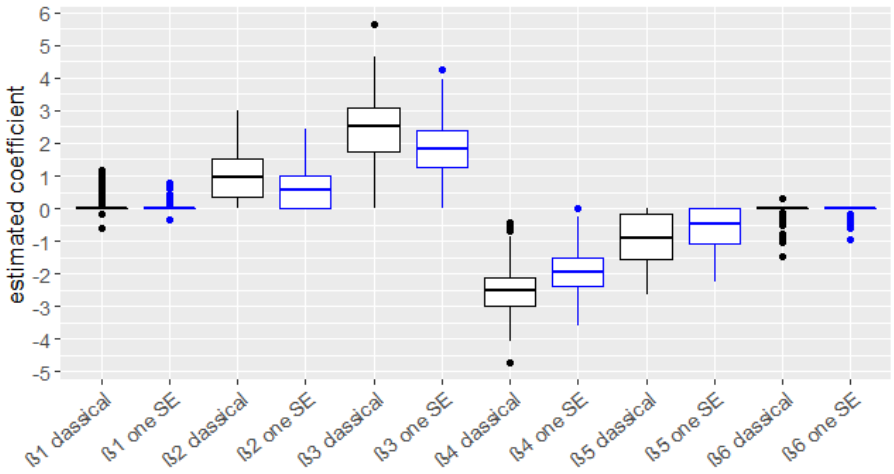


Figure 13: Comparing the estimated coefficients through the boosted logistic model created by the classical choice of the stop point as well as through the choice of the one SE rule. The boosting algorithm was applied on simulated data. Contemplated are the variables that were created in the simulation with an impact on the outcome.

It can be clearly seen that both methods do not reach the modelled values of -5, -3, -1, 1, 3, 5. Also, the estimated coefficients of the model of the one SE rule are slightly apart from the actual values. In addition, a dispersion of the values can be seen. Moreover, the values are more underestimated by the one SE rule than by the classical method. Comparing the mean estimated coefficient of an non-informative variable, in turn, the one SE method yields better results, shown in Figure 14.
The mean estimated effects for one non-informative variable was, like the linear regression, calculated as the average of the summed absolute value of the estimated coefficient for the selected non-informative variables of the models. The median of the mean estimated coefficient of a non-informative variable of the classical method is around 0.0353. By using the one SE rule, however, only 0.0047. In 75% of the cases, the model resulting by the one SE rule estimates a mean coefficient for one variable of 0.0092 or less. Choosing the classical method, this value is only achieved in less than 25% of the cases.

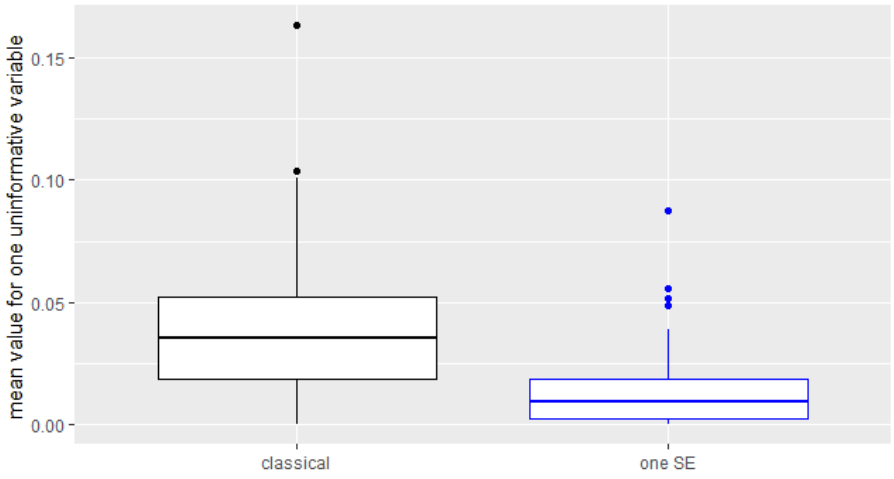


Figure 14: Comparison of two boosted logistic models using simulated data. The stop point of the algorithm was once chosen by the classical method and once by the one SE rule. Here the mean estimated coefficient for one variable, that where created without influence on the outcome, are compared. Therefore, the means of the added absolute values of the coefficients are calculated.

In addition to the variables selected, the AUC was also considered as a measure of prediction accuracy. First, the two boosted models were applied to the training data and then the AUC was calculated. Figure 16 presents the results.

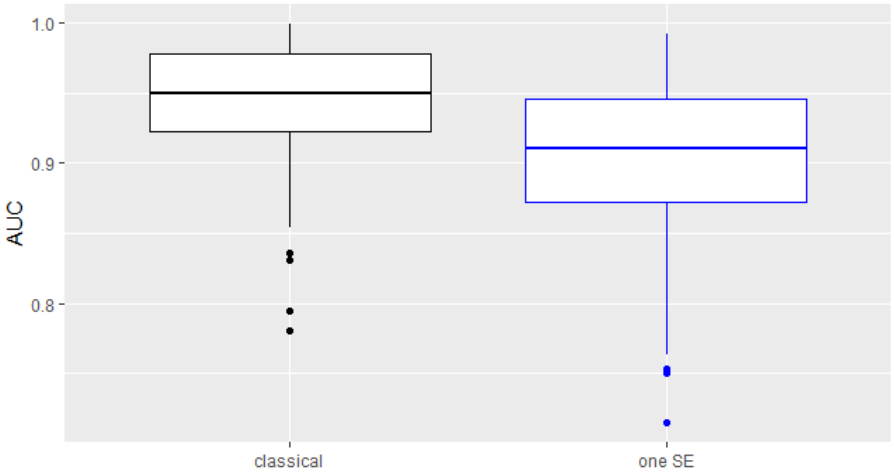


Figure 15: Comparing two boosted logistic models, once selected by the classical method, once by the one SE rule, using simulated data. Considering the calculated AUC for both models, when applying to the training data.

The training data shows that the boosted model by choosing the classic stop point achieves higher and therefore better values as the resulting model when applying the one SE rule. In 50% of the events, the AUC is higher than 0.949 when using the classical method. Using the one SE rule the AUC is only higher than 0.910 in 50% of the events. The worst result was also achieved by the one SE rule with 0.715, by contrast, the worst result of the classical method is only at 0.780. Other results are apparent when the models are fitted on the test data.

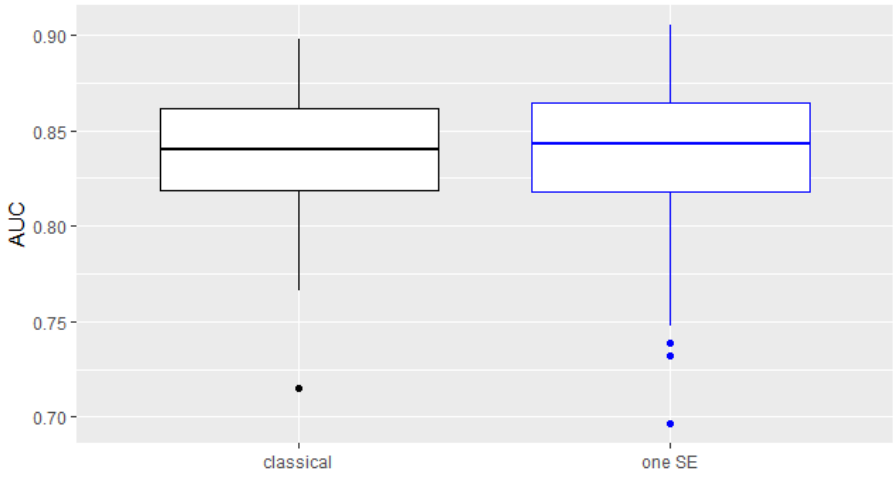


Figure 16: Comparison of the AUC of two boosted logistic models. Therefore simulated data are used and the stop point of the algorithms was once chosen by the classical method and once over the newly applied one SE rule. The AUC was calculated for both models on simulated test data and therefore refers to prediction accuracy.

Here, the AUC values of both models are compared again. The resulting box plots are much more similar here, but the scatter of the results is slightly higher when using the one SE rule. Looking at the median, the model obtained by using the one SE rule, with a value of 0.843, is a little higher than 0.840, which corresponds to the median of the model created by the classical choice of the stop point. However, the worst value is reached again when the one SE rule is applied and is at 0.696. Comparing to the worst AUC produced using the classical method with only 0.715. Taken together, the results of both methods are very similar in this case. The one SE rule on average selects less of the non-informative variables, but also less of the informative. Nevertheless, if the resulting model is applied to the test data, it produces at least as good results as the use of the classically selected stop point.

5 Application

In this Section, the two methods are now applied to high-dimensional genomic data. First, the linear model is treated again, then the logit model is considered. For both records, a leave-one-out cross-validation is performed to effectively subdivide the existing data into test and training data. For this purpose, an observation is taken from a data set with N observations. The remaining
N-1 observations are then used as a training record, the one which is taken out for testing. This process is then repeated until each of the N observations had been taken once and used for testing (cf. Fahrmeir et al., 2013). The resulting models from the classic choice of stop point as well as the one SE rule are now fitted to the training data and applied on the test data. The values of the stop point, the number of selected variables and the respective value for measuring the prediction accuracy are considered.

5.1 Linear Regression

For the application of the linear models the high-dimensional dataset *riboflavin* is used, which is contained in the R packet '*hdi*'. The outcome of this dataset is the logarithm production rate of riboflavin with Bacillus subtilis. It includes expression level for $p = 4088$ genes of $n = 71$ observations (cf. Bühlmann et al., 2014).
These data are split into training and test data using a leave-one-out cross-validation. On the training data now both models, resulting by the previously used method and by the one SE rule, are fitted and the number of selected variables plus the stop points are stored. The models were then applied to the test data set, which is only one taken out observation, and the square error of the prediction (SEP) was calculated. This procedure was repeated 71 times until the leave-one-out cross-validation was finished. From the obtained 71 values of the squared errors, the mean squared error of prediction (MSEP) was calculated. The following graphic shows the differences in the two methods.

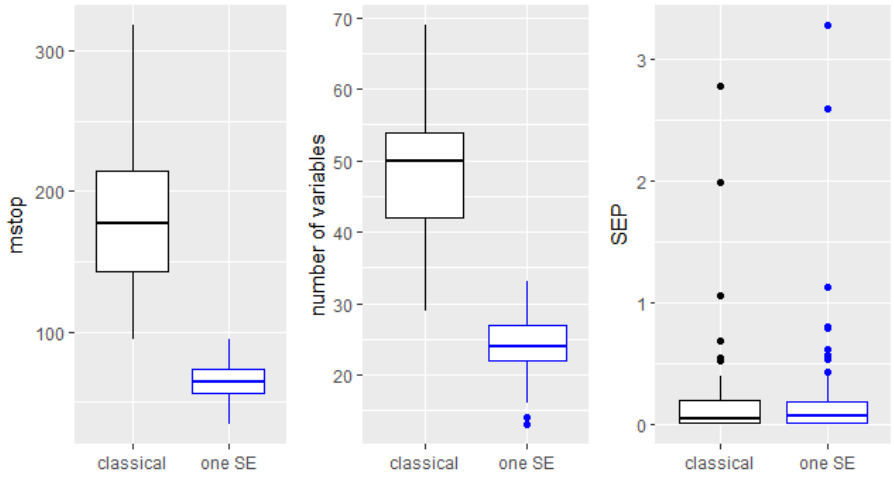


Figure 17: Comparison of two boosted linear models using a data set of the logarithm production rate of riboflavin. The stop point of the algorithm was chosen by the classical method and by the one SE rule. Here the referring boxplots for both methods of the stop points, the number of selected variables and the SEP are shown.

It can be clearly seen that the selected stop point is considerably lower when choosing the one SE rule than over the classical method. As a result, the number of selected variables is significantly lower due to the new one SE rule. The boxplots of squared errors are very similar in both methods. Considering the median, the number of iterated runs is reduced by more than two thirds when applying the one SE rule, as can be seen in Table 2.

Method	median m_{stop}	median number of selected variables	MSEP
normal	177	50	0.204078
one SE	64	24	0.2357279

Table 2: Comparison of two boosted linear models, once using the classical method, once the newly developed one SE rule, using a data set of the logarithm production rate of riboflavin. Displayed are the median values of both models of the stop point, the number of selected variables as well as the MSEP.

Due to the one SE rule, only 24 variables are selected, and thus only half of the number selected by the previous method. The MSEP of both methods is very low, whereas that of the one SE rule is slightly higher. Overall, both methods provide good results, yet the model resulting from the classical choice of the stop

point achieves a slightly lower and thus better value of the mean squared error of prediction.

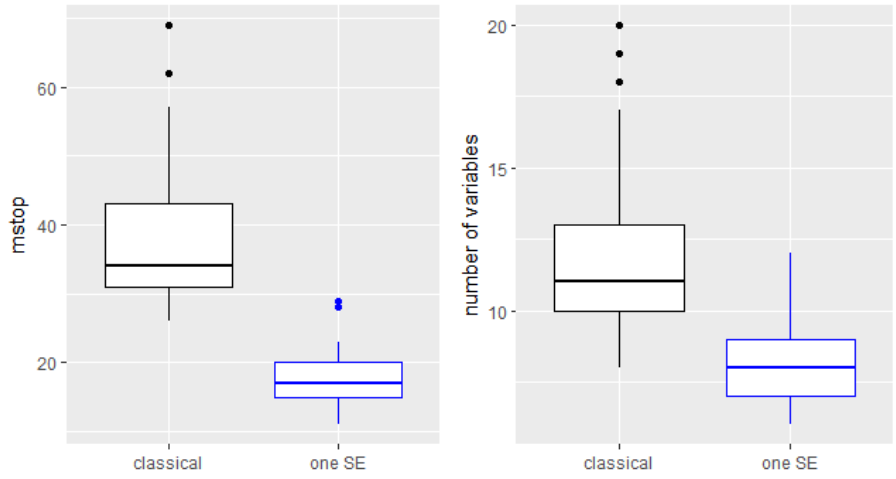
5.2 Logistic Regression

In order to compare the two methods for selecting the stop point in the logistic regression, the data set *'alon'* from the package *'datamicroarray'* was used. It is a high-dimensional data set of colon cancer collected by oligonucleotide arrays. It contains 40 tumor and 22 tumor-free colon tissue observations of the outcome and $p = 2000$ genes as possible influencing variables (cf. Alon et al., 1999). As before, the allocation in training and test data set takes place by means of a leave-one-out cross-validation. Now the boosting algorithm is applied on the training data and two models are created by using the classical choice of the stop point as well as using the one SE rule. The selected stop points as well as the number of selected variables of both models are now determined. Afterwards, the models are applied to the one exception observation that serves as the test dataset and the predicted probability for the occurrence of an event of both models is stored. After finishing the cross-validation, 62 predicted probabilities are obtained, from which a ROC curve was formed and the AUC was calculated. The results are presented in Table 3

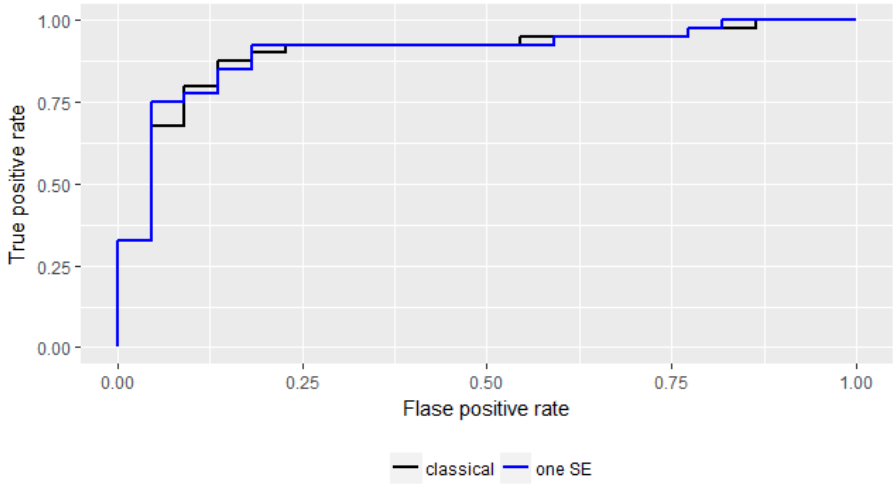
Method	median m_{stop}	median number of selected variables	AUC
normal	34	11	0.8977273
one SE	17	8	0.9

Table 3: Comparison of two boosted linear Models, once using the classical method, once the newly developed one SE rule. Underlying is clone cancer data set. Displayed are the median values of both models of the stop point and the number of selected variables as well as the AUC.

By using the one SE rule, the stop point in the median is reduced by half in comparison to the classical method, which results in a median 34 iteration. Therefore, also the median number of selected variables is lower when using the one SE rule. Nevertheless, the AUC of the model resulting by using the one SE rule is slightly higher than by using the classical method. The exact values, as well as the ROC curves of the models, can be seen in Figure 18.



(a)



(b)

Figure 18: Comparison of two boosted logistic models using a colon cancer data set. The stop point of the algorithm was chosen by the classical method and by the one SE rule. Here the referring boxplots for both methods of the stop points and number of selected variables (a) as well as the ROC curve (b) are shown.

In this application example, the newly applied one SE rule undergoes less iteration and therefore fewer variables are selected. Nevertheless, the AUC values of the associated models are very similar and high, suggesting that both models provide good predictions. However, the use of the one SE rule still leads to a minimal improvement of the prediction accuracy.

6 Discussion

For high-dimensional data sets, an analysis using classical regression approaches is no longer possible. One possibility for modeling such large data sets is the boosting algorithm, which selects the available variables in an iteration process already during the modeling process (cf. Mayr and Hofner, 2018). The resulting models can be controlled by only one main tuning parameter, the stop point of the algorithm (Mayr et al., 2012). In this work, an optimization of the stopping rule by using the one standard error rule was presented. For this purpose, the classic criterion for selecting the stop point was extended by the one SE rule. The results of the two methods were compared using two model types, a classical linear regression and a logistic regression. For this purpose, suitable data were first simulated. Since it is known which variables were used to define the outcome, it is possible to compare exactly how many of the informative and non-informative variables are selected by the two methods and which coefficients they estimate for them. In addition, the prediction accuracy of the two models was compared. Afterwards, the boosting algorithm was also applied on real data sets.

Starting with the modeling of linear regression on simulated data, using once the classical choice of the stop point and once the one SE rule, showed that the occured models, selected the informative variables equally effective. Both methods selected all informative variables. Differences existed in the selection of non-informative variables, where the one SE rule achieved better results. In estimating the coefficients for the informative variables, both methods approximated the modelled coefficients. The one SE rule estimated the coefficients slightly weaker than the classical method but by comparing the mean estimated coefficient of a non-informative variable, the one SE rule was able to produce clearly better results. When considering the mean squared error of prediction, both methods achieved similarly good results, yet the median of the MSEP was slightly lower for the one SE rule. Nevertheless, the MSEP value of the model was slightly better when using the one SE rule. In addition, the linear model was fitted to the *riboflavin* dataset, which contains information about the production rate of riboflavin. In this case, the linear boosted model, where the stop point was chosen by the one SE rule, did not achieve a better MSEP value. The value of the MSEP was slightly lower here for the classical method.

When modeling a logistic regression model using the boosting algorithm, it turned out that by using the one SE rule, fewer of the non-informative variables were selected. However, in the selection of informative variables, the newly developed method gave a slightly worse result compared to the model that emerged from the classical choice of the regression model. In the median, both methods selected the

same number of variables, yet the number tends to go upwards if we use the classical method and lower if we use the one SE rule. It also shows that the estimated coefficients of the informative variables are better when using the classical method for selecting the stop point. The mean estimated coefficient of a non-informative variable, on the other hand, again shows better results for the one SE rule. As a measure of the prediction accuracy, the ROC curve and the associated AUC value were chosen for the logistic regression. Here, a slightly higher and hence better AUC value was found for the one SE rule, but the value also gave a greater variance than when using the classical method. To apply the boosting algorithm to a real dataset, a survey was conducted on the topic of colon cancer. As before, the model resulting from the one SE rule, achieved a slightly better AUC value.

Overall, the newly developed one SE rule, with the exception of the application on the *riboflavin* dataset, gives better results when comparing prediction accuracy. The result for the *riboflavin* dataset could be due to the specific dataset and should be checked against other examples. However, it must also be mentioned that this improvement is only minor, by using the one SE rule.
The weakness of using the one SE rule appears in complex models where less of the informative variables are selected and thus relevant information for modeling can be lost. In addition, a rather simple modeling on the *riboflavin* dataset did not result in any improvement in prediction accuracy, but it has to say that the deterioration in prediction accuracy was only small. And in addition to improving the prediction accuracy, the new stopping rule offers yet another clear strength: the fact that much less of the non-informative variables are skewed and the mean estimated coefficient for non-informative variables are far smaller. This can prevent overfitting the model, what would mean that it follows the errors too much, and in addition, the shrinkage of effect estimates leads to a smaller variance which in turn leads to greater stability (cf. James et al., 2013).

In general, the use of the one SE rule seems to lead to more stable and better models, it should be considered how the one SE rule compares the results of other and more complex model types, such as: the GAM. Also, the consideration of the time expenditure must not be neglected. Since the one SE rule starts at the point at which the classical method stops, it is to be expected that the time and the required computing capacities will increase. This should be considered in further investigation.

7 References

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12):6745–6750.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and Regression Trees*. CRC Press, Boca Raton, 1st ed. edition.

Bühlmann, P. and Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505.

Bühlmann, P., Kalisch, M., and Meier, L. (2014). High-dimensional statistics with a view toward applications in biology. *Annual Review of Statistics and Its Application*, 1(1):255–278.

Fahrmeir, L., Kneib, T., Lang, S., and Marx, B. (2013). *Regression: Models, Methods and Applications*. Springer, Dordrecht.

Freund, Y. (1995). Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232.

Hepp, T., Schmid, M., Gefeller, O., Waldmann, E., and Mayr, A. (2016). Approaches to regularized regression - a comparison between gradient boosting and the lasso. *Methods of information in medicine*, 55(5):422–430.

Hofner, B., Mayr, A., Robinzonov, N., and Schmid, M. (2014). Model-based boosting in r: a hands-on tutorial using the r package mboost. *Computational Statistics*, 29(1-2):3–35.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning*, volume 103. Springer New York, New York, NY.

LeBlanc, M. and Tibshirani, R. (1998). Monotone shrinkage of trees. *Journal of Computational and Graphical Statistics*, 7(4):417–433.

Mayr, A. and Hofner, B. (2018). Boosting for statistical modelling-a non-technical introduction. *Statistical Modelling: An International Journal*, 71:1471082X1774808.

Mayr, A., Hofner, B., and Schmid, M. (2012). The importance of knowing when to stop. a sequential stopping rule for component-wise gradient boosting. *Methods of information in medicine*, 51(2):178–186.

Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics (Oxford, England)*, 23(19):2507–2517.

Tibshirani, R. (1996). Regression shrinkage selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Wilcox, R. R. (2010). *Fundamentals of Modern Statistical Methods: Substantially Improving Power and Accuracy*. Springer Science+Business Media LLC, New York, NY, 2. ed. edition.

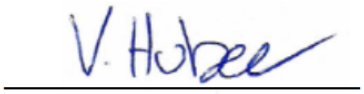
8 List of Figures

9 List of Tables

A Declaration on Oath

I declare that I have written the bachelor thesis independently and without outside help, that I have not used any sources other than those given, and have identified the passages taken from the sources used as such. This term paper has not been presented in any other course in this or any similar form.

Munich, 22 June 2018



Veronika Huber

B RCode

The work is accompanied by a CD on which the entire RCode for all simulations, models and graphics can be found.