Analyzing indonesian rice farms

Jonas Kernebeck, Alexander Flick, Felix Lehner

07/16/2021

Contents

1 Introduction

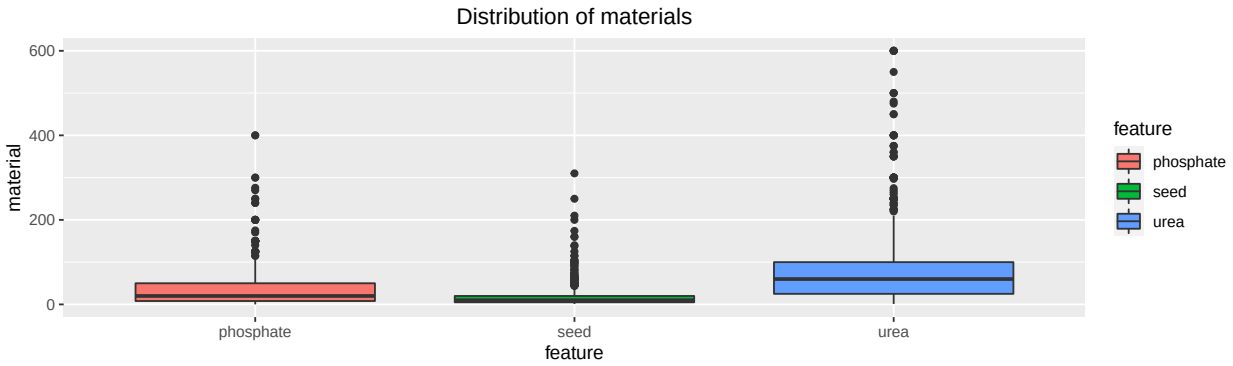
The present data set includes production data for 171 indonesian rice farms. The dataframe contains the following variables:

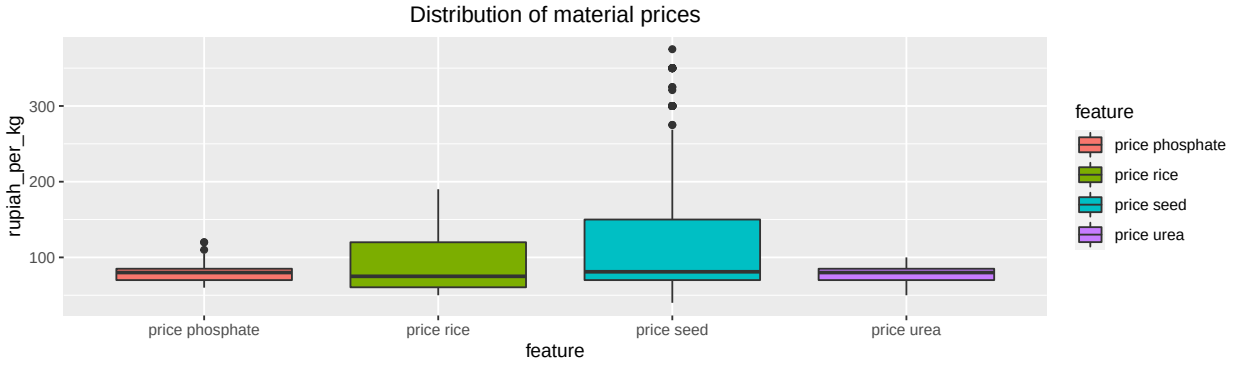
variable	description	expressions
id	unique identifier for a farm	unique id
size	total production area in hectares	0.01 - 5.322
status	status of property rights	"owner", "share", "mixed"
varieties	rice seed varietes	"trad", "high", "mixed"
bimas	bimas-status of the farmers	"no", "yes", "mixed"
seed	seed in kilogram	1 - 1250 kg
urea	urea in kilogram	1 - 1250 kg
phosphate	phosphate in kilogram	0 - 700 kg
pesticide	pesticide cost in Rupiah	0 - 62600 r
pseed	price of seed in Rupiah per kg	40 - 375 r/kg
purea	price of urea in Rupiah per kg	50 - 100 r/kg
pphosph	price of phosphate in Rupiah per kg	60 - 120 r/kg
hiredlabor	hired labor in hours	1 - 4536 h
famlabor	family labor in hours	1 - 1526 h
totlabor	total labor (excluding harvest labor)	1 - 4774 h
wage	labor wage in Rupiah per hour	30 - 175.35 r/h
goutput	gross output of rice in kg	42 - 20960 kg
noutput	gross output minus harvesting cost	42 - 17610 kg
price	price of rough rice in Rupiah per kg	50 - 190 r/kg
region	region of the farm	unique region

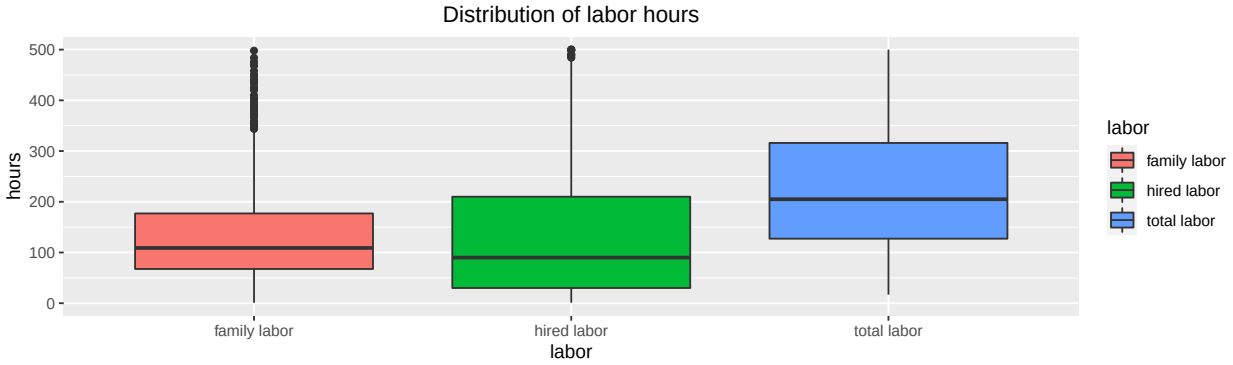
As present in the table, the data set consists of 16 numeric variables and 4 categorical variables. The target variable for the regression modeling will be *goutput*, what represents the gross output of rice in *kg* for the respective rice farm. In the following some explorative data analysis will be made to get to get a first impression of the distribution of the individual variables.

1.1 Numerical Variables

The following figure shows boxplots for the used materials and the prices paid for the materials of the respective rice farms. The boxplots for the materials show, that the distribution of all materials is right-skewed. The spread width of seed is the lowest, followed by phosphate and urea. Therefore *urea* also has the highest variance with 16166 followed by *phosphate* with 2264 and *seed* with 2048. The distribution of *urea* indicates that rice farms in Indonesia may use urea very different, caused by e.g. the bimas-status. The bimas program is a rice intensification program by the government to support local rice production by providing high-yield rice seeds as well as technical assistance. If we look at the prices for phosphate *pphosph* and urea *purea*, we can see a slight left-skewed distribution with low variance (75 for *purea* and 86 for *pphosph*). In contrast to that, the prices for seeds scatter much. The distribution of *pseed* is strongly right-skewed as well as the distribution for the rice price *price*. The price for the rice also scatters, but less than *pseed*. The two prices have a correlation of 0.67. Of course, the price of seeds affects the selling price of rice. The prices may fluctuate due to seasonal or regional factors and have an impact on each other. The distribution of labor hours is also slightly skewed to the right. Overall, the dispersion is lowest for the *famlabor*. For *hiredlabor* and *totlabor* we have a similar spread, but *totlabor* has a higher level overall. This is caused by the *hiredlabor* which is a subset of *totlabor*.

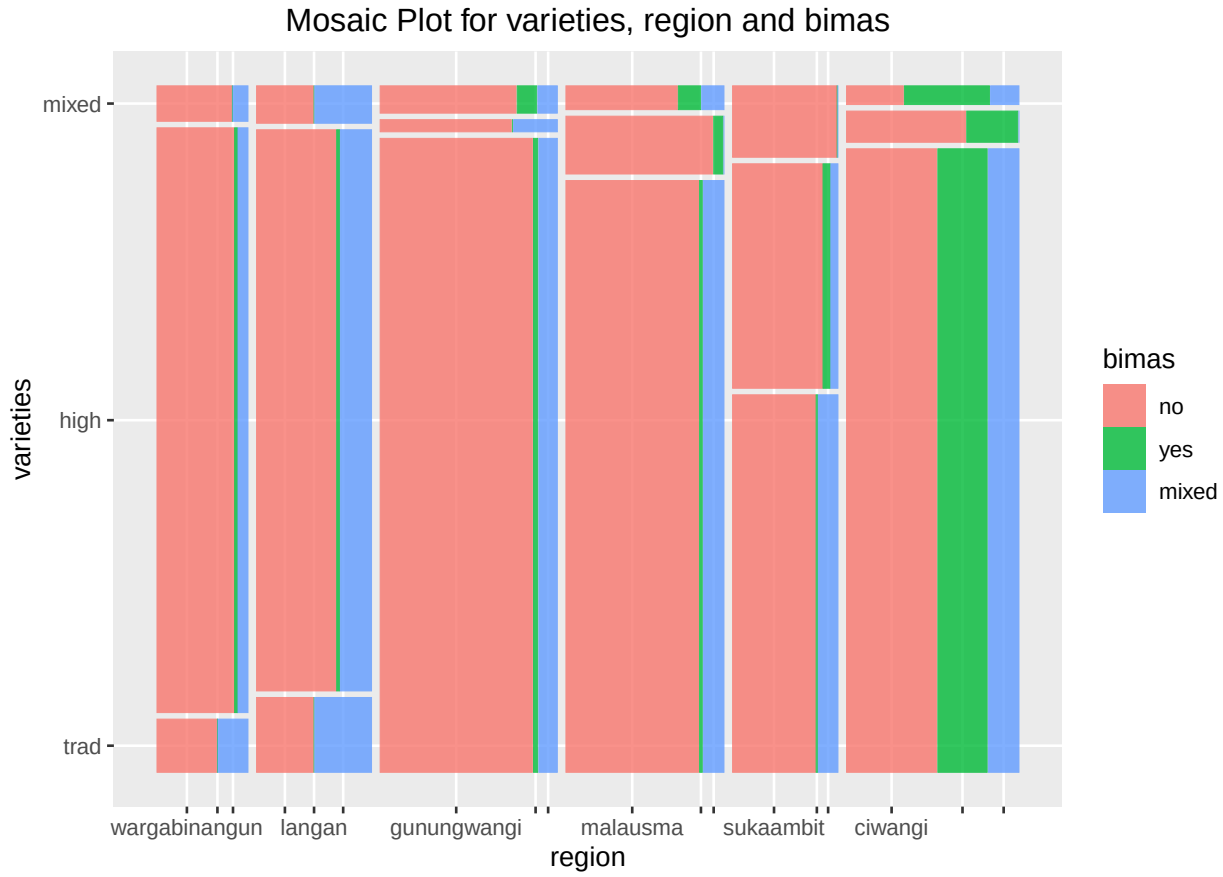

Distribution of materials
material
600
400
200
0
phosphate
seed
urea
feature
feature
phosphate
seed
urea


Distribution of material prices
rupiah_per_kg
300
200
100
price phosphate
price rice
price seed
price urea
feature
feature
price phosphate
price rice
price seed
price urea


Distribution of labor hours
hours
500
400
300
200
100
0
family labor
hired labor
total labor
labor
labor
family labor
hired labor
total labor

1.1 Categorical Variables

The following mosaic plot shows the distribution of of the categorical variables *varietes*, *region* and *bimas*. Overall, all regions are roughly equally represented in the data set. We can detect, that most of the farmers with the bimas status *yes* and *mixed* are located in the region *ciwangi*. The distribution of the different varieties is strongly dependent on the region. While the *high* varieties have the biggest share in the regions *wargabinangun* and *langan*, the *traditional* varieties are dominating the regions *gunungwangi*, *malausma* and *ciwangi*. The *mixed* varities are only used slightly in all regions.



To test wheter the categorical variables have impact on our target variable *goutput*, one- and two-sided anovas are performed. The results of these are summarized in the following table:

formula	F-value	p-value	significant
region	22.981	$<$ 2e-16	yes
varieties	11.764	8.94e-06	yes
bimas	14.817	4.57e-07	yes

formula	F-value	p-value	significant
region+varietes	3.847	3.96e-05	yes
region+bimas	5.651	2.94e-08	yes
varieties+bimas	0.791	0.531	no
region+varieties+bimas	0.860	0.580	no

The anova outputs show, that all of the categorical variables have a significant effect on *goutput*. The null hypothesis, that the mean of *goutput* is the same across the groups is rejected. The results of the two-sided anovas also show a significant interaction effect on *goutput*. While the interaction effect from the *region* with *varieties* and *bimas* is significant, the interaction effect of *varieties* and *bimas* and the interaction effect of all three variables is not.

1.3 Variable selection and transformation

The performance of the regression modeling is highly dependent of the variable selection and transformation. Therefore a suitable choice is very important. The variable *noutput* is a linear transformation of *goutput* as it represents *goutput* decreased by the harvesting costs. Therefore it is not used for the modeling because it would violate the multicollinearity assumption.
The variable *size* also correlates *strongly* with the target variable. This can be intuitively explained by the fact that a larger rice field naturally always produces a higher yield. Since the variables *seed*, *urea*, *phosphate* and *pesticide* are dependent on size, they are transformed into per-hectare sizes by dividing them with the respective hectare size of the farm. The *size* variable is not used for further modeling.
The variables *famlabor* and *hiredlabor* are subsets of the variable *totlabor* and are therefore transformed into the share of *totlabor* by dividing them with the amount of *totlabor*. The variable *totlabor* is after that transformed to a per-hectare size by dividing it with the *size*. The variable *wage* follows a bimodal distribution. Therefore it is transformed into a binary variable, which indicates if the respective value is over or under 100.

1.4 Model evaluation

The data set will be splitted in 60-20-20 parts, where 60% of the data is used for training the model, and 20% for testing and validating respectively. In the modeling part, also cross-validation is used. To evaluate the models and compare them, different metrics will be used. The numeric metrics used are the *MSE*, which stands for the mean squared error and the *AIC*, which stands for the Akaike information criterion. Beside these metrics, also graphical analysis plots like a residual plots are used for evaluation.

3 Outliers most likely due to typos for observation 110, 947 and 1004

2 First Model

The first model to present and evaluate is called Lasso Regression.

2.1 Lasso Regression

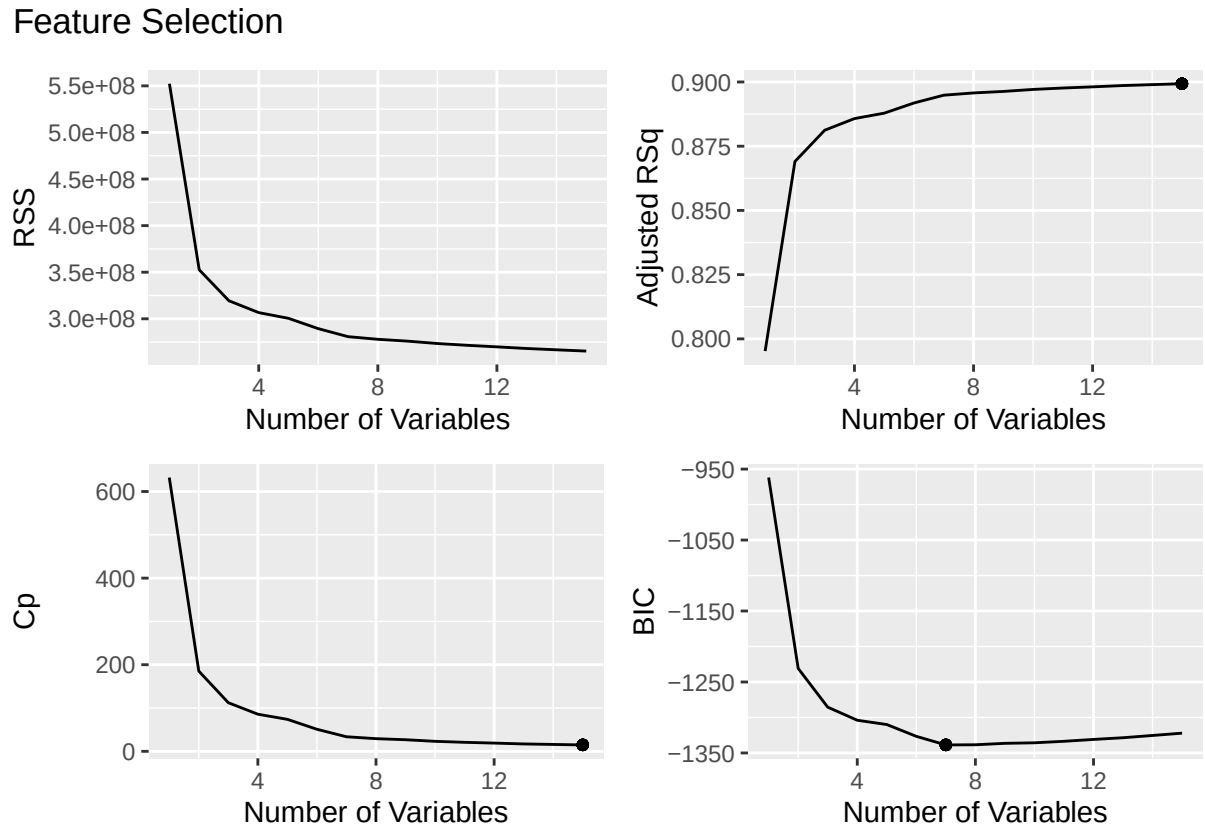
As in most of the regression types, it minimizes the residual sum of squares (short: RSS). But in addition to that a penalty term is included in the formula which shrinks some parameters coefficient estimates to zero.

$$RSS + \lambda \sum_{j=1}^{p} \left| \widehat{\beta}_j \right|$$

,where λ is a hyperparameter, p are the parameters and β the parameter coefficient estimates. The sum of the coefficents starts at $j = 1$ because the β_0 is no parameter coefficient estimate, it is the bias of the model. Furthermore the absolute values of the estimates are calculated and summarized. As you can see from the equation, the lasso regression turns into a simple linear regression if the λ is zero.
The goal and intention of the lasso regression is to create a sparse model which makes it easier to interpret.

2.2 Feature Selection

Before starting with the actual regression, we can investigate which of the features could be important for predicting the goutput of the data. This step is helpful as the number of features in the dataset is over twenty.
The method used for selecting the variables is the forward selection. The package leaps provides therefore a function regsubsets for linear models. It takes the target variable and the features of the dataset as input and gives some hints which variables could fit best to predict the goutput. The used algorithm optimizes the Mallows C_p statistics which is related to the AIC. (James,p.79)
There are more input parameters available for the function, i.e. the maximum size of a subset, the weight vector, the number of the best subsets or the method of variable selection (i.e. forward selection, backward selection, etc.).
The following graphics show the results of the analysis.



Explanation: The graphic show the the RSS, the adjusted R^2, the C_p statistics and the BIC of the models. The metrics help to identify the overall best models of the problem. Each of the metrics show that in general a 3-variable model could be enough as the metrics are getting slightly better as the number of variables increase. Just for the BIC it can be seen that after 7 variables the metric gets worse.
By investigating the output of the regsubsets, it shows which of the variables are selected to give the best results.

The best 3-variable model is selecting size, phosphate and totlabor as the best performing variables. These variables are also in every other greater model. Therefore they are remembered when searching for a optimal model for the lasso regression. Also some other variables like pesticide, variety, wage and region are likely to have an influence on the model.

2.3 Preprocessing

As a specific package for the lasso regression, the glmnet package, is used, the data needs to be preprocessed. One important step therefore is to transform the qualititative variables like factor variables to dummy variables so the model can use them. The method is relatively simple by creating extra features with binary values. model.matrix is doing this transformation automatically by creating a design matrix out of the data frame. It also needs the information which variables to transform for which an expression is needed. This gives the opportunity to select the wished variables and also to do mathematical transformation like logarithm or polynomial conversion before applying and getting the model matrix for the model.
In addition to that a scaling option is implemented to test if the model is better when scaling the data.

2.4 Training and Evaluation

As the input features are transformed properly the lasso regression can be trained. After training the function return a list of models. This is because of the hyperparameter λ. This allows us to fine tune the model and improve its performance. To do this, a simple 10-fold-cross-validation on the training data is applied which is also included in the package glmnet. But before starting the validation a grid of possible lambdas is prepared to have the possibility to change the spectrum of the lambda parameter. Afterwards a plot is obtained showing the mean cross-validated error depending on the λ. The dotted lines in the plot are the λ which minimizes the mean cross-validated error the most and the λ which gives most regularized model such that the cross-validated error is within one standard error. In this task the λ with the minimum mean cross-validated error is chosen.
To evaluate the selected model on the validation data with the chosen λ the metrics MSE, BIC, AIC, AICc (modification of AIC as a correction for small sample sizes) and R^2 are used.
As for the glmnet package no known implemented function is found the metrics BIC,AIC, AICc and R^2 are implemented therefore manually. The formulas used are:

$$\text{BIC} = \chi^2 + k\ln(n)$$

with

$$\mathbf{X}^2 = \text{ Null deviance } - \text{ Residual deviance}$$

and k as the number of parameters estimated by the model and n the number of observations.

$$\text{AIC} = -\chi^2 + 2k$$

$$\text{AICc} = AIC + \frac{2k(k+1)}{n-k-1}$$

and

$$\begin{aligned} R^2 &= 1 - \frac{\text{sum squared regression (SSR)}}{\text{total sum of squares (SST)}} \\ &= 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \end{aligned}$$

where y are the actual values, $\hat{y}$ the predicted values and $\bar{y}$ the mean value of the y.

2.5 Model Selection

After setting the environment and the criterions of the training and evaluating of models, the best model can be selected out of the possible feature subsets. As indicated at the beginning of the Feature Selection section, the variable size, totlabor and phosphate are used to start with. However, this is also done sequentially to obtain the metrics of each model. For the other variables mentioned, this preselection is continued. Additional mathematical transformations of the variables are also considered and included. In order to provide as much variation as possible, the variables are also compared with their size-scaled correspondents. The results of the evaluation are stored in a table and assessed.

Table 3: Lasso comparison for variable selection

var	df	MSE.train	MSE.val	aic	aicc	bic	r_sq
size+seed							
log(seed)	2	0.1675	0.1244	-505.5	-505.4	-496.6	0.8753
seed	2	0.1789	0.1466	-498.5	-498.4	-489.6	0.8530

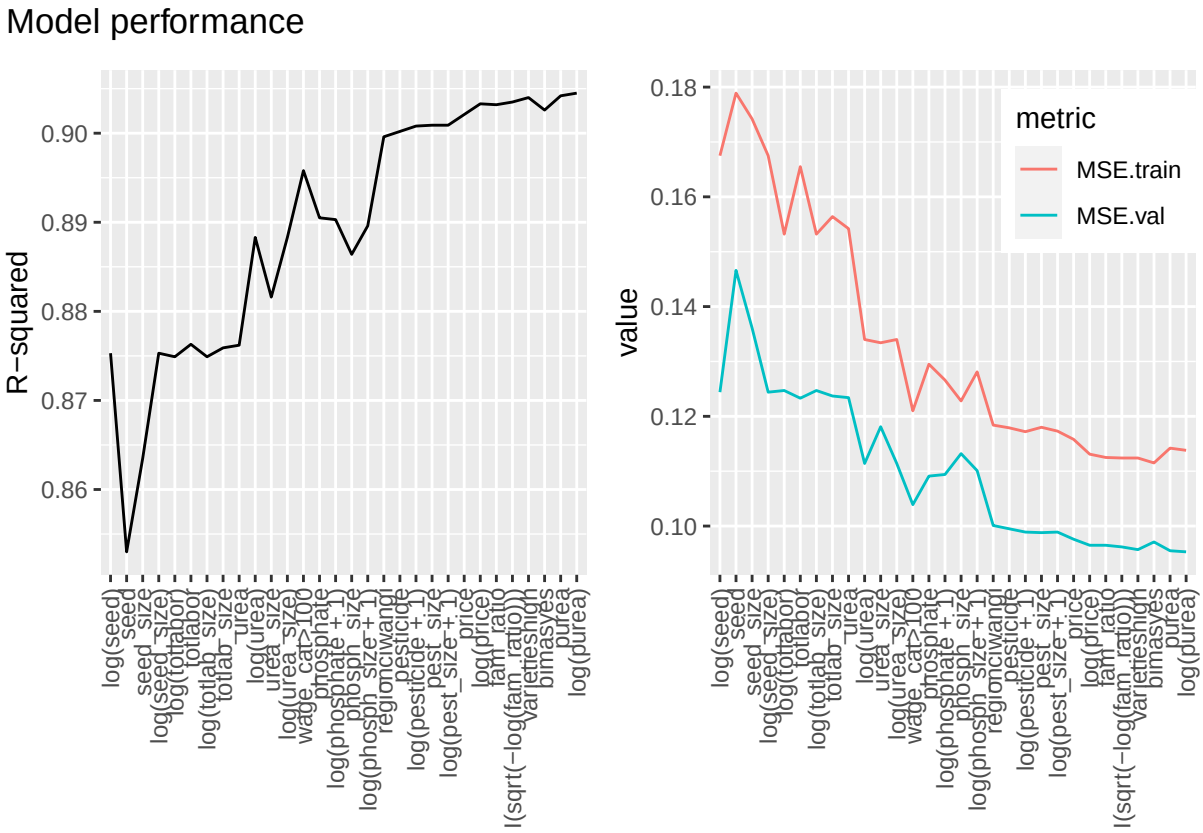
Table 3: Lasso comparison for variable selection *(continued)*

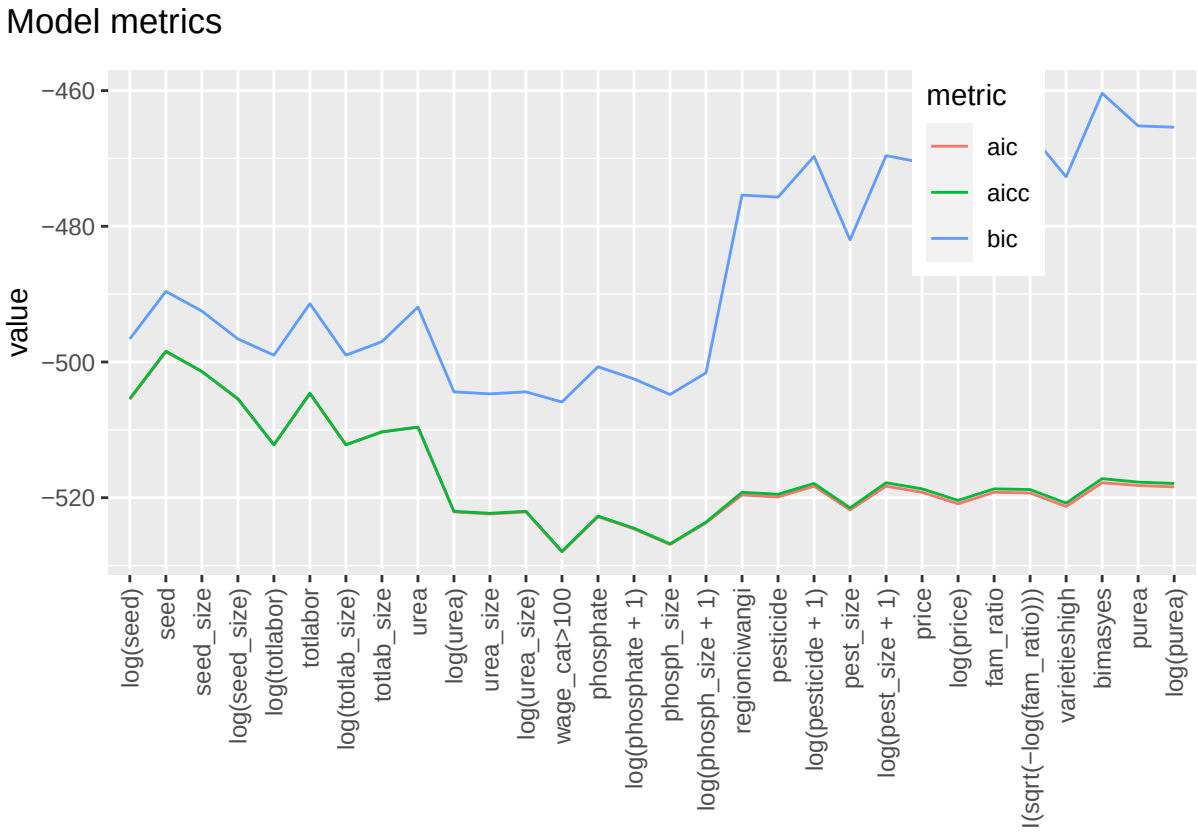
var	df	MSE.train	MSE.val	aic	aicc	bic	r_sq
seed_size	2	0.1742	0.1361	-501.4	-501.4	-492.5	0.8635
log(seed_size)	2	0.1675	0.1244	-505.5	-505.4	-496.6	0.8753
size+seed+totlabor							
log(totlabor)	3	0.1532	0.1247	-512.2	-512.2	-499.0	0.8749
totlabor	3	0.1655	0.1233	-504.7	-504.6	-491.4	0.8763
log(totlab_size)	3	0.1532	0.1247	-512.2	-512.2	-499.0	0.8749
totlab_size	3	0.1564	0.1237	-510.3	-510.3	-497.0	0.8759
size+seed+totlabor+urea							
urea	4	0.1542	0.1234	-509.6	-509.6	-491.9	0.8762
log(urea)	4	0.1340	0.1114	-522.1	-522.0	-504.4	0.8883
urea_size	4	0.1334	0.1181	-522.4	-522.3	-504.7	0.8816
log(urea_size)	4	0.1340	0.1114	-522.1	-522.0	-504.4	0.8883
size+seed+totlabor+urea+wage_cat							
wage_cat>100	5	0.1210	0.1039	-528.0	-527.9	-505.9	0.8958
size+seed+totlabor+urea+phosphate							
phosphate	5	0.1295	0.1091	-522.8	-522.7	-500.7	0.8905
log(phosphate + 1)	5	0.1266	0.1094	-524.6	-524.5	-502.5	0.8903
phosph_size	5	0.1228	0.1132	-526.9	-526.8	-504.8	0.8864
log(phosph_size + 1)	5	0.1281	0.1101	-523.7	-523.6	-501.6	0.8896
size+seed+totlabor+urea+wage_cat+region							
regionciwangi	10	0.1184	0.1001	-519.6	-519.2	-475.4	0.8996
size+seed+totlabor+urea+wage_cat+region+pesticide							
pesticide	10	0.1179	0.0995	-519.9	-519.5	-475.7	0.9002
log(pesticide + 1)	11	0.1172	0.0989	-518.3	-517.9	-469.7	0.9008
pest_size	9	0.1180	0.0988	-521.8	-521.5	-482.0	0.9009
log(pest_size + 1)	11	0.1173	0.0989	-518.3	-517.8	-469.6	0.9009
size+seed+totlabor+urea+wage_cat+region+pesticide+price							
price	11	0.1158	0.0976	-519.2	-518.7	-470.6	0.9021
log(price)	11	0.1131	0.0965	-520.9	-520.4	-472.2	0.9033
size+seed+totlabor+urea+wage_cat+region+pesticide+price+fam_ratio							
fam_ratio	12	0.1125	0.0965	-519.2	-518.7	-466.2	0.9032
I(sqrt(-log(fam_ratio)))	12	0.1124	0.0962	-519.3	-518.8	-466.3	0.9035
size+seed+totlabor+urea+wage_cat+region+pesticide+price+varieties							
varietieshigh	11	0.1124	0.0957	-521.3	-520.8	-472.7	0.9040
size+seed+totlabor+urea+wage_cat+region+pesticide+price+varieties+bimas							
bimasyes	13	0.1115	0.0971	-517.8	-517.2	-460.4	0.9026
size+seed+totlabor+urea+wage_cat+region+pesticide+price+varieties+purea							
purea	12	0.1142	0.0955	-518.2	-517.7	-465.2	0.9042
log(purea)	12	0.1138	0.0953	-518.4	-517.9	-465.4	0.9045

Table columns:

- var: the variable that is added
- df: degrees of freedom
- MSE.train: MSE on training data
- MSE.val: MSE on validation data
- aic: Akaike information criterion
- aicc: Akaike information criterion + penalty term
- bic: Bayesian information criterion
- r_sq: R-squared

As described at the feature selection, the RSS and the R^2 are improving with the increase of features selected. However the other metrics (AIC,BIC and AICc) are behaving like the feature selection predicts. By increasing the number of features after approximately 6 the metrics are generally getting worse. This means that the number of features does not justify the improvement of the RSS and the R^2. The penalty for the more complex model is higher than it has a positive effect. Therefore the smaller model is preferred.





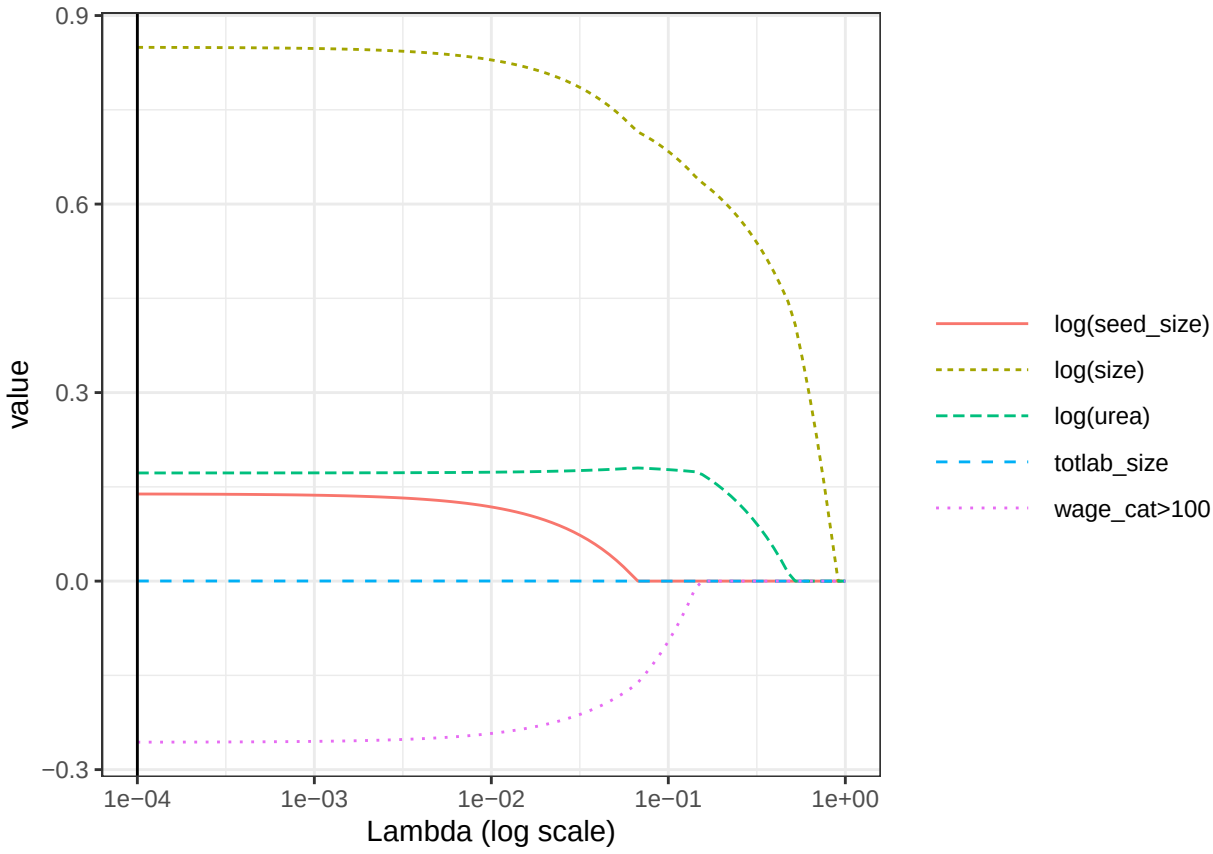
The best model to be chosen is:

$$log(g) = \beta_1 * log(s) + \beta_2 * log(e) + \beta_3 * t + \beta_4 * log(u) + \beta_5 * w + \beta_0$$

where :

- g: goutput
- s: size
- e: seed
- t: totallabor
- u: urea
- w: wage_cat
- β_i: coefficients and intercept

The best model also allows to look at its lambdas and the coefficients.


0.9
0.6
0.3
0.0
−0.3
value
1e−04
1e−03
1e−02
1e−01
1e+00
Lambda (log scale)
log(seed_size)
log(size)
log(urea)
totlab_size
wage_cat>100

3 Second Model

The second model to present and evaluate is called Generalized Additive Model (GAM)

3.1 Generalized Additive Model (GAM)

Generalized additive models (GAMs) provide a general framework for extending a standard linear model by allowing non-linear functions of each of the variables, while maintaining additivity. Just like linear models, GAMs additivity can be applied with both quantitative and qualitative responses. GAMs allow us to fit a non-linear predictor f_j to each variable x_{ij}, so that we will find non-linear relationships that standard multiple linear regression will miss. We do not need to manually try out many different transformations on each variable individually. The general GAM formula is as follows:

$$Y_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \ldots + f_p(x_{ip}) + \epsilon_i$$

There is no constraint that each f_j has to be the same type of function. So f_1 could be a quadratic function, f_2 a smoothing spline function and f_3 a loess function. And the smoothness of the function f_j for the variable X_j can be controlled independently for each variable.

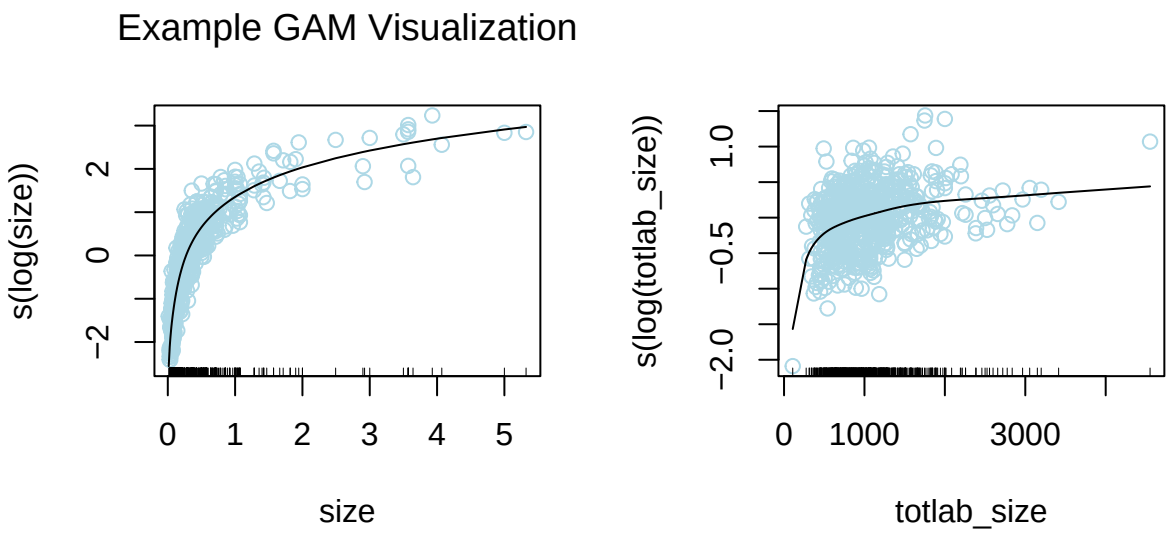
- **Regression splines** are more flexible than polynomials and step functions, and in fact are an extension of the two. They involve dividing the range of X into K distinct regions. Within each region, a polynomial function is fit to the data. However, these polynomials are constrained so that they join smoothly at the region boundaries, or knots.
- **Smoothing splines** are similar to regression splines, but arise in a slightly different situation. Smoothing splines result from minimizing a residual sum of squares criterion subject to a smoothness penalty.
- **Local regression** is similar to splines, but differs in an important way. The regions are allowed to overlap, and indeed they do so in a very smooth way.

A GAM is restricted to be additive. Important interactions might be missed, but we can manually add an interaction term to the GAM model by adding a predictor for $X_j X_k$. Fitting a GAM with a smoothing spline is not quite as simple as fitting a GAM with a natural spline, since in the case of smoothing splines, least squares cannot be used. However, standard software such as the gam() function from the gam library in R can be used to fit GAMs using smoothing splines, via an approach known as backfitting. So there is an important difference between smoothing splines and naturals splines. In the first case what you are fitting is a penalized spline model while in the second just regression splines, i.e. splines without penalty.

The s() function, which is part of the gam library, is used to indicate that we would like to use a smoothing spline. Qualitative variables are automatically converted into dummy variables by the gam function according to the amount of levels they have.

Explanation of GAM visualization

The lower visualization is an example of our 2nd GAM including two variables, size and totlab_size. For both of them we are using a smoothing spline function together with a log transformation. Because the model is additive, we can examine the effect of each f_j on Y individually while holding all of the other variables fixed. All panels from above have the same vertical scale. This allows us to visually assess the relative contributions of each of the variables. We observe that size and totlab_size have a large effect on goutput. The left-hand panel indicates that holding totlab_size fixed, goutput increases with size and is very steep up to about a size of 0.5 hectar and then becomes more and more flat. The right-hand panel indicates that holding size fixed, goutput increases drastically with the increased proportion of labour per size up to a proportion of about 500 hours/hectar and then flattens out. We can also see from the stripchart on the x-axis of the panel that most of the data are of smaller sizes up to 1 hectar and between 300 to 2000 hours of labor invested per hectar.



3.2 Model Selection

For retrieving our final model we make use of an approach called forward selection (cf. James p. 79). We begin with a model, that contains the variable with highest correlation to our dependent variable goutput, which is size. We then add different representations of one variable and add those to the model, which result in the lowest deviance for the new two-variable model. This approach is continued until all variables have been tried out. Forward selection is a greedy approach, and might include variables early that later become redundant.

We will first start with using smoothing spline function for each variable s() and we will use the default family, the gaussian(link = "identity"), in the GAM. We will also use the default degrees of freedom for each spline function, which is 3 and corresponds to a cubic spline. Based on the non-parametric p-value we access the suitability of the complexity of the chosen function and try out a different amount of degrees of freedom or other function e.g. just linear representation.

The model results are mainly compared by using the p-value from the model summary, which is telling if the variable is significant important. We will use a p-value of 5% for the evaluation of variable significance. To compare different representations of the same variable we will use the deviance. In addition we list other metrics like MSE on the training and validation data as well as AIC. For an comparison of larger models with smaller models we will perform additional anova tests.

table for model selection: The following table summarizes the variable selection process for the final GAM model. Starting from the top of the table we add for each grouped row in the table another variable with different representations of the variable for example checking whether log transformation yields any better results than using the variable as is. For each grouped row, the last model, yielding a p-value < 0.05 is the chosen model for further analysis.

Table 4: GAM comparison for variable selection

var	df	MSE.train	MSE.val	dev	aic	p_val_p	p_val_np	df_np
size								
s(size)	609	924954	535892	126.3	783.5	0.0000	0.0000	3
s(log(size))	609	928878	546896	107.5	684.3	0.0000	0.0016	3
size+labour								
s(totlab_size)	605	794773	540756	95.1	617.3	0.0000	0.0001	3
s(log(totlab_size))	605	796454	532193	93.7	608.2	0.0000	0.0140	3
size+labour+urea								
s(urea_size)	601	785027	554113	77.6	500.3	0.0000	0.0031	3
s(log(urea_size))	601	785928	543432	77.7	501.3	0.0000	0.0000	3
size+labour+urea+phosphor								
s(phosph_size)	597	640702	494023	71.7	460.0	0.0000	0.2846	3
s(log(phosph_size + 1))	597	637849	493065	71.7	459.9	0.0000	0.0000	3
size+labour+urea+phosphor+seed								
s(log(seed_size))	593	643332	475996	70.7	459.2	0.0077	0.1453	3
s(seed_size)	593	646291	475982	70.6	458.1	0.0245	0.0851	3
size+labour+urea+phosphor+seed+pesticide								
s(pest_size)	589	609355	448837	68.9	451.8	0.0006	0.2926	3
pest_size	592	619321	450231	69.4	449.4	0.0010	NA	NA
size+labour+urea+phosphor+seed+pesticide+price								
s(price)	588	497317	372599	64.8	415.8	0.0007	0.0000	3
size+labour+urea+phosphor+seed+pesticide+price+family_labour								
s(fam_ratio)	584	492896	366035	64.2	417.8	0.1893	0.2678	3
size+labour+urea+phosphor+seed+pesticide+price+price_info								
s(pseed)	584	480752	374061	63.2	408.6	0.9312	0.0010	3
s(pphosph)	584	457470	352312	59.6	371.9	0.0000	0.0000	3
s(pphosph)+s(purea)	580	444033	356581	59.1	374.8	0.0887	0.0098	3
size+labour+urea+phosphor+seed+pesticide+price+price_info+wage								
s(wage)	580	411551	348047	57.6	359.7	0.0000	0.2419	3
wage_cat>100	583	422753	355094	58.1	358.7	0.0000	NA	NA
size+labour+urea+phosphor+seed+pesticide+price+price_info+wage+categorical								
bimas	581	428066	370729	57.1	352.2	0.0042	NA	NA
bimas+varieties	579	412204	348847	56.7	351.4	0.1074	NA	NA
bimas+status	579	412980	369218	56.9	354.3	0.3734	NA	NA
bimas+region	576	419587	361417	56.6	356.3	0.2124	NA	NA
size+labour+urea+phosphor+seed+pesticide+price+price_info+wage+categorical+excluded								
s(purea)	577	411658	381500	56.4	352.1	0.0432	0.0017	3
s(purea)+varieties	575	395256	358050	55.9	351.2	0.1035	NA	NA

Table columns:

- var: the variable that is added
- df: degrees of freedom
- MSE.train: MSE on training data
- MSE.val: MSE on validation data
- dev: residual deviance (goodness of fit)
- aic: Akaike information criterion
- p_val_p: parametric p-value for last added variable
- p_val_np: non-parametric p-value for last added variable
- df_np: non-parametric degrees of freedom for last added variable

Variable selection explanation:

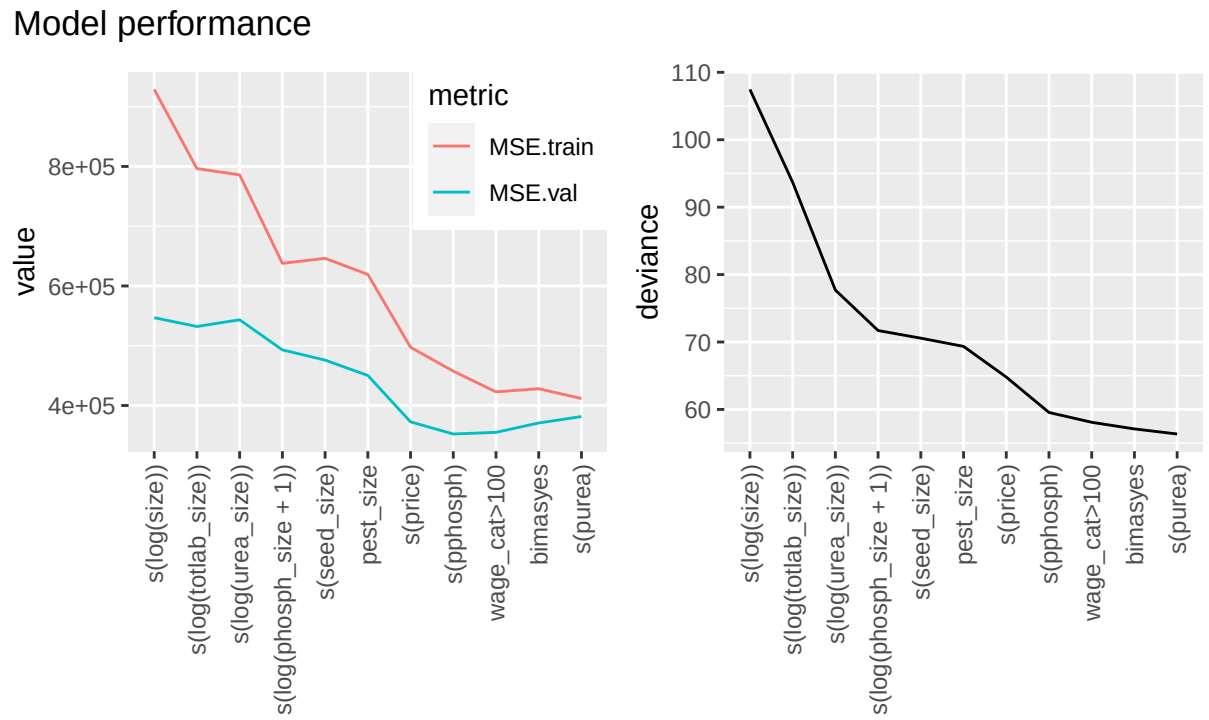
- size, labour, and urea: we will use logarithm of these variables, because of a smaller deviance
- phosphor: this variable has lots of zeros as values. So we can not use directly log transformation because log of 0 is -inf. log(x+1) transformation is the best way to avoid errors created by log transformation and is widely used among data scientists. So we will use this approach. We will use logarithm of phosph_size, because of a smaller deviance.
- seed: we will use seed as is, because of a smaller deviance.
- pesticide: we will use a linear fit of the variable, because the Anova test shows (cf. Anova test for pest_size) that the variable is significantly important for the model, but a non-linear transformations does not yield a significant difference.
- price, family labour: Both variables are significant important for the model.
- material prices:
  - pseed: p-value > 0.05, thus we do not use this variable
  - pphosph: p-value < 0.05, so we add the variable to our model
  - pphosph + purea: adding purea in addition to pphosph does does not improve the model significantly (p-value < 0.05)
- wage: we will use wage_cat instead of a non-linear transformations of wage, due to the results of the anova test (cf. Anova test for wage/wage_cat)
- categorical variables:
  - bimas: p-value < 0.05, so we add the variable to our model
  - bimas+ additional categorical variables: adding the other categorical variables does not significantly improve the model (p-value > 0.05) so no other categorical variables are added in addition to bimas.
- final model: we check again for excluded variables after some more variables have been added if the results change.
  - purea: the p-value now shows a values less than 0.05, thus the variable is significantly improving the model.
  - purea + varieties: adding varieties in addition to purea does not yield a significant improvement of the model.

Anova test for pest_size

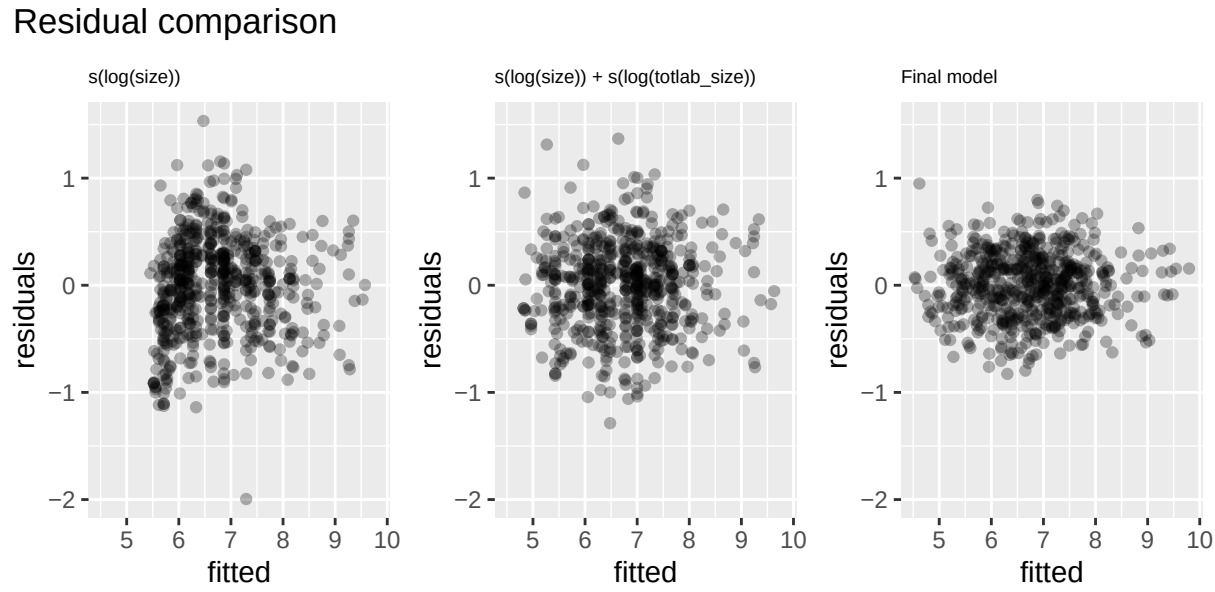
```
## Analysis of Deviance Table
##
## Model 1: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size)
## Model 2: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + pest_size
## Model 3: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + s(pest_size)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       593     70.568
## 2       592     69.352  1  1.21632 0.001266 **
## 3       589     68.944  3  0.40733 0.323386
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Anova test for wage/wage_cat

```
## Analysis of Deviance Table
##
## Model 1: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + pest_size + s(price) +
##     s(pphosph)
## Model 2: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + pest_size + s(price) +
##     s(pphosph) + wage_cat
## Model 3: log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + pest_size + s(price) +
##     s(pphosph) + s(wage)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       584     59.551
## 2       583     58.095  1  1.45619 0.000129 ***
## 3       580     57.626  3  0.46805 0.194241
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



The model performance graphs are visualizing the metrics MSE and deviance for the chosen models based on the forward selection process. The deviance is always decreasing, since this was our selection criteria, but for the MSE values we can see that some variables have a negative influence on the MSE. Interesting here is that the MSE values for the validation data are always below the training data. Normally one would expect to overfit on the training data so that a performance on non-seen data is worse. But here this isn't the case. This might be due to the fact that there are lots of outliers in the data, and the training data set might include proportionally more outliers than the validation data, which negatively affects the MSE.



The residual visualizations are exemplary for our 1st, 2nd and final model. From the first model we can

clearly see a relationship in the residuals. The relationship looks quadratic. This relationship vanishes when the 2nd variable totlab_size is added to our model. The last residual plot of our final model is more centered in comparison to the other models.

3.3 Final Model

Our final GAM has the following formula:

```
## log(goutput) ~ s(log(size)) + s(log(totlab_size)) + s(log(urea_size)) +
##     s(log(phosph_size + 1)) + s(seed_size) + pest_size + s(price) +
##     s(pphosph) + wage_cat + bimas + s(purea)
```

The final GAM doe not include the following variables: * numerical: fam_ratio, pseed * categorical: varieties, status, region

All numerical variables have been included by using smoothing spline function with a default degree of freedom of 3 except the variable pest_size, where we used just a linear fit.

To have a better comparison against the Lasso model we will calculate R-squared and adjusted R-squared.

R-squared (R^2): is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. Essentially, an R-Squared value of 0.9 would indicate that 90% of the variance of the dependent variable being studied is explained by the variance of the independent variable.

$$R^2 = 1 - \frac{Unexplained\,Variation}{Total\,Variation}$$

$R^2 = 0.9064434$

Adjusted R-Squared: The adjusted R-squared compares the descriptive power of regression models that include diverse numbers of predictors. Every predictor added to a model increases R-squared and never decreases it. Thus, a model with more terms may seem to have a better fit just for the fact that it has more terms, while the adjusted R-squared compensates for the addition of variables and only increases if the new term enhances the model above what would be obtained by probability and decreases when a predictor enhances the model less than what is predicted by chance.

$$R^2_{adj} = 1 - \frac{(1 - R^2) * (n - 1)}{n - p - 1}$$

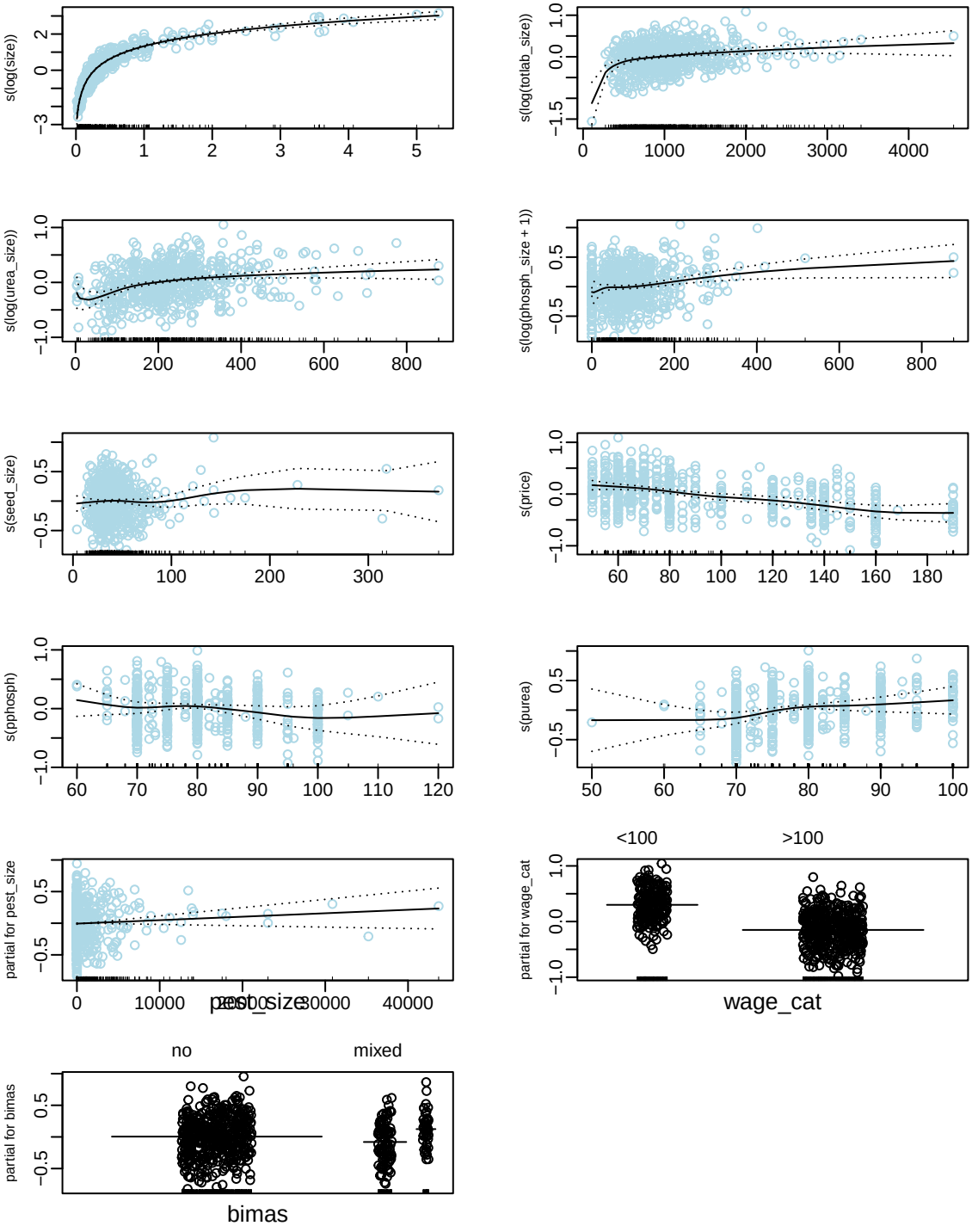
n = total sample size, p = number of predictors

R^2_{adj}= 0.9044163

Visualization explanation Because the model is additive, we can examine the effect of each f_j on Y individually while holding all of the other variables fixed. All panels from the final model have the same vertical scale. This allows us to visually assess the relative contributions of each of the variables. We observe that size, totlab_size, urea_size, phosph_size, seed_size, pest_size and purea have a positive influence on goutput. E.g. with increasing size, goutput is increasing. Wherease price and pphosph have a negatice influence on goutput.

For the categorical variables we can see, that the farmers goutput tends to be higher when paying a wage less than 100 Rupiah per hour and bimas equal to yes is also increasing goutput.

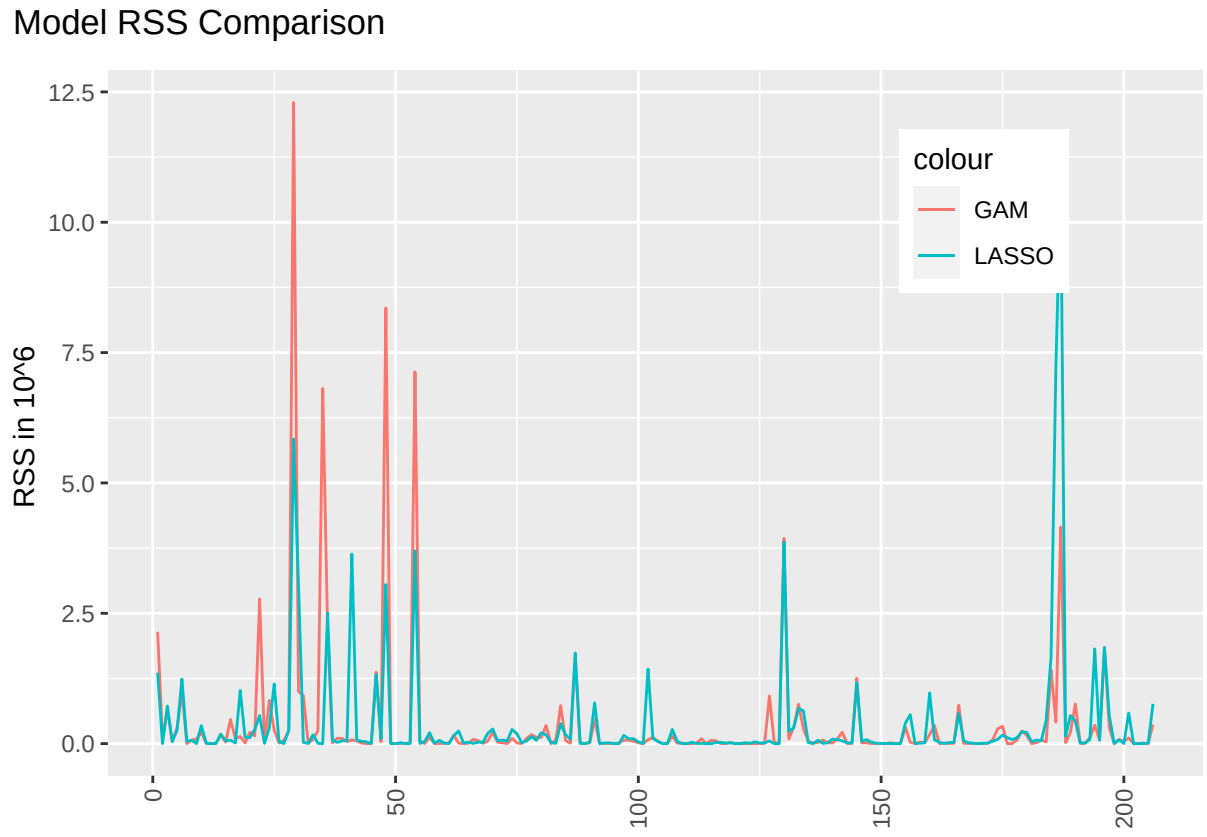
Final GAM visualization

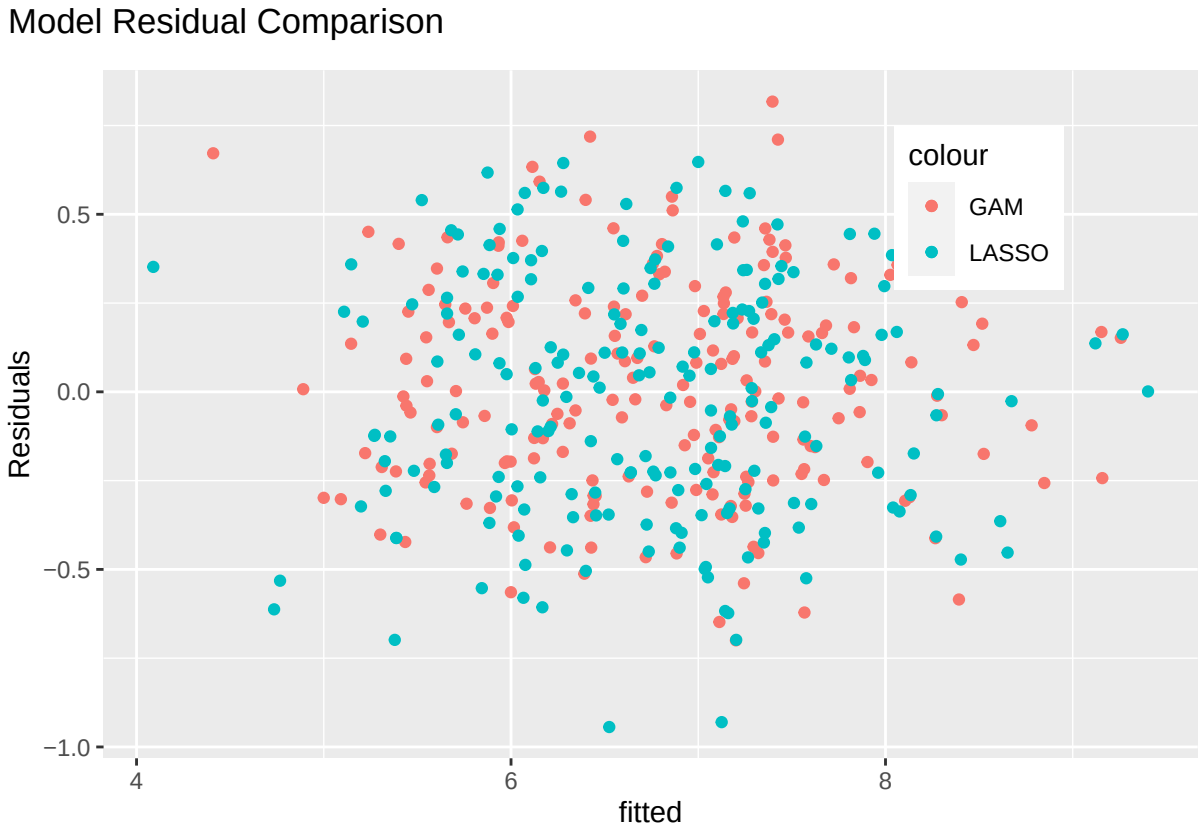


4 Comparison

After the best model of Lasso Regression and the Generalised Additive Model has been determined by validation, a comparison of these is undertaken.
The first striking thing is that in both models the MSE is smaller in the validation set than in the training set.This is contrary to intuition. As already mentioned in the previous chapter GAMs, this could be due to the fact that there are lots of outliers in the data, and the training data set might include proportionally more outliers than the validation data, which negatively affects the MSE.
The test data set that was put away at the beginning is used for this. Again, familiar metrics as in the previous sections are used for evaluation. Among others, the number of degrees of freedom, MSE and the R^2. In addition, the residuals sum of squares and the residuals are compared to see how good the prediction of the model is.

Metric	Lasso	GAM
df	609	577
R^2	0.878	0.868
MSE	3.8238397×10^5	3.7900199×10^5





It can be seen from the tables that both models perform equally well. The Lasso model performs a little better with the R^2, but this should not be a problem. However, the Lasso model has more degrees of freedom than the other model, which is due to the fact that fewer predictor variables are needed than in the GAM model. However, the GAM model performs better on the MSE.
With regard to the graphs, there is also no significant difference between the models. It can be seen, however, that in the RSS plot all two models have similar false predictions at certain points in the test data set, since the peaks are at the same points. This is probably due to extreme values in the data set.
It can be concluded that both models can be used for predicting the output of the data set. It is up to the user to decide which model to use. If a simple model is needed, even for explanation, the Lasso model is preferred, because the GAM model is more complex and is used for non-linear complex relationships in the data set.