

1 LazyModeler: An R package for automatic 2 simplification, check, and visualization of regression 3 models




4 **Lara M. Kösters** ^{1*} and **Kevin Karbstein** ^{1*}

5 ¹ Max Planck Institute for Biogeochemistry, Department of Biogeochemical Integration, Jena, Germany

6 * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](#)).

7 Summary

8 Setting up, simplifying, checking, and visualizing regression models continues to be a time-
9 consuming task involving multiple, sometimes concurrent, workflows and software packages.
10 This particularly applies to big data research where several models need to be set up and
11 optimized. To tackle this problem, we present LazyModeler - a statistical package for the
12 programming language R that allows to easily perform regression modeling. It includes removal
13 of autocorrelated variables, choice between several types of (non)linear regression models,
14 standard stepwise model simplification, various model quality checks, plotting of coefficient
15 estimates and relationships, and output generation. LazyModeler will significantly speed up
16 regression modeling, enabling people to analyze and illustrate their data in a statistically
17 reliable and standardized manner.

18 Statement of need

19 Statistical modeling describes the process of finding a mathematical function with specific
20 statistical assumptions that best fits the observed data ([Crawley, 2007, 2015](#); [Henley et al., 2020](#)). This process attempts, in practice, to find a (causal) relationship between a dependent
21 response variable y and an independent predictor variable x for any postulated hypothesis. For
22 statistical inference and graphics in science, the programming environment R ([R Core Team, 2024](#))
23 has become highly popular.

24
25 Linear regression models, as one of the most basic and powerful tools, have been frequently
26 applied in this context ([Crawley, 2007, 2015](#); [Li, 2023](#); ?). Because of their flexibility, they also
27 allow for non-normally distributed response variables (e.g., in the case of binomial, proportional,
28 or count data), and any kind of transformation for numerical (e.g., polynomial or logarithmic)
29 and categorical (e.g., centered or one-hot/fractional encoded) predictor variables, as well as
30 interactions among them ([Cai et al., 2023](#); [Henley et al., 2020](#); [Karbstein et al., 2019, 2020, 2021](#);
31 [Römermann et al., 2016](#); e.g., ?; ?). Regression models also provide the ability to
32 control for random effects that may influence the variables of interest ([Bauer & Albrecht, 2020](#);
33 e.g., [Wicke et al., 2016](#); ?). Although other statistical technologies can outperform
34 them in highly complex, non-linear scenarios, regression models allow for detailed variable
35 transformation and interaction, mathematical formula specification, calculation of effect sizes,
36 determination of variable significance, and thus hypothesis testing and explanation ([Benjamin et al., 2018](#);
37 [Cai et al., 2023](#); [Karbstein et al., 2023](#); [Li, 2023](#); [Schulz et al., 2020](#); ?). Recent
38 developments make regression models also applicable to nonlinear scenarios ([Bates et al., 2024](#);
39 e.g., ?). Consequently, they are of high practical value in finding and interpreting significant
40 relationships.

In statistical modeling, and especially in real-world applications, multiple predictors are assumed for a given response variable. As a consequence, people strive to exclude the irrelevant from the relevant (statistically significant) information, which is called model simplification (Crawley, 2007, 2015; ?). One of the most widely used optimization workflows is stepwise model simplification. For example, starting from a full/saturated model, the least significant variable ($p > 0.05$) is excluded until the final minimal adequate model is attained [‘backward simplification’; Crawley (2007); (?); Crawley (2015)]. Each model simplification step will be justified with certain metrics (e.g., SSE, AIC, or BIC) (Henley et al., 2020). Given the number of models, variables of interest, and their data characteristics, this task can be extraordinarily time consuming. Currently, only AIC/BIC-based automated simplification is available (e.g., ‘stepAIC,’ ?). Nevertheless, model simplification continues to be a rather manual process [on Google Scholar, only ca. 5,000 “stepAIC” entries despite ca. 5,000,000 “linear regression model” studies (0.1%); e.g., Römermann et al. (2016); Karbstein et al. (2019); Henley et al. (2020); Karbstein et al. (2020); Cai et al. (2023); Li (2023)]. In addition, simplification and other aspects such as data cleaning, model comparison and quality control, and output visualization have not been automated. An easy-to-use, all-in-one function for the entire modeling process within a single software package is missing.

Our R package LazyModeler addresses these issues by automating variable selection, model optimization, and output illustration and generation. In detail, users will be enabled to automatically remove autocorrelated variables, choose between several types of (non)linear regression models (e.g., LM, GLM, LMER, GLMER, GAM, or NLMER), perform stepwise model simplification, check model quality, plot coefficient estimates and relationships, and generate the output of the final model.

Overview and major functions

LazyModeler automatizes all necessary steps needed for use of (non)linear regression models. It comprises three major functions that are included within the main function `optimize_model`.

The first major function `remove_autocorrelations` checks for any autocorrelations ($|r| > 0.7$) (?) given a list of variables sorted by relevance. Automatic removal of these autocorrelations is possible through the use of a function parameter. Removal will follow the order of the list of variables, ensuring that the user’s expertise on the importance of features is respected. A named list is returned with a) a vector containing all removed predictors, and b) a dataframe listing autocorrelations and information on deleted variables.

The main function provides the model formula to the second major function `simplify_model`. If autocorrelations were detected, the formula is updated accordingly. The regression model is then calculated. Options for the models are: `lm`, `glm`, `lmer`, `glmer`, `gam`, or `nlmer`, with all possible distributions of the response variable being allowed. Stepwise backward simplification or forward model selection takes place using an iterative process where each time the metric(s) specified by the user are applied to the model to check whether further simplification/selection is needed. Main variables are kept when they are involved in interactions. Options for the metrics are: `aov`, `aic`, `aicc`, or `bic`. The final model is returned to the main function alongside its metadata as well as simplification history if requested by the user.

Using the third major function `fancy_plotting`, the final model then undergoes multiple visualization steps. Plots to assess model quality are created using the standard plot function available through base R, or model check included in the performance R package (?). Furthermore, the script produces regression, box, or violin plots for each numerical or categorical coefficient as well as plots depicting effects sizes and estimates. All generated plots are returned to the user within a named list. The main function additionally returns the output of both the model simplification/selection and autocorrelation functions as well as the summary of the final model.

90 LazyModeler makes use of the R package corrplot (?) to calculate correlations between
91 variables, lme4 (Bates et al., 2024) for regression modeling, tidyverse (?) for data handling,
92 and spind (?) for calculation of AICc scores. For generation of plots visualizing regression,
93 effect size, and estimates, the script further leverages tidyverse and color palettes included in
94 the colorspace (?) and viridis (?) R packages.

95 **Example**

```
# import example data
data(plants)

# check data structure
str(plants)
summary(plants)

# testing dataset (subset) based on Karbstein et al. 2021
#(https://onlinelibrary.wiley.com/doi/10.1111/mec.15919)

results_example <- optimize_model(plants, quote(sexual_seed_prop ~
altitude + latitude_gps_n + longitude_gps_e + (solar_radiation +
annual_mean_temperature + isothermality)^2 + I(isothermality^2) +
habitat + ploidy), autocorrelation_cols = c("solar_radiation",
"annual_mean_temperature", "isothermality", "altitude",
"latitude_gps_n", "longitude_gps_e"), automatic_removal=TRUE,
autocorrelation_threshold = 0.8, correlation_method="spearman",
model_type = "glm", model_family = "quasibinomial",
assessment_methods=c("anova"), simplification_direction="backward",
omit.na="overall", scale_predictor=TRUE,
plot_quality_assessment="performance", round_p=3,
cor_use="complete.obs", plot_relationships=TRUE, jitter_plots=TRUE,
plot_type="violinplot", stat_test="wilcox",
backward_simplify_model=TRUE, trace=TRUE)
```

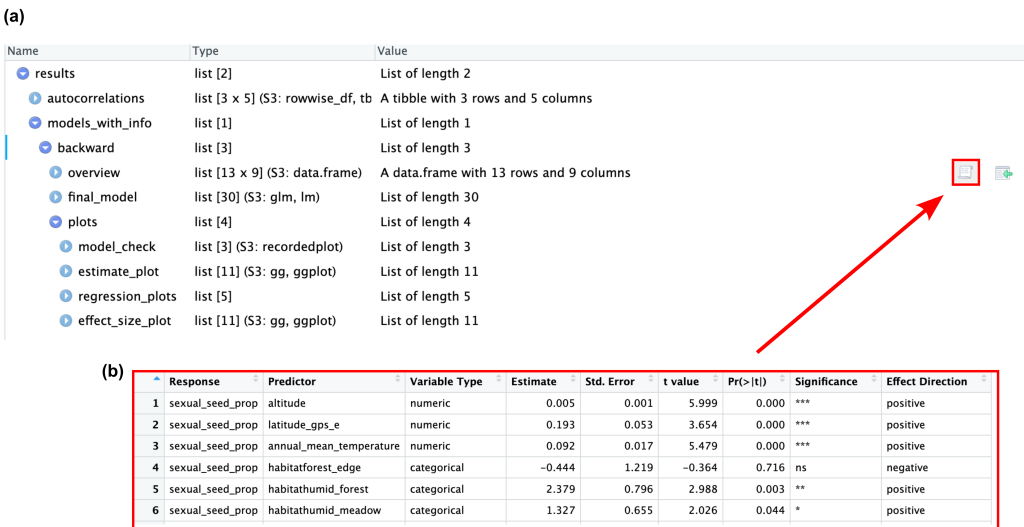


Figure 1: Navigating through the output. For example, (a) simply click on dataframe button highlighted with a red arrow to (b) illustrate the final model output.

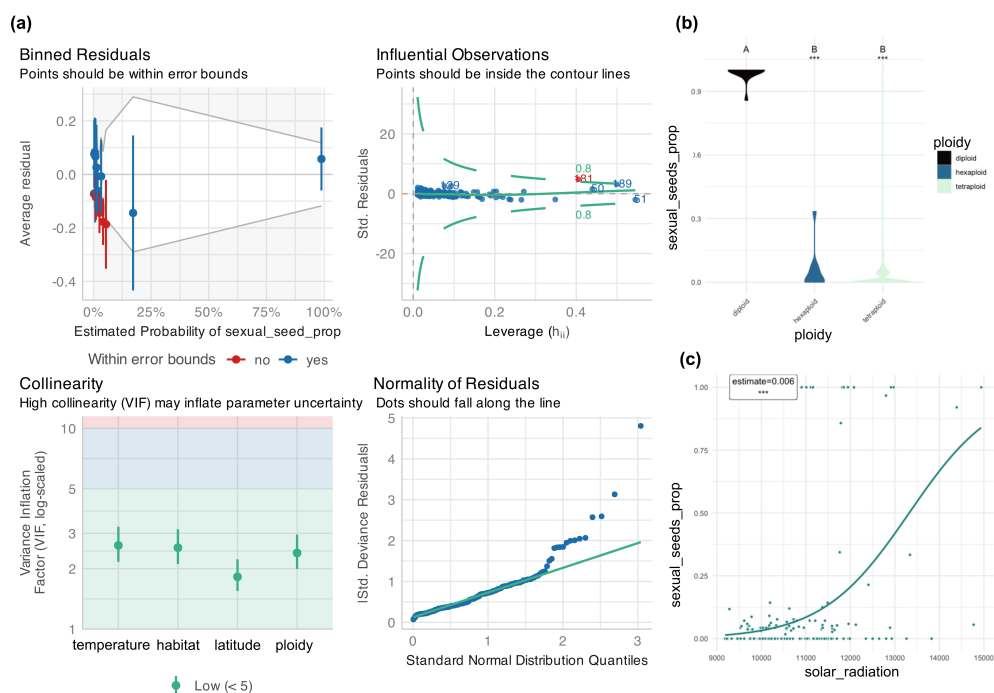


Figure 2: (a) Model quality check and (b,c) exemplary output plots of significant relationships.

Code Availability

The code including basic documentation and an exemplary testing dataset will be made available upon publication on [Github](#) and on [Comprehensive R Archive Network \(CRAN\)](#).

Acknowledgements

We acknowledge financial support from the German Federal Ministry of Education and Research (BMBF) grant 01IS20062.

References

- Bates, D., Maechler, M., Bolker, B., & Walker, S. (2024). *lme4 - Linear mixed-effects models using 'Eigen' and S4*. <https://github.com/lme4/lme4/>
- Bauer, M., & Albrecht, H. (2020). Vegetation monitoring in a 100-year-old calcareous grassland reserve in Germany. *Basic and Applied Ecology*, 42, 15–26. <https://doi.org/10.1016/j.baae.2019.11.003>
- Benjamin, A. S., Fernandes, H. L., Tomlinson, T., Ramkumar, P., VerSteeg, C., Chowdhury, R. H., Miller, L. E., & Kording, K. P. (2018). Modern machine learning as a benchmark for fitting neural responses. *Frontiers in Computational Neuroscience*, 12(July), 1–13. <https://doi.org/10.3389/fncom.2018.00056>
- Cai, L., Kreft, H., Taylor, A., Denelle, P., Schrader, J., Essl, F., Kleunen, M. van, Pergl, J., Pyšek, P., Stein, A., Winter, M., Barcelona, J. F., Fuentes, N., Inderjit, Karger, D. N., Kartesz, J., Kuprijanov, A., Nishino, M., Nickrent, D., ... Weigelt, P. (2023). Global models and predictions of plant diversity based on advanced machine learning techniques. *New Phytologist*, 237(4), 1432–1445. <https://doi.org/10.1111/nph.18533>
- Crawley, M. J. (2007). *The R Book* (p. 942). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9780470515075>
- Crawley, M. J. (2015). *Statistics: an introduction using R* (sec. ed., p. 339). John Wiley & Sons. ISBN: 1118448960
- Henley, S. S., Golden, R. M., & Kashner, T. M. (2020). Statistical modeling methods: challenges and strategies. *Biostatistics & Epidemiology*, 4(1), 105–139. <https://doi.org/10.1080/24709360.2019.1618653>
- Karbstein, K., Prinz, K., Hellwig, F., & Römermann, C. (2020). Plant intraspecific functional trait variation is related to within-habitat heterogeneity and genetic diversity in *Trifolium montanum* L. *Ecology and Evolution*, 10(11), 5015–5033. <https://doi.org/10.1002/ece3.6255>
- Karbstein, K., Römermann, C., Hellwig, F., & Prinz, K. (2023). Population size affected by environmental variability impacts genetics, traits, and plant performance in *Trifolium montanum* L. *Ecology and Evolution*, 13(8), 1–19. <https://doi.org/10.1002/ece3.10376>
- Karbstein, K., Tomasello, S., Hodač, L., Lorberg, E., Daubert, M., & Hörandl, E. (2021). Moving beyond assumptions: Polyploidy and environmental effects explain a geographical parthenogenesis scenario in European plants. *Molecular Ecology*, 30(11), 2659–2675. <https://doi.org/10.1111/mec.15919>
- Karbstein, K., Tomasello, S., & Prinz, K. (2019). Desert-like badlands and surrounding (semi-)dry grasslands of Central Germany promote small-scale phenotypic and genetic differentiation in *Thymus praecox*. *Ecology and Evolution*, 9(24), 14066–14084. <https://doi.org/10.1002/ece3.5844>

- 139 Li, J. (2023). Overview of high dimensional linear regression models. *Theoretical and Natural*
140 *Science*, 5(1), 656–661. <https://doi.org/10.54254/2753-8818/5/20230427>
- 141 R Core Team. (2024). *R: a language and environment for statistical computing*. R Foundation
142 for Statistical Computing. <http://www.r-project.org/>
- 143 Römermann, C., Bucher, S. F., Hahn, M., & Bernhardt-Römermann, M. (2016). Plant
144 functional traits – fixed facts or variable depending on the season? *Folia Geobotanica*,
145 51(2), 143–159. <https://doi.org/10.1007/s12224-016-9250-3>
- 146 Schulz, M.-A., Yeo, B. T. T., Vogelstein, J. T., Mourao-Miranada, J., Kather, J. N., Kording,
147 K., Richards, B., & Bzdok, D. (2020). Different scaling of linear models and deep learning
148 in UKBiobank brain images versus machine-learning datasets. *Nature Communications*,
149 11(1), 4238. <https://doi.org/10.1038/s41467-020-18037-z>
- 150 Wicke, S., Müller, K. F., DePamphilis, C. W., Quandt, D., Bellot, S., & Schneeweiss, G. M.
151 (2016). Mechanistic model of evolutionary rate variation en route to a nonphotosynthetic
152 lifestyle in plants. *Proceedings of the National Academy of Sciences of the United States*
153 *of America*, 113(32), 9045–9050. <https://doi.org/10.1073/pnas.1607576113>

DRAFT