



Robust gradient boosting for generalized additive models for location, scale and shape

Jan Speller¹ · Christian Staerk¹ · Francisco Gude² · Andreas Mayr¹ 

Received: 31 January 2023 / Accepted: 10 July 2023
© The Author(s) 2023

Abstract

Due to the increasing complexity and dimensionality of data sources, it is favorable that methodological approaches yield robust results so that corrupted observations do not jeopardize overall conclusions. We propose a modelling approach which is robust towards outliers in the response variable for generalized additive models for location, scale and shape (GAMLSS). We extend a recently proposed robustification of the log-likelihood to gradient boosting for GAMLSS, which is based on trimming low log-likelihood values via a log-logistic function to a boundary depending on a robustness constant. We recommend a data-driven choice for the involved robustness constant based on a quantile of the unconditioned response variable and investigate the choice in a simulation study for low- and high-dimensional data situations. The versatile application possibilities of robust gradient boosting for GAMLSS are illustrated via three biomedical examples—including the modelling of thyroid hormone levels, spatial effects for functional magnetic resonance brain imaging and a high-dimensional application with gene expression levels for cancer cell lines.

Keywords Distributional regression · Gradient boosting · High-dimensional · Log-logistic · Robust · Variable selection

Mathematics Subject Classification 62-08 · 62F35 · 62J99

✉ Andreas Mayr
amayr@uni-bonn.de

Jan Speller
speller@imbie.uni-bonn.de

Christian Staerk
staerk@imbie.uni-bonn.de

Francisco Gude
Francisco.Gude.Sampedro@sergas.es

¹ Department of Medical Biometrics, Informatics and Epidemiology, Medical Faculty, University of Bonn, Venusberg-Campus 1, Bonn 53127, Germany

² Clinical Epidemiology Unit, USC University Hospital, Travesía da Choupana, 15706 Santiago de Compostela, Galicia, Spain

1 Introduction

With the emergence of complex observational data in biomedical research there is also an increasing need for robust data analyses (see Barrios 2015; Monti and Filzmoser 2022). Robust statistical approaches should ensure that also for analyses purely based on observational or routine data from clinical practice (and hence less controlled settings than clinical trials), the overall conclusions are not based on small amounts of corrupted or inconsistent observations such as outliers. On the other hand, many modern research questions also warrant more complex and flexible analysis tools, which may go beyond the classical focus on the mean of a distribution (Kneib et al. 2021). One of these flexible model classes are Generalized Additive Models for Location, Scale and Shape (GAMLSS, introduced by Rigby and Stasinopoulos 2005).

In contrast to classical regression models, GAMLSS allow for the modelling of multiple distribution parameters (e.g., also scale and shape parameters) in dependence of potentially different sets of covariates for each parameter. As a result, GAMLSS model the complete conditional distribution and not only the mean. Furthermore, the response variable can follow any distribution (continuous, discrete, mixed, e.g. zero-inflated continuous distributions with a spike at zero) and is not restricted to the exponential family as it is the case for regular generalized additive models (Hastie and Tibshirani 1990; Wood 2006). Recently, a robust fitting approach for GAMLSS was proposed by Aeberhard et al. (2021), which tackles the problem of outliers in the response variable. The core idea is to robustify the likelihood via a log-logistic transformation, which ensures that some extreme observations do not dominate the model fit. In contrast to more classical robust regression methods based on composite loss functions (Huber 1981; Amato et al. 2021), the focus on the likelihood facilitates the transfer of this robustness concept towards distributional regression. While the approach by Aeberhard et al. (2021) shows a very promising performance for low-dimensional models, it is not applicable for high-dimensional data with potentially more explanatory variables than observations ($p > n$), and does not provide data-driven variable selection.

To overcome these remaining issues, we incorporate the robustification by Aeberhard et al. (2021) in a boosting algorithm for GAMLSS (Mayr et al. 2012; Thomas et al. 2018). Boosting is a concept from machine learning that was later adapted to fit statistical models (Friedman 2001). An important advantage of these statistical boosting approaches is that they yield interpretable effect estimates for single predictor variables while allowing for high-dimensional data with $p > n$. Furthermore, they can be adapted to carry-out variable selection by stopping the algorithm before it converges (Bühlmann 2006). Recently, some approaches for robust fitting of statistical models via boosting have been proposed (Ju and Salibián-Barrera 2021; Speller et al. 2022), but they focus only on classical mean regression. We hence propose to incorporate the robustified likelihood approach into the boosting framework to fill the gap of robust approaches for fitting GAMLSS in the context of high-dimensional data.

As in classical robust regression approaches (Huber 1981; Maronna et al. 2019), the robustification of the likelihood in Aeberhard et al. (2021) is controlled by a robustness constant, which has to be specified before fitting the model. While there often exist reasonable default values which work well in many settings, the specification is

generally difficult as the amount of required robustness is typically unknown. To facilitate the application of our approach in practice, we additionally propose a data-driven quantile-based way to choose the robustness constant.

We analyse the performance of our approach regarding prediction accuracy and variable selection for different amounts and types of corrupted outcomes via a simulation study (Sect. 3) for Gaussian and Gamma distributed data, comparing it to the classical boosting algorithm for GAMLSS (Mayr et al. 2012; Thomas et al. 2018). In low-dimensional settings, we compare our approach also to the robust penalized maximum likelihood approach of Aeberhard et al. (2021). To further illustrate the practical relevance of our proposed method, we provide three biomedical applications (Sect. 4). We revisit the low-dimensional data application provided in Aeberhard et al. (2021) on spatial effects for functional magnetic resonance brain imaging using our robust boosting approach. Additionally, we present the analysis of a classical medium-sized epidemiological trial on thyroid hormone levels from the general population with data-driven variable selection. As a last illustrative example, we focus on gene expression levels with $p \gg n$ in cancer cell-lines.

2 Methods

2.1 Gradient boosting for GAMLSS

The *gamboostLSS* algorithm combines variable selection and prediction modeling while fitting GAMLSS using a component-wise gradient boosting approach (Mayr et al. 2012). A pre-specified set of regression-type base-learners (typically one for each covariate) is iteratively fitted to the negative gradient of the likelihood. In each iteration of the algorithm, the best-performing base-learner is selected and only this one is updated (Bühlmann 2006). This iteration-based procedure automatically leads to the selection of informative variables or base-learners, while base-learners that are never selected to be updated are effectively excluded from the final model. Typically, model tuning is performed by stopping the algorithm before convergence to avoid overfitting and to enforce variable selection and shrinkage.

To describe the component-wise gradient boosting algorithm in the GAMLSS regression framework, we use the following notation: The response observations y_i for $i = 1, \dots, n$ with conditional density functions $f(y_i | \boldsymbol{\theta}_i)$ are assumed to be independent given the parameter vector $\boldsymbol{\theta}_i = (\theta_{i,k})_{k=1, \dots, K}$. Here, each parameter $\theta_{i,k}$ of the parameter vector $\boldsymbol{\theta}_i$ can depend on a different subset of all available covariates, which is taken into account by using potentially varying index sets J_k for the covariates. In principle, the number of simultaneously modelled parameters K can be arbitrarily high, but for most distributions there are not more than $K = 4$ parameters (originally also denoted by $\boldsymbol{\theta} = (\mu, \sigma, \nu, \tau)$ for location, scale, skewness and kurtosis parameters, Rigby and Stasinopoulos 2005).

Every parameter θ_k is related to its additive predictor η_{θ_k} by its specific monotonic link function g_k :

$$g_k(\theta_k) = \eta_{\theta_k} = \beta_{\theta_k,0} + \sum_{j \in J_k} h_{\theta_k,j}(\mathbf{x}_{k,j}) \text{ for } k = 1, \dots, 4, \tag{1}$$

where $\beta_{\theta_k,0}$ denotes the parameter-specific intercept and $h_{\theta_k,j}(\mathbf{x}_{k,j})$ is the effect of the covariates $\mathbf{x}_{k,j}$ within the additive predictor for parameter θ_k . In the context of component-wise gradient boosting, each $h_{\theta_k,j}$ corresponds to one base-learner. In the simplest case this is linear but can also be a non-linear spline or other type of effect. Note that $h_{\theta_k,j}$ may also depend on multiple covariates like for interactions or spatial effects, where, e.g. coordinates are used for estimating the joint effect via one spatial base-learner (see brain data example in Sect. 4.1).

Once the model and base-learners in Eq. (1) are specified, the actual optimization problem $\hat{\theta} = \operatorname{argmax}_{\theta} \ell(\mathbf{y} | \theta)$ with the log-likelihood function

$$\ell(\mathbf{y} | \theta) = \sum_{i=1}^n \ell(y_i | \theta_i) = \sum_{i=1}^n \log(f(y_i | \theta_{i,1}, \theta_{i,2}, \theta_{i,3}, \theta_{i,4})), \tag{2}$$

is solved iteratively via gradient boosting.

Prior to the first iteration, an offset value is calculated for each additive predictor based on the unconditional distribution of the response variable \mathbf{Y} (e.g. the mean for the location parameter of a Gaussian distribution). Then, the partial derivatives with respect to the additive predictors η_{θ_k} are computed in each iteration $m \geq 1$ and the base-learners are fitted separately to the gradient, while only the best fitting base-learner is chosen to be added to the current additive predictor $\eta_{\theta_k}^{[m]}$ via a small fixed step-length (typically chosen to be 0.1):

$$\frac{\partial}{\partial \eta_{\theta_k}} \ell(\mathbf{y} | \theta) \Big|_{\theta = \hat{\theta}^{[m]}} \xrightarrow[\text{base-learner}]{\text{update best}} \hat{\eta}_{\theta}^{[m+1]} \xrightarrow[k=1, \dots, K]{g_k^{-1}} \hat{\theta}^{[m+1]} \tag{3}$$

This iteration process is executed until the final (optimal) stopping iteration m_{stop} is reached. The boosting algorithm is usually stopped before convergence to avoid overfitting and to improve the prediction performance on test data with shrinkage of effect estimates towards zero. Resampling techniques like cross-validation or bootstrapping are typically used to tune the stopping iteration m_{stop} based on the predictive performance, when no additional validation set for tuning m_{stop} is available.

Similar to most other recent methodological works on multi-dimensional boosting (Strömer et al. 2022, 2023; Hans et al. 2023; Griesbach et al. 2023; Stöcker et al. 2021), we use the so-called *non-cyclic* variant (Thomas et al. 2018) for iteratively updating the previous estimate in the boosting algorithm. Thereby, only the update (base-learner fit) leading to the best overall improvement over all distribution parameters is actually executed and all other additive predictors remain without updates for this specific iteration. This leads to a relatively fast one-dimensional tuning process for m_{stop} , as it is also the case in regular model-based boosting. If there are only updates for the location parameter this could reduce the GAMLSS model to a GAM. In comparison, the older *cyclic* variant (Mayr et al. 2012) applies a grid search and updates all parameters one

after another within one iteration. Typically, the non-cyclic variant leads basically to the same results but substantially reduces the computation time for tuning (Thomas et al. 2018). Both variants are implemented in the R add-on package **gamboostLSS** (Hofner et al. 2016), which builds up on the **mboost** (Hofner et al. 2014) package for model-based boosting.

2.2 Robust boosting for GAMLSS

One way to achieve robust model fitting under potentially contaminated data (particularly for the response values) is to apply some kind of *trimming* of extreme observations to control the influence of single observations. This general concept is widely used in robust statistics, starting with the *trimmed mean* (Maronna et al. 2019; Lugosi and Mendelson 2019), but also for more complex methodologies e.g., for high-dimensional regression (Alfons et al. 2013; Speller et al. 2022). In the case of continuous distributions, in Rigby et al. (2019) a pre-specified amount of extreme observations is trimmed to marginal quantiles towards the center of the outcome distribution. This basically leads to a mixed distribution (with spikes at the quantiles) before model fitting, which can be adjusted for by a bias correction to preserve Fisher consistency.

Another robustification approach through different robust evaluation functions (Eguchi and Kano 2001) on the log-likelihood level to reduce influence of single response observations was adapted and proposed for the GAMLSS framework by Aeberhard et al. (2021). It is implemented via the *log-logistic* function within the R package **GJRM**, which can be used as an extension to the classical **gamlss** package to fit GAMLSS. However, it is limited to smaller numbers of covariates, also in comparison to the number of observations.

To apply robust regression also for larger numbers of covariates up to high-dimensional cases, where the number of covariates p may exceed the sample size n , we propose a robust gradient boosting approach to fit GAMLSS. Similar to Aeberhard et al. (2021), we consider the robustified log-likelihood function $\tilde{\ell}_c$ as a direct penalisation of Eq. (2) by the *log-logistic* function ρ_c

$$\tilde{\ell}_c(\mathbf{y} | \boldsymbol{\theta}) = \sum_{i=1}^n \rho_c(\ell(y_i | \boldsymbol{\theta}_i)), \quad (4)$$

leading to a similar optimisation problem as before, where the log-likelihood is replaced by its robustified version:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \tilde{\ell}_c(\mathbf{y} | \boldsymbol{\theta}) \quad (5)$$

The *log-logistic* function ρ_c is defined by

$$\rho_c(z) := \log\left(\frac{1 + \exp(z + c)}{1 + \exp(c)}\right) \text{ with } c > 0 \quad (6)$$

with derivative

$$\rho'_c(z) = \frac{\partial}{\partial z} \rho_c(z) = \frac{\exp(z + c)}{1 + \exp(z + c)}. \tag{7}$$

The log-logistic function and its derivative are illustrated in Fig. 1 for different values of the robustness constant c .

For each $c > 0$, the log-logistic function ρ_c is twice continuously differentiable on \mathbb{R} , convex and bounded from below, limiting the impact of observations with small log-likelihood values on the fitting process. Positive log-likelihood values remain nearly unchanged. Negative log-likelihood values are trimmed towards zero so that the influence on the overall likelihood is bounded for single observations. A larger value of c leads to fewer observations whose influence get restricted, while asymptotically, for $c \rightarrow \infty$, it corresponds to the original, non-robust log-likelihood procedure. Smaller c values have the opposite effect and lead to a sharper cut-off. The slope of ρ_c is monotonically increasing as can be seen in Fig. 1, with a slope close to 0 for most negative values. In $z = -c$ the derivative ρ'_c reaches its point of inflection.

In the context of gradient boosting, the most important part of the fitting procedure is the computation of the partial derivatives of the (robustified) log-likelihood for each additive predictor η_{θ_k} :

$$\frac{\partial}{\partial \eta_{\theta_k}} \tilde{\ell}_c(y_i | \theta_i) = \rho'_c(\ell(y_i | \theta_i)) \cdot \frac{\partial}{\partial \eta_{\theta_k}} \ell(y_i | \theta_i), \quad i = 1, \dots, n \tag{8}$$

The derivative ρ'_c is bounded on the unit interval $[0, 1]$, which means that the gradient from the robustified log-likelihood in Eq. (8) is the log-likelihood gradient (3) multiplied by weights $w_i = \rho'_c(\ell(y_i | \theta_i)) \in (0, 1)$ for $i = 1, \dots, n$, downweighting

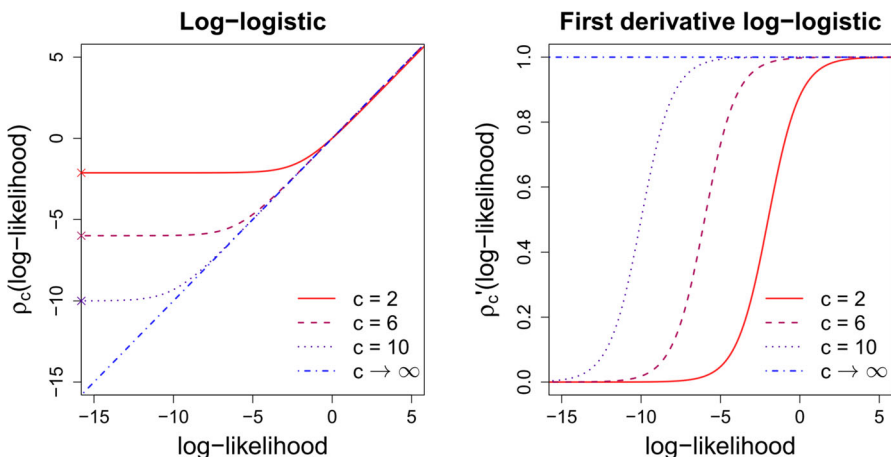


Fig. 1 Log-logistic function ρ_c (left) and its derivative ρ'_c (right) for $c \in \{2, 6, 10\}$ and for the limiting case $c \rightarrow \infty$. The infimum of ρ_c is marked with a cross in corresponding colour on the y-axis, showing how slow ρ_c decays for smaller log-likelihood values. For the limiting case $c \rightarrow \infty$, ρ_c is the identity function, resulting in the original log-likelihood function. Compare figure to (Aeberhard et al. 2021, Web Appendix C, Figure S1)

observations with extreme (negative) log-likelihood values. Therefore, the choice of the robustness constant c entails a trade-off between robustness for corrupted data and efficiency for uncorrupted data.

For positive values of c , there is a natural lower bound of ρ_c given by the infimum over all log-likelihood values $z \in \mathbb{R}$ (see Fig. 1), which can be calculated by considering the following limit:

$$b_{\text{low}}(c) := \inf_{z \in \mathbb{R}} \rho_c(z) = \lim_{z \rightarrow -\infty} \log\left(\frac{1 + \exp(z + c)}{1 + \exp(c)}\right) = -\log(1 + \exp(c)) \quad (9)$$

The value $b_{\text{low}}(c)$ is negative for all $c > 0$. This means that all log-likelihood values, which have lower values than this boundary, are at least lifted to $b_{\text{low}}(c)$. Since the scale of the log-likelihood itself is depending on the statistical model and on the data, this should also be taken into account for the choice of the robustness constant c . As already discussed before, the value of c determines the robustness of the final model. We opt for a choice of c , which is simple to interpret: we propose to use an intercept model with maximum likelihood estimates for all distributional parameters exclusively based on the full response sample \mathbf{y} to generate initial log-likelihood values. Under an assumed amount τ of corruption of the response data, the quantile q_τ gives an upper bound for corrupted observations. For the general distribution case this quantile is given by:

$$q_\tau := \text{quantile}_\tau\left(\ell(\mathbf{y} | \boldsymbol{\theta}(\mathbf{y}) = \hat{\boldsymbol{\theta}}_{\text{offset}}(\mathbf{y}) = \hat{\boldsymbol{\theta}}^{[0]}(\mathbf{y}))\right) \quad (10)$$

If we now equate the τ quantile of the log-likelihood values q_τ with the lower bound b_{low} , we can solve for c_τ , which can be used for our model fit – guaranteeing that all log-likelihood values smaller than q_τ are bounded by our method:

$$q_\tau = b_{\text{low}}(c_\tau) = -\log(1 + \exp(c_\tau)) \Leftrightarrow c_\tau = \log(\exp(-q_\tau) - 1) \quad (11)$$

Note that Eq. (11) can only be applied for $q_\tau < 0$, which is most likely true in most practical cases. Exceptions for instance are distributions with very small variances, and are captured via an appropriate boundary within our implementation.

In cases where no prior information on the amount of corrupted observations is available (which might be the case in most practical settings), we recommend to use the default value of $\tau = 0.05$. This means that 5% of the observations have lower log-likelihood values than b_{low} , irrespective of the particular GAMLSS distribution. Exemplary, for an assumed amount of corruption $\tau = 0.05$ the intercept model of the Gaussian distribution leads to the quantile

$$q_{0.05} = \text{quantile}_{0.05}(\ell(\mathbf{y} | \mu = \text{mean}(\mathbf{y}), \sigma = \text{sd}(\mathbf{y}))), \quad (12)$$

with offset values for $(\mu = \hat{\theta}_1 = \hat{\eta}_{\theta_1}, \sigma = \hat{\theta}_2 = \log(\hat{\eta}_{\theta_2}))$ as the sample mean and the sample variance from the response \mathbf{y} . These are the maximum likelihood estimators for both parameters resulting in a robustness constant $c_{0.05}$.

Example code for our approach can be found in the supplement. The implementation and the code to reproduce the simulations is available on GitHub: <https://github.com/JSpBonn/RobustGAMLSS>.

3 Simulations

The specific goals of our simulation study were to investigate

- A: how the robust boosting approach behaves under uncorrupted data situations in comparison to (non-robust) classical boosting for GAMLSS,
- B: in which data situations the robust methods are beneficial, but also limited in its usage facing corrupted data of different types,
- C: how sensitive the results are with respect to the choice of τ corresponding to the robustness constant c_τ ,
- D: and how the robust boosting method performs in comparison to the original robust fitting procedure for GAMLSS of Aeberhard et al. (2021) in low-dimensional settings.

Especially for high-dimensional settings we also investigated the performance of the boosting approaches regarding variable selection and compared the computational runtime with the original version for the low-dimensional settings.

3.1 Settings

We considered a low-dimensional setting with $p = 5$ and a high-dimensional setting with $p = 1000$ explanatory variables with the same number of observations $n = 1000$. The general structure of a Gaussian distributed outcome y_i for $i \in \{1, \dots, 1000\}$ was given by

$$y_i \sim \mathcal{N}\left(\mu_i = 1 + 2 \cdot x_{i1} - 1 \cdot x_{i2}, \sigma_i^2 = \exp(0.5 - 0.25 \cdot x_{i1} + 0.5 \cdot x_{i3})^2\right),$$

while all other covariates remained uninformative. We considered a *Toeplitz* correlation structure with $\rho = 0.5$ for neighbouring covariates, i.e., \mathbf{x}_i followed a multivariate Gaussian distribution $\mathcal{N}_p(\mathbf{1}, \Sigma)$ with covariance matrix entries $\sigma_{i,j} = 0.5^{|i-j|}$, resulting in diagonal entries $\sigma_{i,i} = 1$ and, exemplary, correlation $\sigma_{2,5} = 0.5^3$ between covariates \mathbf{x}_2 and \mathbf{x}_5 .

We simulated different amounts of corrupted response observations $\pi \in \{0\%, 5\%, 10\%, 15\%, 20\%\}$ by adding or subtracting $4 \cdot \text{sd}(\mathbf{y})$ to generate a *symmetric* error (see Fig. 2 for an illustration). Furthermore, we also simulated a *skewed* corruption, where all error entries were added with their absolute values as one-sided bias. We applied Eqs. (10) and (11) for $\tau \in \{0.01, 0.05, 0.10\}$ to specify the corresponding robustness constants c_τ .

The same data generating process was used for a *skewed* corrupted Gamma distributed outcome y_i with expected value $\mu = \exp(1 + 1.5 \cdot x_{i1} - 0.75 \cdot x_{i2})$ and a variance of μ^2/σ with $\sigma = \exp(0.5 - 0.25 \cdot x_{i1} + 0.5 \cdot x_{i3})^2$.

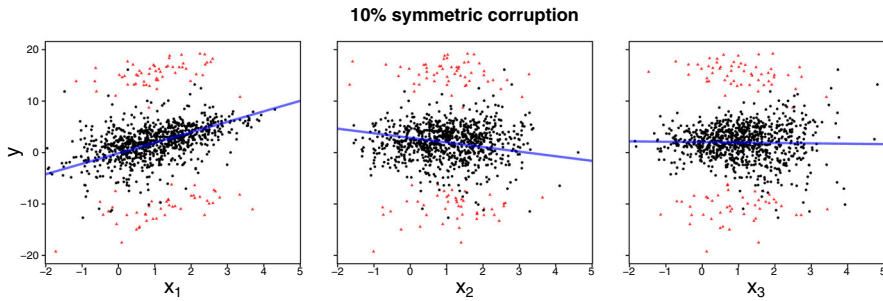


Fig. 2 Illustration of the low-dimensional simulation setting with $n = 1000$ samples and $p = 5$ covariates. The response observations y_i follow a Gaussian distribution $\mathcal{N}(\mu_i = 1 + 2x_{i1} - 1x_{i2}, \sigma_i^2 = \exp(0.5 - 0.25x_{i1} + 0.5x_{i3})^2)$ conditioned on three informative covariates x_1, x_2, x_3 . The black dots represent non-corrupted response observations, while 10% of them were symmetrically corrupted (red triangles) by randomly subtracting or adding $4 \cdot \text{sd}(y)$ from the uncorrupted responses y . The blue line corresponds to a multiple linear regression model between x_1, \dots, x_5 and y based on the full sample adjusted on the means of the remaining 4 covariates, respectively

All individual explanatory variables were considered as component-wise linear base-learners within the boosting algorithms for each distribution parameter. The stopping iteration m_{stop} as the main tuning parameter of the boosting models was optimised in a resource-efficient way based on a validation data set of size $n_{\text{validation}} = 1000$ (which was corrupted in the same way as the training data), by minimising the mean empirical risk. To compare the different models and address questions A to D, we measured the predictive performance on an uncorrupted test data set of size $n_{\text{test}} = 1000$ via the negative log-likelihood values of a Gaussian or Gamma distribution, respectively, at the optimal stopping iteration.

As additional performance measures, we also considered the mean absolute deviation of the estimates of η_μ and η_σ to their true values, the selected variables in the final models (*true* and *false positive rate*), the final stopping iteration, also in detail for both parameters separately (due to the usage of the *non-cyclic* updating method), the computational runtime and in the specific case of the robustified gradient boosting approach also the c_τ values. All simulations were performed $B = 100$ times using R version 4.0.5. Additional results of the simulation study (e.g., for different amounts of corruption and boundary values for c) can be found in the supplement.

3.2 Results

An overview of the prediction performance is given in Table 1 based on the average negative log-likelihood (NLL) values over $B = 100$ simulation runs. The classic boosting model shows low NLL values in all non-corrupted settings – the lowest for all three low-dimensional cases, the second lowest behind the robust boosting approach for $\tau = 0.01$ in both high-dimensional Gaussian cases and lowest for the high-dimensional Gamma setting. Larger choices of a quantile τ in uncorrupted settings lead to an increasing loss in prediction accuracy. As a summary for question A we can conclude that there is typically a loss in prediction accuracy for non corrupted

Table 1 Mean negative log-likelihood (NLL) values and standard deviation based on $B = 100$ simulation runs for different amounts of corruption (symmetric and skewed for the Gaussian distribution and skewed for the Gamma distribution) for the considered methods in a low-dimensional ($p = 5$ and $n = 1000$) and a high-dimensional ($p = 1000$ and $n = 1000$) setting. All models are evaluated on an uncorrupted test data set of size $n_{\text{test}} = 1000$. For each setting, the lowest NLL value is printed in bold, corresponding to the method with the best prediction performance

Gaussian:						
Method		Amount of symmetric corruption				
		0%	5%	10%	15%	20%
low-dim.	<i>Classic Boosting:</i>	2176.44 (30.93)	2440.73 (32.75)	2615.32 (32.59)	2765.44 (46.38)	2867.05 (43.59)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2178.07 (32.30)	2180.14 (30.27)	2185.85 (29.80)	2192.14 (31.59)	2204.76 (65.93)
	$\tau = 0.05$	2212.16 (42.60)	2184.69 (34.95)	2179.22 (32.37)	2180.47 (32.21)	2181.73 (32.03)
	$\tau = 0.10$	2262.18 (53.98)	2250.88 (50.46)	2202.20 (39.63)	2187.76 (35.37)	2185.40 (33.96)
<i>Robust ML:</i>	2177.16 (32.66)	2177.29 (32.09)	2181.48 (31.78)	2187.79 (31.73)	2198.73 (32.20)	
high-dim.	<i>Classic Boosting:</i>	2222.98 (33.91)	2628.86 (26.83)	2754.30 (29.60)	2849.43 (27.55)	2928.72 (28.03)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2221.78 (35.39)	2259.44 (56.60)	2401.24 (97.03)	2532.50 (106.82)	2646.16 (98.03)
	$\tau = 0.05$	2256.04 (47.98)	2225.49 (38.62)	2215.21 (34.90)	2215.79 (35.15)	2217.35 (39.00)
	$\tau = 0.10$	2315.24 (62.62)	2303.61 (59.21)	2243.23 (47.00)	2224.53 (40.03)	2222.29 (38.93)
Gaussian:						
Method		Amount of skewed corruption				
		0%	5%	10%	15%	20%
low-dim.	<i>Classic Boosting:</i>	2176.44 (30.93)	2443.10 (32.67)	2621.84 (28.98)	2757.22 (29.12)	2866.06 (29.47)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2178.07 (32.30)	2180.59 (30.38)	2185.69 (30.33)	2188.16 (31.13)	2192.36 (31.78)
	$\tau = 0.05$	2212.16 (42.60)	2185.18 (34.93)	2181.22 (33.28)	2183.32 (33.35)	2186.20 (34.13)
	$\tau = 0.10$	2262.18 (53.98)	2244.35 (49.95)	2204.66 (40.59)	2194.32 (37.04)	2194.24 (37.15)
<i>Robust ML:</i>	2177.16 (32.66)	2177.26 (32.07)	2181.12 (31.79)	2186.52 (31.84)	2196.65 (31.44)	
high-dim.	<i>Classic Boosting:</i>	2222.98 (33.91)	2624.11 (39.53)	2758.18 (30.48)	2863.78 (27.82)	2951.13 (28.13)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2221.78 (35.39)	2275.30 (44.39)	2348.28 (86.01)	2475.76 (104.90)	2632.52 (110.42)
	$\tau = 0.05$	2256.04 (47.98)	2241.66 (39.95)	2237.98 (37.22)	2240.63 (36.53)	2248.76 (39.91)
	$\tau = 0.10$	2315.24 (62.62)	2315.00 (60.03)	2261.77 (47.97)	2245.77 (44.00)	2245.74 (41.15)
Gamma:						
Method		Amount of skewed corruption				
		0%	5%	10%	15%	20%
low-dim.	<i>Classic Boosting:</i>	2590.05 (50.36)	3002.72 (53.61)	3154.26 (48.03)	3251.65 (45.37)	3327.47 (44.25)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2590.47 (50.91)	2591.19 (50.26)	2592.79 (49.18)	2595.64 (49.31)	2619.67 (120.00)
	$\tau = 0.05$	2606.88 (54.34)	2597.05 (52.90)	2596.48 (52.07)	2596.13 (52.15)	2595.86 (52.05)
	$\tau = 0.10$	2640.63 (58.32)	2613.19 (55.32)	2599.97 (53.19)	2600.11 (53.19)	2599.69 (53.08)
<i>Robust ML:</i>	3210.59 (47.93)	3278.21 (46.55)	3334.76 (45.24)	3384.58 (44.11)	3430.72 (43.36)	
high-dim.	<i>Classic Boosting:</i>	2607.30 (49.10)	3050.23 (67.59)	3208.94 (59.98)	3309.64 (57.31)	3389.79 (54.64)
	<i>Robust Boosting:</i>					
	$\tau = 0.01$	2608.34 (49.31)	2610.48 (49.21)	2629.87 (70.15)	2870.85 (250.12)	3318.07 (131.87)
	$\tau = 0.05$	2635.44 (56.55)	2623.83 (52.78)	2625.44 (54.33)	2627.88 (55.09)	2632.31 (55.56)
	$\tau = 0.10$	2676.26 (66.78)	2646.50 (58.00)	2629.81 (54.73)	2631.92 (56.26)	2636.40 (56.91)

data when using the robust boosting approach especially for higher τ quantiles, while for the choice $\tau = 0.01$ the loss in accuracy is negligible.

On the other hand, already for a small amount of corruption of 5% of observations, classical boosting falls short of all robust models, even in comparison to the NLL values of robust boosting for 0% corruption, and performs increasingly worse for larger amounts of corruption. In the low-dimensional case, this is particularly due to the incorporation of corrupted data points in the fitting process, while in the high-dimensional case, the classical boosting algorithm tends to stop quite early to avoid overfitting (cf. supplement Table S8 and S16).

Regarding question B, the compact answer would be that all robust boosting models are beneficial for all considered amounts of corruption $> 0\%$. In detail, there is a tendency for $\tau = 0.01$ to perform better for smaller amounts, but in more extreme situations for the high-dimensional case and for amounts of corruption close to 20% it will not be robust enough anymore and its performance gets closer to classic boosting (while still being better). In such extreme cases, the quantile with $\tau = 0.10$ has its greatest advantages. The intermediate choice of $\tau = 0.05$ incorporates both beneficial behaviours and does not induce a large loss in prediction accuracy for no or smaller amounts of corruption, but still yields robust results for higher amounts in all simulation settings.

The robustification via the log-logistic function in combination with the idea of choosing the robustness constant c_τ in a data-riven way (based on reasonably low quantile values τ) leads to overall good performances of the robust boosting models. The comparison of different τ values shows limited differences in their NLL values. Still, when considering question C concerning the sensitivity regarding the choice of τ , it seems better to choose one of the smaller quantiles, for which the results stay robust until confronted with extreme corruption amounts ($\tau = 0.01$ and 20%, especially for high-dimensional data), while performing similar to the classical boosting when confronted with non-corrupted data.

Additional simulation results in the supplement to the paper show that the overall behaviour of all models had similar tendencies independently of the kind of corruption (symmetric or skewed), the dimensionality (low- or high-dimensional) and the distribution (Gaussian or Gamma). In particular, results regarding the mean absolute value of the estimated parameters and the variable selection properties are also in line with the performance regarding the NLL values (cf. Table S2, S3 and Table S5, S6 for the Gaussian and S12 and S14 for the Gamma setting of the supplement, respectively). Especially in the high-dimensional settings, the robust boosting models result in larger true positive rates than the classical boosting model when the data is corrupted by any amount.

Regarding question D, the comparison with the robust penalised maximum likelihood (robust ML) approach of Aeberhard et al. (2021) was only possible for the low-dimensional settings. There, however, this approach resulted in a very competitive performance regarding NLL values for the Gaussian setting. The initially chosen robustness constant $c_{0.05}$ based on the $\tau = 0.05$ quantile is optimised during their fitting approach, which comes at the cost of longer computation runtimes (cf. Table S7, S15). Within the simulations for the Gamma distribution the NLL values are increased in comparison to the boosting approaches. Furthermore, the robust penalised ML

approach is not applicable to high-dimensional settings and does not incorporate data-driven variable selection as the proposed robust boosting algorithm.

4 Biomedical applications

To demonstrate the flexibility and adaptability of gradient boosting due to its modular structure in combination with the benefits of robust model-fitting for GAMLSS, we present three application examples from three different biomedical research fields. Additionally to the Gaussian distribution, we also consider the Gamma distribution for modelling positive right-skewed data. We use the same parameterization as for the simulations, where the expected value is equal to μ and the shape parameter σ leads to a variance of μ^2/σ (with log-link for both parameters). For all examples, we chose the robustness constant c in the same way as in the simulations via Eqs. (10) and (11) with $\tau = 0.05$ for all applications and distributions. Additional information including runtimes are presented in the supplement Tables S17-S21.

4.1 Functional magnetic resonance brain imaging

When modelling brain activity parameters based on functional magnetic resonance brain imaging data, the assumption of homoscedasticity can be too restrictive (Landau et al. 2004). Furthermore, measurements can easily be influenced, e.g. via movement of the patient during testing, but also through physiological conditions like nearby veins. In such situations, robust regression methods with the capability of simultaneously fitting multiple parameters while accounting for potential heteroscedasticity are particularly suited.

We focus on the same application as Aeberhard et al. (2021) (based on Landau et al. 2004), who modelled brain activity via the *median Fundamental Power Quotient* (medFPQ) and reported both a robust and a classical GAMLSS. Given is the measured resonance of an experimental stimulus of a human brain subject for one 2D slice through the dorsal cerebral cortex for $n = 1567$ brain voxels. The dataset is publicly available via the R package **gamair**, where three replicates of FPQ for each voxel are given by their median resulting in a right-skewed, non-negative distribution of medFPQ. Note that no extreme observations (potential outliers) were excluded, which stresses the general idea of robust model fitting, for instance in contrast to (Wood 2006, p.228).

In Aeberhard et al. (2021) the response variable medFPQ was modeled via a Gamma distribution in dependence of the spatial covariates x_1 and x_2 , which represent the location of each voxel within the 2D brain slice. We fitted similar Gamma regression models, but using gradient boosting for GAMLSS based on the classical and new robust versions in combination with a spatial base-learner (for x_1 and x_2) for best comparability on the full data set. Therefore, both boosting approaches were tuned using 25-fold bootstrapping, which resulted in optimal stopping iterations $m_{\text{stop}} = 1480$ for the classical Gamma fit and $m_{\text{stop}} = 197$ for the robust version.

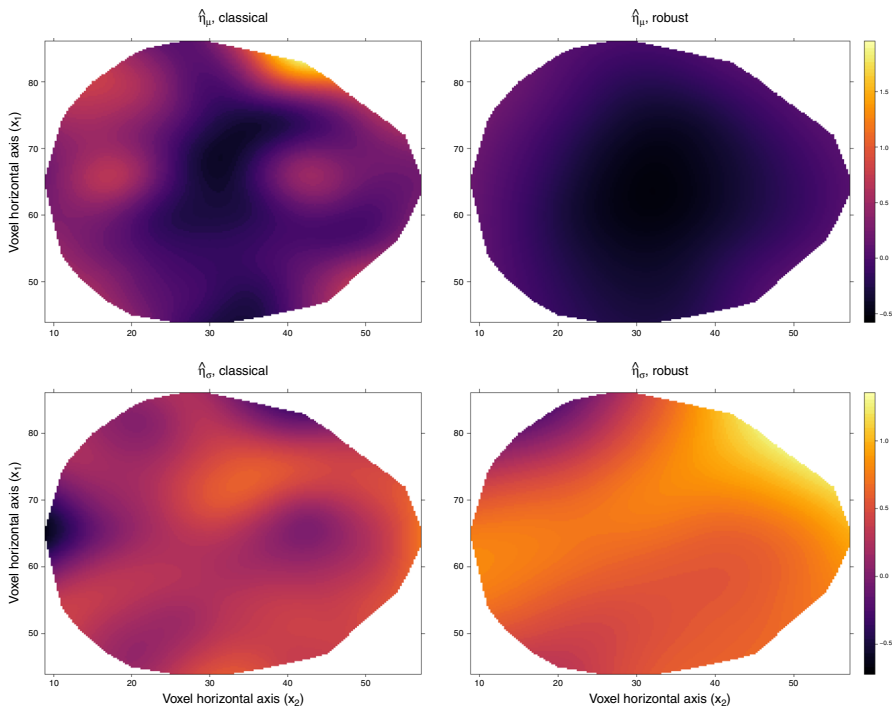


Fig. 3 Shown are the additive predictors $\hat{\eta}_\mu$ and $\hat{\eta}_\sigma$ for the classical boosting GAMLSS fit (left) and the new robust boosting approach (right), where the main brain activity (medFPQ) was modelled by a Gamma distribution depending on the location of voxels x_1 and x_2 . Early stopping with a 25-fold bootstrap approach leads, for the classical variant, to $m_{\text{stop}} = 1480$ (963 updates μ , 517 for σ) and, for the robust version, to $m_{\text{stop}} = 197$ (130 updates μ , 67 for σ). Note that, in comparison to (Aeberhard et al. 2021, Figure 5), a different parameterization for σ was used, which inverts and rescales the colour palette for $\hat{\eta}_\sigma$

Figure 3 displays the results of the predicted medFPQ values for both boosting approaches as coloured deviations for different brain activities. It can be observed that the robust approach led to much smoother spatial effect estimates for both parameters. The classical boosting approach, in contrast, fitted rougher and more detailed spatial effects. This may also be due to the fact that the boosting algorithm stopped much later for the classical approach (a brain image with converged effect estimates can be found for both approaches in supplement Figure S3). For the location parameter (η_μ), the classical model shows more heterogeneity and a larger effect in the left frontal lobe of the brain, while the robust model updated more regions for the shape parameter (η_σ), especially in the frontal lobe (stronger, but similar to the classical one). Overall, for this example our robust boosting results for GAMLSS are comparable to the ones presented by Aeberhard et al. (2021) (when taking the different parameterizations into account). An additional direct comparison between the two robust approaches (also regarding computation time) was included in the supplement Figure S4 using Gaussian models.

4.2 Thyroid hormone levels of a general adult population

Thyroid hormones (TH) play a major role in regulating basal metabolism in humans. TH synthesis is regulated by feedback mechanisms: decreased thyroid hormone levels lead to increased synthesis of hypothalamic thyrotropin-releasing hormone which increases the secretion of thyroid-stimulating hormone (TSH). TSH stimulates the production of thyroid hormones, thyroxine (T4) and triiodothyronine (T3). In order to examine the relationship between thyroid function and hematological parameters, we analyzed the data from the *A-Estrada Glycation and Inflammation Study* (AEGIS), a cross-sectional study conducted in the municipality of A-Estrada, in North-western Spain. An outline of the project can be found at <https://www.clinicaltrials.gov> (code: NCT01796184) and details also in Alende-Castro et al. (2019).

Blood samples were taken from $n = 1516$ participants to describe the relationship between blood cell components and thyroid function. In total, we considered $p = 33$ potential explanatory variables for $n = 1151$ complete observations, which were not previously diagnosed with any kind of thyroid dysfunction. To model the thyroid hormone level of T3 as the response variable (see Fig. 4), we boosted a Gaussian GAMLSS with all available candidate variables based on the classical boosting approach as well as the new robust variant. Besides fitting a model on the full data set, we performed 10-fold cross-validation to compare the selection frequencies of the variables in the boosting models and the prediction accuracy via the log-likelihood on test data. To avoid further reduction of the sample size for tuning via a validation set, the stopping iteration m_{stop} of boosting was determined by 25-fold bootstrap for all models (separately on each of the training folds of the 10-fold cross-validation). For comparison, we also fitted the original robust approach by Aeberhard et al. (2021) for the full model without variable selection.

While boosting the non-robust Gaussian model selected 5 variables for the location parameter μ and 6 variables for the dispersion parameter σ , the new robust boosting version stopped later and selected 15 predictors for the location, but only one variable (*transferrin saturation*) for σ , which was also included relatively late in the final model (see supplement Table S11 for a complete overview of selected variables). From a

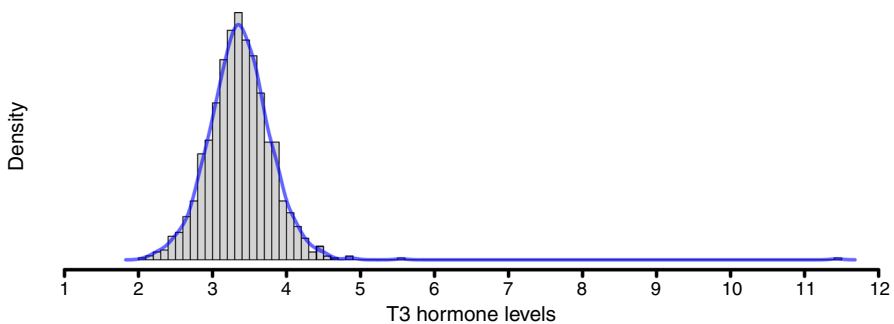


Fig. 4 Density of triiodothyronine (T3) hormone levels from $n = 1151$ participants of the AEGIS study, which were part of our analysis. The blue line is based on a non-parametric kernel density estimator

biological point of view, these results are consistent with previous studies in which small differences in thyroid function are associated with significant differences in a range of clinical parameters. Anemia and other blood abnormalities are common in thyroid function abnormalities. Iron metabolism (hemoglobin, ferritin, transferrin) is also very intricately connected to thyroid hormone metabolism (Bremner et al. 2012).

Regarding prediction accuracy the classical boosting fit yielded slightly favorable results, with median negative log-likelihood values of 55.34 (IQR = 16.93) compared to 65.49 (IQR = 26.26) for boosting the robust version, which might be also related to the later stopping (m_{stop} was on average 27.50 (sd = 12.55) for the normal Gaussian model and 46.10 (sd = 7.25) for the robust one). Similar likelihood results were also found for the full model by the original robust GAMLSS approach (55.22 with IQR = 22.96). In general, there seems to be only one unconditioned extreme outlier (Fig. 4), which can also be handled well by the non-robust boosting approach given the relatively high sample size. But also the robust boosting variant does not yield a large loss in prediction accuracy when applied to quasi non-corrupted data (cf. Table 1). The most frequently selected variables for each approach were also the ones from the final model fitted on the full data.

4.3 Protein expression data via NCI-60 cancer cell panel

High-dimensional data situations are very common in the field of genetics, where the number of potential covariates often exceeds the number of observations. We illustrate the performance of our approach for this situation via the NCI-60 data set, which includes only $n = 59$ human cancer cell lines in comparison to $p = 14951$ gene expression measurements as potential predictors. The data set is available at <https://discover.nci.nih.gov/cellminer/>.

We followed the same data processing steps as in Speller et al. (2022), Sun et al. (2020) and considered the positive and highly right-skewed *KRT19* protein antibody array from the “Protein: Antibody Array DTB” set as response variable (Sun et al. 2020). The *KRT19* protein data is an example with extreme kurtosis and some (extreme) outliers (the median value is 1.10, while the 75% quantile is 32.14 with values up to 257), implying a challenging modelling task which is particularly interesting for robust regression approaches.

We applied classical gradient boosting for GAMLSS using the Gamma distribution to compare it with our robust boosting approach. The predictor variables are given as the gene expression measurements, which we included via separate linear base-learners for all boosting models. Besides considering both classical and robust boosting on the full data set, we additionally performed leave-one-out-cross-validation (LOOCV); in this case with $n = 59$, each model was fitted on 58 observations as training data and the remaining observations were used as test data to evaluate the negative log-likelihood of the original Gamma distribution. To avoid forming a validation set, all models were stopped early at their optimal stopping iteration via an inner 25-fold bootstrap on the respective training observations. To investigate the variable selection behaviour of boosting, we also computed selection frequencies on the 59 LOOCV training folds for all potential predictor variables.

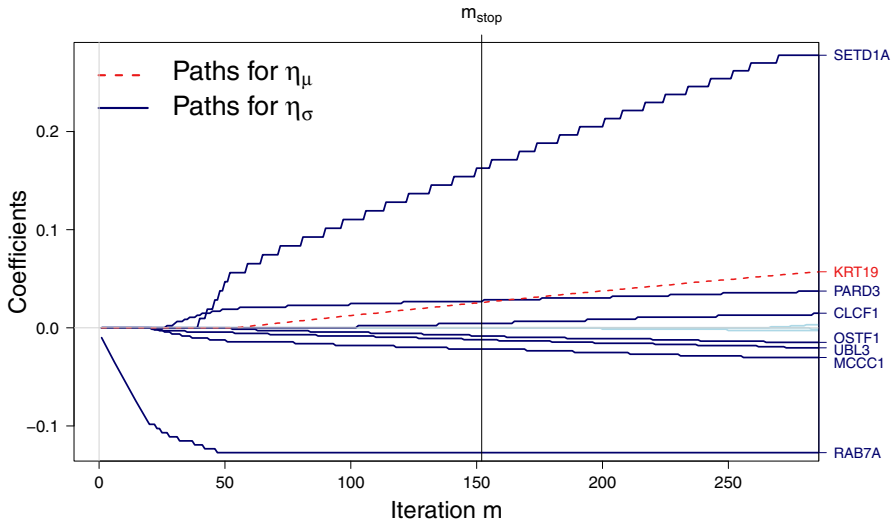


Fig. 5 Coefficient-paths for robust boosting for GAMLSS of the NCI-60 data. One gene was selected for the parameter μ (dashed red path) and 7 were selected for σ (darkblue paths) before the algorithm stopped at $m_{\text{stop}} = 152$. Coefficients which are updated the first time after this iteration and are hence not included in the final model are coloured lightblue

The robust model fit on the full data resulted in an optimal stopping iteration of $m_{\text{stop}} = 152$, with only one selected predictor—the *KRT19* gene—for the location parameter μ and 7 selected genes for the shape parameter σ (see Fig. 5 for an illustration for the coefficient-paths). The early stopping of the boosting algorithm results in shrinkage of effect estimates, which is nicely visible for *SETD1A*. Updated variables after iteration m_{stop} are not included in the final model (see Fig. 5, lightblue paths deviate only from zero for iterations $m > m_{\text{stop}}$). The only updated base-learner for η_{μ} is the one referring to the *KRT19* gene, which is included relatively late, suggesting that the default offset value for μ was a good starting point and that generally more information may be contained for the shape parameter σ .

The non-robust Gamma regression, in contrast, resulted in an early stopping of the boosting algorithm at iteration $m_{\text{stop}} = 1$. As a result, only the intercept for σ was updated once, while μ stayed at its offset value. This illustrates an interesting property of gradient boosting: when updates are non-beneficial for the prediction, the tuned boosting model can stay extremely sparse to avoid overfitting. In case of extreme $p \gg n$ situations with highly skewed data and outliers, this behaviour could also be very reasonable. However, this does not necessarily mean that there are no valid information in the data, as could be observed for the robust approach.

To further assess the performance of both methods, Table 2 summarizes the results of LOOCV regarding prediction accuracy and variable selection. The non-robust boosting variant selected no genes at all and due to early stopping resulted in intercept models. In contrast, the robust approach selected on average two genes for μ , while especially the gene expression of *KRT19* was selected in nearly all cross-validation

Table 2 Overview of LOOCV for all 59 observations of the NCI-60 data set, where GAMLSS were fitted via gradient boosting for the response *KRT19* protein modelled via a Gamma distribution. The selected variables, the mean model size and the mean negative log-likelihood values are given for the boosting models, where the stopping iteration m_{stop} was tuned via 25-fold bootstrapping separately for all splits

Method Parameter	Selected predictors (Frequency out of 59)	Model size mean (sd)	Neg. log-lik. mean (sd)
<i>Robust boosting</i> (Gamma $\tau = 0.05$)		8.95 (1.71)	3.53 (2.09)
μ	KRT19(58), EPS8L2(2)	1.02 (0.13)	
σ	RAB7A(59), UBL3(58), MCCC1(56), SETD1A(55), PARD3(54), OSTF1(47), CLCF1(36), PIK3R1(21), NRN1(17), ALDH3B1(17), VGLL4(16), ADRB2(8), RHOF(8), ZNF432(4), SNX4(2), ANKRA2(1), UFL1(1), DUSP5(1), POLR2L(1), RPS6KA4(1), SAP18(1), PSMB5(1), TERF2IP(1), HNF1B(1), TASP1(1)	7.93 (1.71)	

steps (58). For σ , the results are also in line with the robust model on the full data. The 7 most frequently selected genes in LOOCV were also selected for σ on the full data (cf. Figure 5). Taking the high-dimensionality and the small sample size of $n = 59$ into account, the selection process for the predictors appears very stable for the robust approaches and points to the same small subset of genes for further investigation. Regarding prediction accuracy, the robust GAMLSS fits were also beneficial and yielded lower negative log-likelihood values on the test observations (mean 3.53 (sd = 2.09) in comparison to 4.39 (sd = 1.97)).

5 Discussion

We have proposed a robust boosting approach for generalized additive models for location, scale and shape (GAMLSS, Rigby and Stasinopoulos 2005). It reduces the effects of outliers in the response variable while incorporating variable selection for potentially high-dimensional data. The core of the approach is a robustification of the likelihood via the log-logistic function as recently proposed by Aeberhard et al. (2021). We incorporated the robustified likelihood as the loss function for a component-wise gradient boosting algorithm for GAMLSS (Mayr et al. 2012). To the best of our knowledge, our approach allows for the first time to use a robust boosting approach to fit distributional regression models via GAMLSS. Additionally, we have proposed a quantile-based data-driven method on how to specify the necessary robustness constant in practice.

The results of our simulation study suggest that our approach works well and provides robustness for reasonable amounts of corrupted observations (up to 15-20%), both for symmetric and skewed corruption types. At the same time, when applied on

data without any outliers, the performance only slightly decreases compared to a classical non-robust boosting algorithm. A similar behaviour could also be observed in the three biomedical applications: our robust boosting approach was able to replicate similar results compared to a previous robust analysis for functional magnetic resonance brain imaging, showing more robust effect estimates than classical boosting. However, in a much larger epidemiological data set on thyroid hormone levels with only one extreme outlier, the robust boosting version did not outperform the classical algorithm regarding prediction accuracy. This might be due to the fact that also classical boosting can lead to a relatively robust model fit by imposing stronger regularisation via early stopping. This could also be observed in a more extreme way for the high-dimensional gene expression data from cancer cells: Here, the classical boosting algorithm basically selects only an intercept model while the robust version allows to identify the most influential genes, outperforming the classical boosting algorithm with respect to prediction accuracy.

Our approach also has several limitations: Due to the iterative fitting of the base-learners to the gradient of the loss, statistical boosting algorithms typically cannot provide standard errors for effect estimates. This is also true for the proposed robust variant, making it mostly suitable for exploratory data analysis. There are some workarounds based on permutation tests (Mayr et al. 2017; Hepp et al. 2019) and bootstrapping (Hofner et al. 2016), though they tend to become computationally very demanding for high-dimensional data. Also without these additional resampling measures, boosting and tuning a robust GAMLSS leads to a longer computation time than the classical approach by Aeberhard et al. (2021). The use of boosting for model fitting in low-dimensional settings is hence most favorable only for situations where the variable selection properties or shrinkage of effect estimates are required. Another limitation is that we have tested the robust version only on a limited combination of distributions (Gaussian, Gamma), base-learners (linear, spatial) and focused on the non-cyclic variant. There is no reason to believe why the approach should not work for other GAMLSS distributions, the cyclic update variant and/or base-learners available in the boosting framework (Mayr and Hofner 2018; Hofner et al. 2014) – nevertheless this should be properly tested. Furthermore, even when the event of almost exclusively positive log-likelihood values is unlikely to occur in practical applications, it is possible that our proposed quantile-based selection of the tuning constant fails for extreme data situations. Further research is also warranted on the robustification of base-learners, as our approach is currently limited to outliers in the response variable. Another line of research could also focus on an automated selection of the most appropriate (robust) response distribution based on prediction performance.

In this context, quantile regression (Koenker and Hallock 2001) could also be a natural alternative approach for robust distributional regression. In contrast to GAMLSS, it does not carry the risk of miss-specifying the distribution. However, in quantile regression the focus is not really on the complete conditional distribution, as each quantile is fitted separately, which can lead to crossing of neighboring predicted quantiles. GAMLSS, on the other hand, model the complete distribution and allow for the interpretation of effects directly on location or scale parameters. If quantiles are needed, they can still be computed from the predicted distribution – without the risk of quantile crossing.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s11634-023-00555-5>.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aeberhard WH, Cantoni E, Marra G, Radice R (2021) Robust fitting for generalized additive models for location, scale and shape. *Stat Comput*. <https://doi.org/10.1007/s11222-020-09979-x>
- Alende-Castro V, Alonso-Sampedro M, Vazquez-Temprano N, Tuñez C, Rey D, García-Iglesias C, Sopena B, Gude F, Gonzalez-Quintela A (2019) Factors influencing erythrocyte sedimentation rate in adults: new evidence for an old test. *Medicine*. <https://doi.org/10.1097/MD.00000000000016816>
- Alfons A, Croux C, Gelper S (2013) Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *Annals Appl Stat* 7(1):226–248
- Amato U, Antoniadis A, De Feis I, Gijbels I (2021) Penalised robust estimators for sparse and high-dimensional linear models. *Stat Methods Appl* 30:1–48
- Barrios EB (2015) Robustness, data analysis, and statistical modeling: the first 50 years and beyond. *Commun Statist Appl Methods* 22(6):543–556
- Bremner AP, Feddema P, Joske DJ, Leedman PJ, O'Leary PC, Olynyk JK, Walsh JP (2012) Significant association between thyroid hormones and erythrocyte indices in euthyroid subjects. *Clin Endocrinol* 76(2):304–311. <https://doi.org/10.1111/j.1365-2265.2011.04228.x>
- Bühlmann P (2006) Boosting for high-dimensional linear models. *Ann Stat* 34(2):559–583
- Eguchi S, Kano Y (2001) Robustifying maximum likelihood estimation by psi-divergence. *ISM Res Memo* 802
- Friedman JH (2001) Greedy function approximation: a gradient boosting machine. *Ann Stat* 29(5):1189–1232
- Griesbach C, Mayr A, Bergherr E (2023) Variable selection and allocation in joint models via gradient boosting techniques. *Mathematics* 11(2):411
- Hans N, Klein N, Faschingbauer F, Schneider M, Mayr A (2023) Boosting distributional copula regression. *Biometrics*. <https://doi.org/10.1111/biom.13765>
- Hastie T, Tibshirani R (1990) *Generalized additive models*. Chapman & Hall, London
- Hepp T, Schmid M, Mayr A (2019) Significance tests for boosted location and scale models with linear base-learners. *Int J Biostat*. <https://doi.org/10.1515/ijb-2018-0110>
- Hofner B, Mayr A, Robinzonov N, Schmid M (2014) Model-based boosting in R: a hands-on tutorial using the R package mboost. *Comput Stat* 29:3–35
- Hofner B, Kneib T, Hothorn T (2016) A unified framework of constrained regression. *Stat Comput* 26:1–14
- Hofner B, Mayr A, Schmid M (2016) gamboostLSS: an R package for model building and variable selection in the gamlss framework. *J Stat Softw*. <https://doi.org/10.18637/jss.v074.i01>
- Huber PJ (1981) *Robust statistics*. Wiley, New York
- Ju X, Salibián-Barrera M (2021) Robust boosting for regression problems. *Comput Stat Data Anal*. <https://doi.org/10.1016/j.csda.2020.107065>
- Kneib T, Silbersdorff A, Säfken B (2021) Rage against the mean – a review of distributional regression approaches. *Econom Stat*. <https://doi.org/10.1016/j.ecosta.2021.07.006>
- Koenker R, Hallock KF (2001) Quantile regression. *J Econ Perspect* 15(4):143–156

- Landau S, Ellison-Wright IC, Bullmore ET (2004) Tests for a difference in timing of physiological response between two brain regions measured by using functional magnetic resonance imaging. *J Roy Stat Soc: Ser C (Appl Stat)* 53(1):63–82
- Lugosi G, Mendelson S (2019) Mean estimation and regression under heavy-tailed distributions: a survey. *Found Comput Math* 19(5):1145–1190
- Maronna RA, Martin RD, Yohai VJ, Salibián-Barrera M (2019) Robust statistics: theory and methods (with R). John Wiley & Sons, New York. 2nd ed
- Mayr A, Hofner B (2018) Boosting for statistical modelling: a non-technical introduction. *Stat Model* 18(3–4):365–384
- Mayr A, Fenske N, Hofner B, Kneib T, Schmid M (2012) Generalized additive models for location, scale and shape for high dimensional data—a flexible approach based on boosting. *J Roy Stat Soc: Ser C (Appl Stat)* 61(3):403–427
- Mayr A, Schmid M, Pfahlerberg A, Uter W, Gefeller O (2017) A permutation test to analyse systematic bias and random measurement errors of medical devices via boosting location and scale models. *Stat Methods Med Res* 26(3):1443–1460
- Monti GS, Filzmoser P (2022) Robust logistic zero-sum regression for microbiome compositional data. *Adv Data Anal Classif* 16:301–324. <https://doi.org/10.1007/s11634-021-00465-4>
- Rigby RA, Stasinopoulos DM (2005) Generalized additive models for location, scale and shape. *J Roy Stat Soc: Ser C (Appl Stat)* 54(3):507–554
- Rigby RA, Stasinopoulos MD, Heller GZ, De Bastiani F (2019) Distributions for modeling location, scale and shape: using GAMLSS in R. CRC Press, Boca Raton
- Speller J, Staerk C, Mayr A (2022) Robust statistical boosting with quantile-based adaptive loss functions. *Int J Biostat*. <https://doi.org/10.1515/ijb-2021-0127>
- Stöcker A, Brockhaus S, Schaffer SA, Bv Bronk, Opitz M, Greven S (2021) Boosting functional response models for location, scale and shape with an application to bacterial competition. *Stat Model* 21(5):385–404
- Strömer A, Staerk C, Klein N, Weinhold L, Titze S, Mayr A (2022) Deselection of base-learners for statistical boosting – with an application to distributional regression. *Stat Methods Med Res* 31(2):207–224
- Strömer A, Klein N, Staerk C, Klinkhammer H, Mayr A (2023) Boosting multivariate structured additive distributional regression models. *Stat Med* 42(11):1779–1801. <https://doi.org/10.1002/sim.9699>
- Sun Q, Zhou W, Fan J (2020) Adaptive Huber regression. *J Am Stat Assoc* 115(529):254–265
- Thomas J, Mayr A, Bischl B, Schmid M, Smith A, Hofner B (2018) Gradient boosting for distributional regression: faster tuning and improved variable selection via noncyclical updates. *Stat Comput* 28(3):673–687. <https://doi.org/10.1007/s11222-017-9754-6>
- Wood SN (2006) Generalized additive models: an introduction with R. Chapman & Hall/CRC, London

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.