Hindawi
Computational and Mathematical Methods in Medicine
Volume 2017, Article ID 6083072, 12 pages
https://doi.org/10.1155/2017/6083072

Review Article

An Update on Statistical Boosting in Biomedicine

Andreas Mayr,[1,2] Benjamin Hofner,[3] Elisabeth Waldmann,[1] Tobias Hepp,[1] Sebastian Meyer,[1] and Olaf Gefeller[1]

[1]*Institut für Medizininformatik, Biometrie und Epidemiologie, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany*
[2]*Institut für Statistik, Ludwig-Maximilians-Universität München, Munich, Germany*
[3]*Paul-Ehrlich-Institut, Langen, Germany*

Correspondence should be addressed to Andreas Mayr; andreas.mayr@fau.de

Received 24 February 2017; Accepted 8 June 2017; Published 2 August 2017

Academic Editor: Andrzej Kloczkowski

Statistical boosting algorithms have triggered a lot of research during the last decade. They combine a powerful machine learning approach with classical statistical modelling, offering various practical advantages like automated variable selection and implicit regularization of effect estimates. They are extremely flexible, as the underlying base-learners (regression functions defining the type of effect for the explanatory variables) can be combined with any kind of loss function (target function to be optimized, defining the type of regression setting). In this review article, we highlight the most recent methodological developments on statistical boosting regarding variable selection, functional regression, and advanced time-to-event modelling. Additionally, we provide a short overview on relevant applications of statistical boosting in biomedicine.

1. Introduction

Statistical boosting algorithms are one of the advanced methods in the toolbox of a modern statistician or data scientist [1]. While still yielding classical statistical models with well-known interpretability, they offer multiple advantages in the presence of high-dimensional data as they are applicable in $p > n$ situations with more explanatory variables than observations [2, 3]. Key features in this context are automated variable selection and model choice [4, 5].

The research field embraces the world of statistics and computer science, bridging the gap between two rather different points of view on how to extract information from data [6]: on the one hand, there is the classical statistical modelling community who focus on models *describing* and *explaining* the outcome to find an approximation to the underlying stochastic data generating process. On the other hand, there is the machine learning community who focus primarily on algorithmic models *predicting* the outcome while treating the nature of the underlying process as unknown. Statistical boosting algorithms have their roots in machine learning [7] but were later adapted to estimate classical statistical models [8, 9]. A pivotal aspect of these algorithms is that they incorporate data-driven variable selection and shrinkage of effect estimates similar to classical penalized regression [10].

In a review some years ago [1], we highlighted this evolution of boosting from machine learning to statistical modelling. Furthermore, we emphasized the similarity of two boosting approaches, gradient boosting [2] and likelihood-based boosting [3], introducing *statistical boosting* as a generic term for these algorithms.

An accompanying article [11] highlighted the multiple extension of the basic algorithms towards (i) enhanced variable selection properties, (ii) new types of predictor effects, and (iii) new regression settings. Substantial methodological developments on statistical boosting algorithms throughout the last few years (e.g., stability selection [12]) and a growing community have opened the door to new model classes and frameworks (e.g., joint models [13] and functional data [14]), asking for an up-to-date review on the available extensions.

This article is structured as follows: In Section 2 we shortly highlight both basic structure and properties of statistical boosting algorithms and point to their connections to classical penalization approaches such as the lasso. In

Section 3 we focus on new developments regarding variable selection (including exemplary analysis of gene expression data), which can also be combined with boosted functional regression models presented in Section 4. Section 5 focuses on advanced survival models such as joint modelling; in Section 6 we briefly summarize other relevant developments and applications in the framework of statistical boosting.

2. Statistical Boosting

2.1. From Machine Learning to Statistical Models. The original boosting concept by Schapire [15] and Freund [7] emerged from the field of supervised learning where typically a function is trained based on data with known outcome classes or labels to correctly classify new observations. The aim of the boosting concept is to *boost* (i.e., to improve) the accuracy of weak classifiers (i.e., classifiers with poor correct classification rates) by iteratively applying them to reweighted data. Even if these so called *base-learners* individually only slightly outperform random guessing, the ensemble solution can often be boosted to a perfect classification [16].

The introduction of AdaBoost [17] was the breakthrough for boosting in the field of supervised machine learning, allegedly leading Leo Breiman to praise its performance: *Boosting is the best off-the-shelf classifier in the world* [18].

The main target of classical machine learning approaches is predicting observations y_{new} of the outcome Y given one or more input variables $\mathbf{X} = \{X_1, \ldots, X_p\}$. The estimation of the prediction rule (also called generalization function) is based on an observed sample $(y_1, \mathbf{x}_1), \ldots, (y_n, \mathbf{x}_n)$. However, the focus is not on quantifying or describing the underlying data generating process, but on predicting $\widehat{y}_{\text{new}}$ for new observations x_{new} as accurately as possible. As a consequence, many machine learning approaches (also including the original AdaBoost with trees or stumps as base-learners) can be regarded as black box prediction schemes. Although typically yielding accurate predictions [19], they do not offer much insight into the structure of the relationship between explanatory variables $\mathbf{X}$ and the outcome Y.

Statistical regression models on the other hand particularly aim at describing and explaining the underlying relationship in a structured way. Not only can the impact of single explanatory variables be quantified in terms of variable importance measures [20, 21], but also the actual effect of these variables is interpretable. The work of Friedman et al. [8, 9] laid the groundwork to understand the concept of boosting from a statistical perspective and to adapt the general idea in order to estimate statistical models.

2.2. General Model Structure. The aim of *statistical boosting* algorithms is to estimate and select the effects in structured additive regression models. The most important model class are generalized additive models ("GAM" [22]), where the conditional distribution of the response variable is assumed to follow an exponential family distribution. The expected response is modelled given the observed value $\mathbf{x}$ of one or more explanatory variables using a link function g as

$$g\left(\mathbb{E}\left(Y \mid \mathbf{X} = \mathbf{x}\right)\right) = f\left(\mathbf{x}\right). \tag{1}$$

In the typical case of multiple explanatory variables, the function $f(\mathbf{x})$, which is often called additive predictor, consists of the additive effects of the single predictors:

$$f\left(\mathbf{x}\right) = \beta_0 + f_1\left(x_1\right) + \cdots + f_p\left(x_p\right), \tag{2}$$

where β_0 represents a common intercept and the functions $f_j(x_j)$, $j = 1, \ldots p$, are the individual effects of the variables x_j. The generic notation $f_j(x_j)$ may comprise different types of predictor effects such as classical linear effects, $x_j\beta_j$, smooth nonlinear effects constructed via regression splines, spatial effects, or random effects of the explanatory variable x_j, to name but a few.

In statistical boosting algorithms, like the two approaches described in the following sections, the different effects are estimated by separate base-learners $h_1(\cdot), \ldots, h_p(\cdot)$ (*componentwise boosting* [2]). These base-learners are typically the corresponding simple regression-type prediction functions; for a linear effect, the corresponding base-learner would be a simple linear model: $h_j(x_j) = \beta_0 + \beta_1 x_j$.

2.3. The Generic Structure of Statistical Boosting. For a generic overview on the structure of statistical boosting algorithms, see Box 1. The base-learners are applied one by one, and in every iteration only the best performing base-learner j^* is selected to be updated. The final additive model is hence the sum of all selected base-learner fits.

The main tuning parameter is m_{stop}, the number of boosting iterations that is carried out. In order to avoid overfitting and to ensure variable selection, the algorithm is typically stopped before convergence *(early stopping)*. The selection of m_{stop} is based on the predictive performance evaluated via cross-validation or resampling [23]. This early stopping leads to an implicit penalization [24], similar to the lasso (see Section 2.6).

2.4. Gradient Boosting. In gradient boosting [2, 8], the iterative procedure fits the base-learners $h_1(x_1), \ldots, h_p(x_p)$ one by one to the negative gradient of the loss function $\rho(y, f(\cdot))$, evaluated at the previous iteration:

$$\mathbf{u}^{[m]} = \left(-\frac{\partial}{\partial f}\rho\left(y_i, f\left(\mathbf{x}_i\right)\right)\Big|_{f=\widehat{f}^{[m-1]}(x_i)}\right)_{i=1,\ldots,n}. \tag{3}$$

The loss function describes the discrepancy between the observed outcome y and the additive predictor $f(\mathbf{x}_i)$ and is the target function that should be minimized to get an optimal fit. In case of GAMs, the loss function is typically the negative log-likelihood of the corresponding exponential family. For Gaussian distributed outcomes, this reduces to the L_2 loss $\rho(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$, where the gradient vector $\mathbf{u}^{[m]}$ is simply the vector of residuals $y - f(\mathbf{x})$ from iteration $m-1$ and boosting hence corresponds to refitting of residuals.

In each boosting iteration, only the best-fitting base-learner h_{j^*} is selected based on the residual sum of squares of the base-learner fit

$$j^{*[m]} = \arg\min_{1 \le j \le p} \sum_{i=1}^{n} \left(u_i^{[m]} - \widehat{h}_j^{[m]}\left(x_{ij}\right)\right)^2. \tag{4}$$

Initialization

(1) Start with iteration counter $m = 0$. Initialize the additive predictor $\widehat{f}^{[0]}$ with an offset value. Specify a set of prediction functions as base-learners $h_1(x_1), \ldots, h_p(x_p)$; typically each base-learner is a regression function incorporating one possible candidate variable.

Component-wise fitting of base-learners

(2) Set iteration counter $m := m + 1$.

(3) Fit the base-learners $\widehat{h}_j(\cdot)$, $j = 1, \ldots, p$, one-by-one:

Gradient boosting

Base-learners are fitted to the negative gradient vector of the loss function (e.g. the negative log-likelihood), evaluated at the current additive predictor $\widehat{f}^{[m-1]}$. To ensure small steps, the base-learner fits are multiplied by a small step-length factor ν, $0 \leq \nu \leq 1$: $\widehat{h}_j^{[m]}(\cdot) := \nu \cdot \widehat{h}_j^{[m]}(\cdot)$.

Likelihood-based boosting

Base-learners are estimated via maximizing the overall likelihood, using one step of Fisher scoring with the current additive predictor $\widehat{f}^{[m-1]}$ as offset. To ensure small steps, a penalty term is attached to the likelihood.

Update best performing component

(4) Select the best performing base-learner $j^{*[m]}$:

Gradient boosting

Based on the smallest residual sum of squares with respect to the negative gradient vector.

Likelihood-based boosting

Based on the largest overall likelihood after the update.

(5) Update the additive predictor via the corresponding base-learner:

$$\widehat{f}^{[m]} = \widehat{f}^{[m-1]} + \widehat{h}_{j^*}^{[m]}(x_{j^*})$$

Iteration

Iterate steps (2) to (5) until $m = m_{\text{stop}}$. The parameter m_{stop} is the main tuning parameter, typically selected via resampling procedures.

Box 1: The structure of statistical boosting algorithms.

Only this base-learner h_{j^*} is added to the current additive predictor $f(\cdot)$. In order to ensure small updates, only a small proportion of the base-learner fit (typically the step length is $\nu = 0.1$ [2]) is actually added. Note that the base-learner $h_j(\cdot)$ can be selected and updated various times; the partial effect of variable x_j is the sum of all corresponding base-learners that had been selected:

$$\widehat{f}_j\left(x_j\right) = \sum_m \nu \cdot \widehat{h}_j^{[m]}\left(x_j\right) \mathbb{I}_{j=j^*[m]}. \tag{5}$$

This componentwise procedure of fitting the base-learners one by one to the current gradient of the loss function can be described as *gradient descent in function space* [25], where the function space is spanned by the base-learners. The algorithm effectively optimizes the loss function step by step, eventually converging to the minimum.

Gradient boosting is implemented in the add-on package *mboost* [26] for the open source programming environment R [27], providing a large number of preimplemented loss functions for various regression settings, as well as different base-learners to represent various types of effects (see [28] for an overview; recent updates are summarized in Appendix).

2.5. Likelihood-Based Boosting. Likelihood-based boosting [3, 29] is the other general approach in the framework of statistical boosting algorithms; it received much attention particularly in the context of high-dimensional biomedical data (see [11] and the references therein). Although it follows a very similar structure to gradient boosting (see Box 1), both approaches only coincide in special cases such as classical Gaussian regression via the L_2 loss [1, 30]. In contrast to gradient boosting, the base-learners are directly estimated via optimizing the overall likelihood, using the additive predictor from the previous iteration as offset. In case of the L_2 loss, this has a similar consequence as refitting the residuals.

In every step, the algorithm hence optimizes regression models as base-learners one by one by maximizing the likelihood (using one-step Fisher scoring), selecting only the base-learner j^* which leads to the largest increase in the likelihood. In order to obtain small boosting steps, a quadratic penalty term is attached to this likelihood. This has a similar effect to multiplying the fitted base-learner by a small step length factor as in gradient boosting.

Likelihood-based boosting for generalized linear and additive regression models is provided by the R add-on package *GAMBoost* [31], and an adapted version for boosting Cox regression is provided with *CoxBoost* [32]. For a comparison of both statistical boosting approaches, that is, likelihood-based and gradient boosting in case of Cox proportional hazard models, we refer to [33].

2.6. Connections to L_1-Regularization. Statistical boosting algorithms result in regularized models with shrunk effect estimates although they only apply implicit penalization [24] by stopping the algorithm before convergence. By performing

regularization without the use of an explicit penalty term, boosting algorithms clearly differ from other direct regularization techniques like the *lasso* [34]. However, both approaches sometimes result in very similar models after being tuned to a comparable degree of regularization [10].

This close connection has been first noted between the lasso and *forward stagewise regression*, which can be viewed as special case of the gradient boosting algorithm (Box 1), and led, along with the development of *least angle regression* (LARS), to the formulation of the *positive cone condition* (PCC) [35].

If this condition holds, LARS, lasso, and forward stagewise regression coincide. Figuratively speaking, the PCC requires that all coefficient estimates monotonically increase or decrease with relaxing degree of regularization and applies, for example, to the case of low-dimensional settings with orthogonal X. It should be noted that the PCC is connected to the *diagonal dominance condition* for the inverse covariance matrix of X, which allows for a more convenient way to investigate the equivalence of these approaches in practice [36, 37].

Given that the solution of the lasso is optimal with respect to the L_1-norm of the coefficient vector, these findings led to the notion of boosting as some "sort of L_1-sparse" regularization technique [38], but it remained unclear which optimality constraints possibly apply to forward stagewise regression if the PCC is violated.

By extending $\mathbf{X}$ with a negative version of each variable and enforcing only positive updates in each iteration, Hastie et al. [39] demonstrated that forward stagewise regression always approximates the solution path of a similarly modified version of the lasso. From this perspective, they showed that forward stagewise regression minimizes the loss function subject to the L_1-*arc-length*: This means that the travelled path of the coefficients is penalized (allowing as little overall changes in the coefficients as possible, regardless of their direction), whereas the L_1-norm considers only the absolute sum of the current set of estimates.

In the same article, Hastie et al. [39] further showed that these properties hold for general convex loss functions and therefore apply not only to forward stagewise regression but also for the more general gradient boosting method (in case of logistic regression models as well as for many other generalized linear regression settings).

The consequence of these differing optimization constraints can be observed in the presence of strong collinearity, where the lasso estimates tend to be very unstable regarding different degrees of regularization while boosting approaches avoid too many changes in the coefficients as they consider the overall travelled path [10].

It has to be acknowledged, however, that direct regularization approaches as the lasso are applied more often in practice [38]. Statistical boosting, on the other hand, is far more flexible due to its modular nature allowing combining any base-learner with any type of loss function [10, 38].

3. Enhanced Variable Selection

Early stopping of statistical boosting algorithms via cross-validation approaches plays a vital role in ensuring a sparse model with optimal prediction performance on new data. Resampling, that is, random sampling of the data drawn without replacement, tends to result in sparser models compared to other sampling schemes [23], including the popular bootstrap [40]. By using base-learners of comparable complexity (in terms of degrees of freedom) selection bias can be strongly reduced [4]. The resulting models have optimal prediction accuracy on the test data. Yet, despite regularization the final models are often relatively rich [23].

3.1. Stability Selection. Meinshausen and Bühlmann [41] proposed a generic approach called stability selection to further refine the models and enhance sparsity. This approach was then transferred to boosting [12].

In general, stability selection can be combined with any variable selection approach and is particularly useful for high-dimensional data with many potential predictors. To assess how stable the selection of a variable is, B random subsets that comprise half of the data are drawn. On each of these subsets, the model is fitted until a predefined number of q base-learners are selected. Usually, $B = 100$ subsets are sufficient. Computing the relative frequencies of random subsamples in which specific base-learners were selected gives a notion of how stable the selection is with respect to perturbations of the data. Base-learners are considered to be of importance if the selection frequency exceeds a prespecified threshold level $\pi_{\text{thr}} \in [0.5, 1]$.

Meinshausen and Bühlmann [41] showed that this approach controls the per-family error rate (PFER); that is, it provides an upper bound for the expected number of false positive selections (V):

$$\mathbb{E}(V) \leq \frac{q^2}{(2\pi_{\text{thr}} - 1)\, p}, \tag{6}$$

where p is the number of base-learners. This upper bound is rather conservative and hence was further refined by Shah and Samworth [42] for specific assumptions on the distribution of the selection frequencies. Stability selection with all available error bounds is implemented for a variety of modelling techniques in the `R` package **stabs** [43].

An important issue is the choice of the hyperparameters of stability selection. The choice of a fixed value of q should be made such that it is large enough to select all hypothetically influential variables [12, 44]. A sensible value for q should usually be smaller than or equal to the number of base-learners selected via early stopping with cross-validation.

In general, the size of q is of minor importance if it is in a sensible range. With fixed q, either the threshold π_{thr} can be chosen or, as can be seen from (6) using equality, the upper bound for the PFER can be prespecified and the threshold can be derived accordingly. The latter would be the preferred choice if error control is of major importance and the former if error control is just considered a byproduct (see, e.g., [44]). For an interpretation of the PFER, particularly with regard to standard error rates such as the per-comparison error rate or the familywise error rate, we refer to Hofner et al. [12]. Note that, for fixed q, it is computationally easy to change any of the

other two parameters (π_{thr} or the upper bound for the PFER) as the resampling results can be reused [12].

The result of stability selection is not a new prediction model but a set of *stable* base-learners: In fact they might not reflect any model which can be derived with a specific penalty parameter using the original modelling approach. This means that, for boosting, no m_{stop} value might exist that results in a model with the stably selected base-learners. The provided set of stable base-learners is a fundamentally new solution and not necessarily one with a high prediction accuracy [44].

3.2. Extension and Application of Boosting with Stability Selection. Variable selection is particularly important in high-dimensional gene expression data and other large scale biomedical data sources. Recently, stability selection with boosting was successfully applied to select a small number of informative biomarkers for survival of breast cancer patients [44]. The model was derived based on a novel boosting approach that optimizes the concordance index [45, 46]. Hence, the resulting prediction rule was optimal with respect to its ability to discriminate between patients with longer and shorter survival, that is, its discriminatory power.

Thomas et al. [47] derived a modified algorithm for boosted generalized additive models for location, scale, and shape (GAMLSS [48]) to allow a combination of this very flexible model class with stability selection. The basic idea of GAMLSS is to model all parameters of the conditional distribution by their own additive predictor and associated link function. Extensive simulation studies showed that the new fitting algorithm leads to comparable models as the previous algorithm [49, 50] but is superior regarding the computational speed, especially in combination with cross-validation approaches. Furthermore, simulations showed that this algorithm can be successfully combined with stability selection to select sparser models identifying a smaller subset of truly informative variables from high-dimensional data. The algorithm is implemented in the `R` add-on package *gamboostLSS* [51].

3.3. Stability Selection for Gene Expression Data. In the following, we demonstrate the application of stability selection based on gradient boosting on three high-dimensional datasets comprising gene expression levels. This includes oligonucleotide arrays for colon cancer detection (with $n = 62$ observations and $p = 2000$ gene expression levels) [52], prediction of metastasis of breast carcinoma ($n = 168$, $p = 2905$) [53], and Riboflavin production by Bacillus subtilis ($n = 71$, $p = 4088$) [54]. All three datasets are publicly available via the `R` packages *datamicroarray* [55] and *hdi* [56].

Regarding the parameters needed to be specified for stability selection, we investigate two different error rates PFER $\in \{1, 3\}$ and a constant $q = 20$. For the sake of comparison, we additionally apply 25-fold bootstrap for variable selection, which is the default setting for cross-validation in *mboost*.

Table 1 shows the total number of variables selected by each method. It can be seen that stability selection considerably reduces the set of variables in comparison with 25-fold bootstrap. In addition, relaxing the error bound results in larger sets except for the data on breast carcinoma, where only 1 base-learner entered the stable set.

Table 1: Number of variables considered to be informative in different scenarios of stability selection and the default 25-fold bootstrap tuning of `mboost` without stability selection for comparison.

	Colon cancer	Breast carcinoma	Riboflavin production
PFER = 1, $q = 20$	2	1	4
PFER = 3, $q = 20$	3	1	5
25-fold bootstrap	11	28	39

3.4. Further Approaches for Sparse Models. In order to construct risk prediction signatures on molecular data, such as DNA methylation, Sariyar et al. [57] proposed an adaptive likelihood-based boosting algorithm. The authors included a step size modification factor c_f which represents an additional tuning parameter, adaptively controlling the size of the updates. In case of sparse settings, the approach decreases shrinkage of effect estimates (by using a larger step length) leading to a smaller bias. In settings with larger numbers of informative variables, the approach allows fitting models with lower degree of sparsity when necessary by smaller updates. The modification factor c_f has to be selected together with m_{stop} via cross-validation or resampling on a two-dimensional grid.

Zhang et al. [58] argue that variable ranking in practice is more favourable than variable selection, as ranking allows easily applying a thresholding rule in order to identify a subset of informative variables. The authors implemented a pseudo-boosting approach, which is technically not based on statistical boosting but is adapted to rank and select variables for statistical models. Note that also stability selection can be seen as a variable ranking scheme based on their selection frequency, as its selection feature is only triggered by implementing the threshold π_{thr}.

Another recent proposal is to incorporate shadow-variables *(probing)* which are permuted variants of the original predictors in the candidate model [59]. The statistical boosting algorithm is stopped, when the first shadow-variable is selected. This way the focus of the tuning procedure is effectively shifted from prediction accuracy towards selection accuracy, which could be a fast and promising procedure to ensure sparse models.

Following a gradient based approach, Huang et al. [60] adapted the sparse boosting approach by Bühlmann and Yu [61] in order to promote similarity of model sparsity structures in the integrative analysis of multiple datasets, which is an important topic regarding the trend towards big data.

4. Functional Regression

Due to technological developments, more and more data is measured continuously over time. Over the last years, a lot of methodological research focused on regression methods for this type of functional data. A groundbreaking work in this

new and evolving field of statistics is provided by Ramsay and Silverman [62].

Functional regression models can contain either functional responses (defined on a continuous domain), functional covariates, or both. This leads basically to three different classes of functional regression models, that is, function-on-scalar (response is functional), scalar-on-function (functional explanatory variable), and function-on-function regression. For recent reviews on functional regression, see Greven and Scheipl [63] and Morris [64].

4.1. Boosting Functional Data. The first statistical boosting algorithm for functional regression, allowing for data-driven variable selection, was proposed by Brockhaus et al. [65]. The authors' approach focused on linear array models [66] providing a unified framework for all three settings outlined above. Since the general structure of their gradient boosting algorithm is similar to the one in Box 1, the resulting models still have the same form as in (2), only that the response Y and the covariates may be functions. The underlying functional partial effects $h_j(\mathbf{x})$ can be represented using tensor product basis

$$h_j(\mathbf{x})(t) = \left(b_j(\mathbf{x})^\top \otimes b_Y(t)^\top\right)\theta_j, \tag{7}$$

where θ_j is the vector of coefficients, b_j and b_Y are basis functions, and $\otimes$ denotes the Kronecker product.

This functional array model is limited in two ways: (i) the functional responses need to be measured on a common grid and (ii) covariates need to be constant over the domain of the response. As particularly the second assumption might often not be fulfilled in practice, Brockhaus et al. [14] soon thereafter proposed a general framework for boosting functional regression models avoiding this assumption and dropping the linear array structure.

This newer framework [14] comprises also all three model classes outlined above and particularly focuses on historical effects, where functional response and functional covariates are observed over the same time interval. The underlying assumption is that observations of the covariate affect the response only up to the corresponding time point t

$$\mathbb{E}(Y(t) \mid X = x) = \sum_{j=1}^{J} \int_{t_1}^{t} x_j(s)\,\beta_j(s,t)\,ds, \tag{8}$$

where s represents the time points the covariate was observed for. In other words, only the part of the covariate function lying in the past (not the future) can affect the present response. However, this is a sensible restriction in most practical applications.

Both approaches for boosting functional regression are implemented in the `R` add-on package *FDboost* [67], which relies on the fitting methods and infrastructure of *mboost*.

4.2. Extensions of Boosting Functional Regression. Boosting functional data can be combined with stability selection (see Section 3.1) to enhance the variable selection properties of the algorithm [14, 65].

The boosting approach for functional data has already been extended towards the model class of generalized additive models for location, scale, and shape (GAMLSS) for a scalar-on-function setting by Brockhaus et al. [68]. The functional approach was named signal regression models for location, scale, and shape [68]. The estimation via gradient boosting is based on the corresponding gamboostLSS algorithm for boosting GAMLSS [49, 50].

In an approach to analyse the functional relationship between bioelectrical signals like electroencephalography (EEG) and facial electromyography (EMG), Rügamer et al. [69] focused on extending the framework of boosting functional regression by incorporating factor-specific historical effects, similar to (8).

Although functional data analysis triggered a lot of methodological research, a recent systematic review by Ullah and Finch [70] revealed that the number of actual biomedical applications of functional data analysis in general and functional regression in particular is rather small. The authors argued that the potential benefits of these flexible models (like richer interpretation and more flexible structures) are not yet well understood by practitioners and that further efforts are necessary to promote the actual usage of these novel techniques.

5. Boosting Advanced Survival Models

Cox regression is still the dominant model class for boosting time-to-event data; see [33] for a comparison of two different boosting algorithms and [71] for different general approaches to estimate Cox models in the presence of high-dimensional data. However, over the last years several alternatives emerged [45, 46, 72]. In this section we will particularly focus on boosting joint models of time-to-event outcomes and longitudinal markers but will also briefly refer to other recent extensions.

5.1. Boosting Joint Models. The concept of joint modelling of longitudinal and time-to-event data [73] has found its way into the statistical literature in the last few years as it thoroughly addresses questions on continuous data recorded over time and event times related to this continuous data. Modelling those two processes independently leads to misspecified models prone to bias. There are various joint modelling approaches and thus also various different model equations based on different covariates, distributions, and covariance structures. The type we are going to refer to in this review is the following:

$$\begin{aligned} y_{ij} &= \eta_{\mathrm{l}}\left(x_{ij}\right) + \eta_{\mathrm{ls}}\left(x_i, t_{ij}\right) + \varepsilon_{ij} \\ &\lambda\left(t \mid \alpha, \eta_{\mathrm{s}}\left(x_i, t\right), \eta_{\mathrm{ls}}\left(x_i, t\right)\right) \\ &\quad = \lambda_0(t)\exp\left(\eta_{\mathrm{s}}\left(x_i, t\right) + \alpha\eta_{\mathrm{ls}}\left(x_i, t\right)\right), \end{aligned} \tag{9}$$

where y_{ij} is the jth observation of the ith individual with $i = 1, \ldots, n$ and $j = 1, \ldots, n_i$ and $\lambda(t \mid \alpha, \eta_{\mathrm{s}}(x_i, t), \eta_{\mathrm{ls}}(x_i, t))$ is the hazard function for individual i at time point t. Both outcomes, the longitudinal measurement y_i and the time t_i, recorded alongside with the censoring indicator δ_i, are modelled based on two subpredictors each: one that is supposed

to have an impact on only one of them (the longitudinal subpredictor $\eta_{\text{l}}(x_{ij})$ and the survival subpredictor $\eta_{\text{s}}(x_{ij}, t)$) and the other being shared by both parts of the model (the shared subpredictor $\eta_{\text{ls}}(x_{ij}, t)$). All those subpredictors are functions of different, possibly time-dependent variables x_i. The type of model presented here does not include fixed time varying covariates for the survival part of the model; please note that those models do exist but are not implemented in the boosting framework yet. It however includes time itself and, just like most joint models, some type of random effects. The function $\lambda_0(t)$ is the baseline hazard. Most approaches for joint models are based on likelihood or Bayesian inference using the joint likelihood resulting as a product from the corresponding likelihoods of the above processes [74, 75]. Those approaches are, however, unable to conduct variable selection and cannot deal with high-dimensional data.

Waldmann et al. [13] suggested a boosting algorithm tackling these challenges. The model used in that paper is a reduced version of (9) in which no survival subpredictor is considered and a fixed baseline hazard λ_0 is used. The algorithm is a version of the classical boosting algorithm as represented in Box 1, which is adapted to the special case of having to estimate a set of different subpredictors (similar to the GAMLSS framework [49]). The algorithm is therefore composed of three steps which are performed circularly. In the first step a regular boosting step to update the longitudinal subpredictor $\eta_{\text{l}}(x_{ij})$ is performed and the parameters of the shared subpredictor are treated as fixed. In the second step, the parameters of the longitudinal subpredictor are fixed and a boosting step for the shared subpredictor $\eta_{\text{ls}}(x_{ij})$ is conducted. The third step is a simple optimization step: based on the current values of the parameters in both subpredictors the likelihoods are optimized with respect to λ_0, σ^2, and α (cf. [76]). The number of iterations now depends on two stopping iterations which have to be optimized on a two-dimensional grid via cross-validation.

Waldmann et al. [13] showed that the benefits of boosting algorithm (automated variable selection and handling of $p > n$ situations) can be transferred to joint modelling and hence lay the groundwork to further extended joint modelling approaches.

5.2. An Example of Boosting Joint Models. The example presented in the following is similar to the simulation study in [13]. The simulated data consists of $N = 500$ individuals and a maximum of $n_i = 5$ observations per individual. Some observations are however truncated due to the risk function induced by the survival part of the model. The actual number of observations hence was 2350. The longitudinal subpredictor contains two informative variables and the intercept ($\beta_{\text{l}(0,1,2)} = (2, 1, -2)$) as well as 1250 noninformative variables. The shared subpredictor has two fixed time invariant variables ($\beta_{\text{ls}(1,2)} = (1, -2)$), a time effect ($\beta_t = 1$), random intercept and slope, and also 1250 noninformative variables. In total there are hence 2508 covariates for 2350 observations, a situation clearly infeasible for ordinary joint modelling approaches.

We ran the above presented algorithm on this simulated example. By tenfold cross-validation we found the optimal stopping iterations to be $m_{\text{stop,l}} = 125$ and $m_{\text{stop,ls}} = 130$. The algorithm was able to detect the informative variables and the resulting coefficients were close to the original values $\widehat{\beta}_{\text{l}(0,1,2)} = (2.042, 0.993, -1.999)$, $\beta_{\text{ls}(1,2,t)} = (0.971, -1.980, 0.876)$. The longitudinal subpredictor furthermore selected three and the shared subpredictor two noninformative variables; hence only 0.2% of the noninformative variables were selected, all of which had absolute values below 0.023. Those results are typical findings for simulations done with the package based on the code for the approach presented here. It is available in the R add-on package *JMboost* [77], currently on GitHub.

5.3. Other New Approaches on Boosting Survival Data. Reulen and Kneib [78] extended the framework of statistical boosting towards multistate models for patients exposed to competing risks (e.g., adverse events, recovery, death, or relapse). The approach is implemented in the *gamboostMSM* package [79], relying on the infrastructure of *mboost*. Möst and Hothorn [80] focused on boosting the patient-specific survivor function based on conditional transformation models [81] incorporating inverse probability of censoring weights [82].

When statistical boosting algorithms are used to estimate survival models, the motivation most often is the presence of high-dimensional data. De Bin et al. [83] investigated several approaches (including gradient boosting and likelihood-based boosting) to incorporate both clinical and high-dimensional omics data in prediction models.

Guo et al. [84] proposed a new adaptive likelihood-based boosting algorithm to fit Cox models, incorporating a direct lasso-type L_1 penalization in the fitting process in order to avoid the inclusion of variables with small effect. The general motivation is similar to the step length modification factor proposed by Sariyar et al. [57]. In another approach, Sariyar et al. [85] combined a likelihood-based boosting approach for the Cox model with random forests in order to screen for interaction effects in high-dimensional data. Hieke et al. [86] combined likelihood-based boosting with resampling to identify prognostic SNPs in potentially small clinical cohorts.

6. New Frontiers and Applications

Also other new topics have been incorporated into the framework of statistical boosting, but not all of them can be presented in detail here. However, we want to give a short overview of the most relevant developments, many of which were actually motivated by biomedical applications.

Weinhold et al. [87] proposed to analyse DNA methylation data (signal intensities M and U), via a "ratio of correlated gammas" model. Based on a bivariate gamma distribution for M and U values, the authors derived the density for the ratio $M/(M+U)$ and optimized it via gradient boosting.

A boosting algorithm for differential item functioning in Rasch models was developed by Schauberger and Tutz [88] for the broader area of psychometrics, while Casalicchio et al. focused on boosting subject-specific Bradley-Terry-Luce models [89].

Napolitano et al. [90] developed a sampled boosting algorithm for the analysis of brain perfusion images: Gradient boosting is carried out multiple times on different training sets. Each base-learner refers to a voxel and after every sampling iteration a fixed fraction of selected voxels is randomly left out from the following boosting fit, to force the algorithm to select new voxels. The final model is then computed as the global sum of all solutions. Feilke et al. [91] proposed a voxelwise boosting approach for the analysis of contrast-enhanced magnetic resonance imaging data (DCE-MRI), which was additionally enhanced by a spatial penalty to account for the regional structure of the voxels.

Pybus et al. [92] proposed a hierarchical boosting algorithm for classification in an approach to detect positive selection in genomic regions (cf. [93]). Truntzer et al. [94] compared the classification performance of gradient boosting with other methods combining clinical variables and high-dimensional mass spectrometry data and concluded that the variable selection properties of boosting also led to a very good performance regarding prediction accuracy.

Regarding boosting location and scale models (modelling both expected value and variance in the spirit of GAMLSS [48]), Messner et al. [95] proposed a boosting algorithm for predictor selection in ensemble postprocessing to better calibrate ensemble weather forecasts. The idea of ensemble forecasting is to account for model errors and to quantify forecast uncertainty. Mayr et al. [96] used boosted location and scale models in combination with permutation tests to assess simultaneously systematic bias and random measurement errors of medical devices. The use of a permutation test tackles one of the remaining problems of statistical boosting approaches in practical biomedical research: The lack of standard errors for effect estimates makes it necessary to incorporate resampling procedures to construct confidence intervals or to assess significance of effects.

The methodological development in [96] was motivated by the analysis of biomedical data. Statistical boosting algorithms, however, have been applied over the last few years in various biomedical applications without the need for methodological extensions. Most applications focus on prediction modelling or variable selection.

To give an idea of the variety of topics, we briefly mention a selection of the most recent ones from the last two years. These applications comprise the development of birth weight prediction formulas for particularly small babies [97], prediction of smoking cessation and its relapse in HIV-infected patients [98], *Escherichia coli* Fed-Batch Fermentation Modelling [99], prediction of cardiovascular death for older patients in the emergency department [100], and identification of factors influencing therapeutic decisions regarding rheumatoid arthritis [101].

7. Discussion

In this article, we have highlighted several new research areas in the field of statistical boosting leaving the traditional GAM modelling approach. A particularly active research area during the last few years addresses the development of boosting algorithms for new model classes extending the GAM framework. These include, among others, the simultaneous modelling of location, scale, and shape parameters within the GAMLSS framework [49], the modelling of functional data [65], and, recently, the class of joint models for longitudinal and survival data [13]. It goes without saying that these developments will make boosting algorithms available for practical use in much more sophisticated clinical and epidemiological applications.

Another line of research aims at exploring the connections between statistical boosting methods and machine learning techniques that were originally developed independently of boosting. An important example is stability selection, a generic methodology that, at the time of its development, mainly focused on penalized regression models such as the lasso. Only recently has stability selection been adapted to become a tool for variable selection within the boosting framework (e.g., [47]). Other work in this context is the analysis of the connections between boosting and penalized regression [10] and the work by Sariyar et al. [85] exploring a combination of boosting and random forest methods.

Finally, as already noted by Hothorn [24], boosting may be regarded not only as a framework for regularized model fitting but also as a generic optimization tool in its own right. In particular, boosting constitutes a robust algorithm for the optimization of objective functions that, due to their structure or complexity, may pose problems for Newson-Raphson-type and related methods. This motivated the use of boosting in the articles by Hothorn et al. [81] and Weinhold et al. [87].

Regarding future research, a huge challenge for the use of boosting algorithms in biomedical applications arises from the *era of big data*. Unlike other machine learning methods like random forests, the sequential nature of boosting methods hampers the use of parallelization techniques within the algorithm, which may result in issues with the fitting and tuning of complex models with multidimensional predictors and/or sophisticated base-learners like splines or higher-sized trees. To overcome these problems in classification and univariate regression, Chen and Guestrin [102] developed the extremely fast and sophisticated *xgboost* environment.

For the more recent extensions discussed in this paper, however, *big data* solutions for statistical boosting have yet to be developed.

Appendix

Developments regarding the mboost Package

This appendix describes important changes during the last years that were implemented in the R package *mboost* after the tutorial paper [28] on its use was published.

Starting from *mboost 2.2*, the default for the degrees of freedom was changed; they are now defined as

$$\mathrm{df}(\lambda) = \mathrm{trace}\left(2S - S^{\top}S\right), \tag{A.1}$$

with smoother matrix $S = X(X^{\top}X + \lambda K)^{-1}X$. Analyses have shown that this leads to a reduced selection bias; see [4]. Earlier versions used the trace of

the smoother matrix as degrees of freedom; that is, $df(\lambda) = \text{trace}(S)$. One can change to the old definition by setting `options(mboost_dftraceS = TRUE)`. For parallel computations of cross-validated stopping values, *mboost* now uses the package *parallel*, which is included in the standard `R` installation. The behavior of `bols(x, intercept = FALSE)` was changed when `x` is a factor: the intercept is simply dropped from the design matrix and the coding can be specified as usual for factors. Additionally, a new contrast was introduced: `"contr.dummy"` (see the manual of `bols` for details). Finally, the computation of B-spline basis at the boundaries was changed such that equidistant boundary knots are used per default.

With *mboost 2.3*, constrained effects [103, 104] are fitted per default using quadratic programming methods (option `type = "quad.prog"`) improving the speed of computation drastically. In addition to monotonic, convex, and concave effects, new constraints were introduced to fit `"positive"` or `"negative"` effects or effects with boundary constraints (see `bmono` for details). Additionally, a new function to assign m_{stop} values to a model object was added (`mstop(mod) <- i`) as well as two new distribution families `Hurdle` [105] and `Multinomial` [76]. Finally, a new option was implemented to allow for stopping based on out-of-bag data during fitting (via `boost_control(..., stopintern = TRUE)`).

With *mboost 2.4*, bootstrap confidence intervals were implemented in the novel `confint` function [104]. The stability selection procedure was moved to a dedicated package **stabs** [43], while a specific function for gradient boosting was implemented in package *mboost*.

From *mboost 2.5* onward, cross-validation does not stop on errors in single folds anymore and was sped up by setting `mc.preschedule = FALSE` if parallel computations via `mclapply` are used. A documentation for the function `plot.mboost` was added, which allows visualizing model results. Values outside the boundary knots are now forbidden during fitting, while linear extrapolation is used for prediction.

With *mboost 2.6* a lot of bug fixes and small improvements were provided. Most notably, the development of the package is now hosted entirely on github in the collaborative project boost-R/mboost and the package maintainer changed.

The *mboost 2.7* version provides a new family `Cindex` [45], variable importance measures (`varimp`), and improved plotting facilities.

The current CRAN version *mboost 2.8* includes major changes to the `Binomial` family which now additionally provides an alternative implementation of Binomial regression models along the lines of the classic `glm` implementation, which can be used via `Binomial(type = "glm")`. This family also works with a two-column matrix containing the number of successes and number of failures. Furthermore, models with zero steps (i.e., models containing only the offset) are supported and cross-validation can now select models without base-learners. Finally, a new base-learner `bkernel` for pathway-based kernel boosting in genome-wide association studies (GWAS) was added [106].

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors thank Corinna Buchstaller for her help with the literature search. The first and the last author's work on this article was supported by the Deutsche Forschungsgemeinschaft (DFG) (`http://www.dfg.de`), Grant no. SCHM 2966/1-2. Support of the Interdisciplinary Center for Clinical Research (IZKF) of the Friedrich-Alexander-Universität Erlangen-Nürnberg via the Projects J49 (grant to Andreas Mayr) and J61 (grant to Elisabeth Waldmann) is also gratefully acknowledged.

References

[1] A. Mayr, H. Binder, O. Gefeller, and M. Schmid, "The evolution of boosting algorithms: from machine learning to statistical modelling," *Methods of Information in Medicine*, vol. 53, no. 6, pp. 419–427, 2014.

[2] P. Bühlmann and T. Hothorn, "Rejoinder: boosting algorithms: regularization, prediction and model fitting," *Statistical Science*, vol. 22, no. 4, pp. 516–522, 2007.

[3] G. Tutz and H. Binder, "Generalized additive modeling with implicit variable selection by likelihood-based boosting," *Biometrics*, vol. 62, no. 4, pp. 961–971, 2006.

[4] B. Hofner, T. Hothorn, T. Kneib, and M. Schmid, "A framework for unbiased model selection based on boosting," *Journal of Computational and Graphical Statistics*, vol. 20, no. 4, pp. 956–971, 2011.

[5] T. Kneib, T. Hothorn, and G. Tutz, "Variable selection and model choice in geoadditive regression models," *Biometrics*, vol. 65, no. 2, pp. 626–634, 2009.

[6] L. Breiman, "Statistical modeling: the two cultures," *Statistical Science*, vol. 16, no. 3, pp. 199–231, 2001.

[7] Y. Freund, "Boosting a weak learning algorithm by majority," in *Proceedings of the Third Annual Workshop on Computational Learning Theory, COLT 1990*, M. A. Fulk and J. Case, Eds., pp. 202–216, University of Rochester, Rochester, NY, USA, August 1990.

[8] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[9] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The Annals of Statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[10] T. Hepp, M. Schmid, O. Gefeller, E. Waldmann, and A. Mayr, "Approaches to regularized regression - A comparison between gradient boosting and the lasso," *Methods of Information in Medicine*, vol. 55, no. 5, pp. 422–430, 2016.

[11] A. Mayr, H. Binder, O. Gefeller, and M. Schmid, "Extending statistical boosting," *Methods of Information in Medicine*, vol. 53, no. 6, pp. 428–435, 2014.

[12] B. Hofner, L. Boccuto, and M. Göker, "Controlling false discoveries in high-dimensional situations: Boosting with stability selection," *BMC Bioinformatics*, vol. 16, no. 1, article no. 144, 2015.

[13] E. Waldmann, D. Taylor-Robinson, N. Klein et al., "Boosting joint models for longitudinal and time-to-event data," *Biometrical Journal*, 2017.

[14] S. Brockhaus, M. Melcher, F. Leisch, and S. Greven, "Boosting flexible functional regression models with a high number of functional historical effects," *Statistics and Computing*, vol. 27, no. 4, pp. 913–926, 2017.

[15] R. E. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, pp. 197–227, 1990.

[16] R. Schapire and Y. Freund, *Boosting: Foundations and Algorithms*, vol. 14, MIT Press, 2012.

[17] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the Thirteenth International Conference on Machine Learning Theory*, pp. 148–156, San Francisco: Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1996.

[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, New York, NY, USA, 2nd edition, 2009.

[19] A. J. Wyner, M. Olson, J. Bleich, and D. Mease, "Explaining the success of adaboost and random forests as interpolating classifiers," *Journal of Machine Learning Research*, vol. 18, no. 48, pp. 1–33, 2017.

[20] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: illustrations, sources and a solution," *BMC Bioinformatics*, vol. 8, article 25, 2007.

[21] A. Hapfelmeier, T. Hothorn, K. Ulm, and C. Strobl, "A new variable importance measure for random forests with missing data," *Statistics and Computing*, vol. 24, no. 1, pp. 21–34, 2014.

[22] T. J. Hastie and R. J. Tibshirani, *Generalized Additive Models*, vol. 43, Chapman and Hall, London, UK, 1990.

[23] A. Mayr, B. Hofner, and M. Schmid, "The importance of knowing when to stop: a sequential stopping rule for component-wise gradient boosting," *Methods of Information in Medicine*, vol. 51, no. 2, pp. 178–186, 2012.

[24] T. Hothorn, "Boosting – An unusual yet attractive optimiser," *Methods of Information in Medicine*, vol. 53, no. 6, pp. 417-418, 2014.

[25] L. Mason, J. Baxter, P. Bartlett, and M. Frean, "Boosting algorithms as gradient descent," in *Proceedings of the 13th Annual Neural Information Processing Systems Conference, NIPS 1999*, pp. 512–518, usa, December 1999.

[26] T. Hothorn, P. Bühlmann, T. Kneib, M. Schmid, and B. Hofner, "mboost: Model-Based Boosting," 2017, R package version 2.8-0. https://CRAN.R-project.org/package=mboost.

[27] R Development Core Team, *R: A Language and Environment for Statistical Computing*, Vienna, Austria, 2016, ISBN 3-900051-07-0. https://www.R-project.org.

[28] B. Hofner, A. Mayr, N. Robinzonov, and M. Schmid, "Model-based boosting in R: a hands-on tutorial using the R Package mboost," *Computational Statistics*, vol. 29, no. 1-2, pp. 3–35, 2014.

[29] G. Tutz and H. Binder, "Boosting ridge regression," *Computational Statistics and Data Analysis*, vol. 51, no. 12, pp. 6044–6059, 2007.

[30] P. Bühlmann and B. Yu, "Boosting with the L2 loss: regression and classification," *Journal of the American Statistical Association*, vol. 98, pp. 324–338, 2003.

[31] H. Binder, *GAMBoost: Generalized Linear and Additive Models by Likelihood Based Boosting*, 2011, R package version 1.2-2. https://CRAN.R-project.org/package=GAMBoost.

[32] H. Binder, *CoxBoost: Cox Models by Likelihood-based Boosting for a Single Survival Endpoint or Competing Risks*, 2013, R package version 1.4. https://CRAN.R-project.org/package=CoxBoost.

[33] R. De Bin, "Boosting in Cox regression: a comparison between the likelihood-based and the model-based approaches with focus on the R-packages CoxBoost and mboost," *Computational Statistics*, vol. 31, no. 2, pp. 513–531, 2016.

[34] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society - Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[35] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[36] N. Meinshausen, G. Rocha, and B. Yu, "Discussion: a tale of three cousins: Lasso, L2 boosting and Dantzig," *The Annals of Statistics*, vol. 35, no. 6, pp. 2373–2384, 2007.

[37] J. Duan, C. Soussen, D. Brie, J. Idier, and Y.-P. Wang, "On LARS/homotopy equivalence conditions for over-determined LASSO," *IEEE Signal Processing Letters*, vol. 19, no. 12, 2012.

[38] P. Bühlmann, J. Gertheiss, S. Hieke et al., "Discussion of 'the evolution of boosting algorithms' and 'extending statistical boosting'," *Methods of Information in Medicine*, vol. 53, no. 6, pp. 436–445, 2014.

[39] T. Hastie, J. Taylor, R. Tibshirani, and G. Walther, "Forward stagewise regression and the monotone lasso," *Electronic Journal of Statistics*, vol. 1, pp. 1–29, 2007.

[40] S. Janitza, H. Binder, and A.-L. Boulesteix, "Pitfalls of hypothesis tests and model selection on bootstrap samples: causes and consequences in biometrical applications," *Biometrical Journal*, vol. 58, no. 3, pp. 447–473, 2016.

[41] N. Meinshausen and P. Bühlmann, "Stability selection," *Journal of the Royal Statistical Society Series B*, vol. 72, no. 4, pp. 417–473, 2010.

[42] R. D. Shah and R. J. Samworth, "Variable selection with error control: another look at stability selection," *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, vol. 75, no. 1, pp. 55–80, 2013.

[43] B. Hofner, T. Hothorn, and stabs., *Stability Selection with Error Control*, 2017, R package version 0.6-2, https://CRAN.R-project.org/package=stabs.

[44] A. Mayr, B. Hofner, and M. Schmid, "Boosting the discriminatory power of sparse survival models via optimization of the concordance index and stability selection," *BMC Bioinformatics*, vol. 17, no. 1, article no. 288, 2016.

[45] A. Mayr and M. Schmid, "Boosting the concordance index for survival data - a unified framework to derive and evaluate biomarker combinations," *PLoS ONE*, vol. 9, no. 1, Article ID e84483, 2014.

[46] Y. Chen, Z. Jia, D. Mercola, and X. Xie, "A gradient boosting algorithm for survival analysis via direct optimization of concordance index," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 873595, 2013.

[47] J. Thomas, A. Mayr, B. Bischl, M. Schmid, A. Smith, and B. Hofner, "Gradient boosting for distributional regression: faster tuning and improved variable selection via noncyclical updates," *Statistics and Computing*, pp. 1–15, 2017.

[48] R. A. Rigby and D. M. Stasinopoulos, "Generalized additive models for location, scale and shape," *Journal of the Royal Statistical Society. Series C. Applied Statistics*, vol. 54, no. 3, pp. 507–554, 2005.

[49] A. Mayr, N. Fenske, B. Hofner, T. Kneib, and M. Schmid, "Generalized additive models for location, scale and shape for high dimensional data—a flexible approach based on boosting," *Journal of the Royal Statistical Society. Series C. Applied Statistics*, vol. 61, no. 3, pp. 403–427, 2012.

[50] B. Hofner, A. Mayr, and M. Schmid, "gamboostLSS: an R package for model building and variable selection in the GAMLSS framework," *Journal of Statistical Software*, vol. 74, no. 1, 2016.

[51] B. Hofner, A. Mayr, N. Fenske, J. Thomas, and M. Schmid, "gamboostLSS: Boosting Methods for GAMLSS Models," 2017, R package version 2.0-0, https://CRAN.R-project.org/package=gamboostLSS.

[52] U. Alon, N. Barka, D. A. Notterman et al., "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 96, no. 12, pp. 6745–6750, 1999.

[53] E. Gravier, G. Pierron, A. Vincent-Salomon et al., "A prognostic DNA signature for T1T2 node-negative breast cancer patients," *Genes Chromosomes and Cancer*, vol. 49, no. 12, pp. 1125–1134, September 2009.

[54] P. Bühlmann, M. Kalisch, and L. Meier, "High-dimensional statistics with a view toward applications in Biology," *Annual Review of Statistics and Its Application*, vol. 1, pp. 255–278, 2014.

[55] J. A. Ramey, "Datamicroarray: Collection of Data Sets for Classification," https://github.com/ramhiser/datamicroarray, 2016.

[56] R. Dezeure, P. Bühlmann, L. Meier, and N. Meinshausen, "High-dimensional inference: confidence intervals, p-values and R-software hdi," *Statistical Science*, vol. 30, no. 4, pp. 533–558, 2015.

[57] M. Sariyar, M. Schumacher, and H. Binder, "A boosting approach for adapting the sparsity of risk prediction signatures based on different molecular levels," *Statistical Applications in Genetics and Molecular Biology*, vol. 13, no. 3, pp. 343–357, 2014.

[58] C.-X. Zhang, J.-S. Zhang, and S.-W. Kim, "PBoostGA: pseudo-boosting genetic algorithm for variable ranking and selection," *Computational Statistics*, vol. 31, no. 4, pp. 1237–1262, 2016.

[59] J. Thomas, T. Hepp, A. Mayr, and B. Bischl, "Probing for sparse and fast variable selection with model-based boosting".

[60] Y. Huang, J. Liu, H. Yi, B.-C. Shia, and S. Ma, "Promoting similarity of model sparsity structure in integrative analysis of cancer genetic data," *Statistics in Medicine*, vol. 36, no. 3, pp. 509–559, 2017.

[61] P. Bühlmann and B. Yu, "Sparse boosting," *Journal of Machine Learning Research*, vol. 7, pp. 1001–1024, 2006.

[62] J. O. Ramsay and B. W. Silverman, *Applied functional data analysis: methods and case studies*, vol. 77 of *Springer Series in Statistics*, Springer, Berlin, Germany, 2002.

[63] S. Greven and F. Scheipl, "A general framework for functional regression modelling," *Statistical Modelling*, vol. 17, no. 1-2, pp. 1–35, 2017.

[64] J. S. Morris, "Functional regression," *Annual Review of Statistics and Its Application*, vol. 2, pp. 321–359, 2015.

[65] S. Brockhaus, F. Scheipl, T. Hothorn, and S. Greven, "The functional linear array model," *Statistical Modelling*, vol. 15, no. 3, pp. 279–300, 2015.

[66] I. D. Currie, M. Durban, and P. H. Eilers, "Generalized linear array models with applications to multidimensional smoothing," *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, vol. 68, no. 2, pp. 259–280, 2006.

[67] S. Brockhaus and D. Rügamer, *Brockhaus S, Rügamer D. FDboost: Boosting Functional Regression Models; R package version 0.2-0*, 2016, https://CRAN.R-project.org/package=FDboost, 2016.

[68] S. Brockhaus, A. Fuest, A. Mayr, and S. Greven, "Signal regression models for location, scale and shape with an application to stock returns".

[69] D. Rügamer, S. Brockhaus, K. Gentsch, K. Scherer, and S. Greven, "Boosting factor-specific functional historical models for the detection of synchronisation in bioelectrical signals".

[70] S. Ullah and C. F. Finch, "Applications of functional data analysis: a systematic review," *BMC Medical Research Methodology*, vol. 13, no. 1, article 43, 2013.

[71] C. Zemmour, F. Bertucci, P. Finetti et al., "Prediction of early breast cancer metastasis from dna microarray data using high-dimensional Cox regression models," *Cancer Informatics*, vol. 14, supplement 2, pp. 129–138, 2015.

[72] M. Schmid and T. Hothorn, "Flexible boosting of accelerated failure time models," *BMC Bioinformatics*, vol. 9, article 269, 2008.

[73] M. S. Wulfsohn and A. A. Tsiatis, "A joint model for survival and longitudinal data measured with error," *Biometrics*, vol. 53, no. 1, pp. 330–339, 1997.

[74] C. L. Faucett and D. C. Thomas, "Simultaneously modelling censored survival data and repeatedly measured covariates: a Gibbs sampling approach," *Statistics in Medicine*, vol. 15, no. 15, pp. 1663–1685, 1996.

[75] D. Rizopoulos, "JM: an R package for the joint modelling of longitudinal and time-to-event data," *Journal of Statistical Software*, vol. 35, no. 9, pp. 1–33, 2010.

[76] M. Schmid, S. Potapov, A. Pfahlberg, and T. Hothorn, "Estimation and regularization techniques for regression models with multidimensional prediction functions," *Statistics and Computing*, vol. 20, no. 2, pp. 139–150, 2010.

[77] E. Waldmann and A. Mayr, "JMboost: Boosting Joint Models for Longitudinal and Time-to-Event Outcomes," R package version 0.1-0. https://github.com/mayrandy/JMboost.

[78] H. Reulen and T. Kneib, "Boosting multi-state models," *Lifetime Data Analysis*, vol. 22, no. 2, pp. 241–262, 2016.

[79] H. Reulen, "gamboostMSM: Estimating multistate models using gamboost()," 2014. R package version 1.1.87. https://CRAN.R-project.org/package=gamboostMSM.

[80] L. Möst and T. Hothorn, "Conditional transformation models for survivor function estimation," *The International Journal of Biostatistics*, vol. 11, no. 1, pp. 23–50, 2015.

[81] T. Hothorn, T. Kneib, and P. Bühlmann, "Conditional transformation models," *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, vol. 76, no. 1, pp. 3–27, 2014.

[82] M. J. van der Laan and J. M. Robins, *Unified Methods for Censored Longitudinal Data and Causality*, New York, NY, USA, Springer Science & Business Media, 2003.

[83] R. De Bin, W. Sauerbrei, and A.-L. Boulesteix, "Investigating the prediction ability of survival models based on both clinical and omics data: two case studies," *Statistics in Medicine*, vol. 33, no. 30, pp. 5310–5329, 2014.

[84] Z. Guo, W. Lu, and L. Li, "Forward Stagewise Shrinkage and Addition for High Dimensional Censored Regression," *Statistics in Biosciences*, vol. 7, no. 2, pp. 225–244, 2015.

[85] M. Sariyar, I. Hoffmann, and H. Binder, "Combining techniques for screening and evaluating interaction terms on high-dimensional time-to-event data," *BMC Bioinformatics*, vol. 15, no. 1, article 58, 2014.

[86] S. Hieke, A. Benner, R. F. Schlenk, M. Schumacher, L. Bullinger, and H. Binder, "Identifying prognostic SNPs in clinical cohorts: complementing univariate analyses by resampling and multivariable modeling," *PLoS ONE*, vol. 11, no. 5, Article ID e0155226, 2016.

[87] L. Weinhold, S. Wahl, S. Pechlivanis, P. Hoffmann, and M. Schmid, "A statistical model for the analysis of beta values in DNA methylation studies," *BMC Bioinformatics*, vol. 17, no. 1, article 480, 2016.

[88] G. Schauberger and G. Tutz, "Detection of differential item functioning in Rasch models by boosting techniques," *British Journal of Mathematical and Statistical Psychology*, vol. 69, no. 1, pp. 80–103, 2016.

[89] G. Casalicchio, G. Tutz, and G. Schauberger, "Subject-specific Bradley-Terry-Luce models with implicit variable selection," *Statistical Modelling*, vol. 15, no. 6, pp. 526–547, 2015.

[90] G. Napolitano, J. C. Stingl, M. Schmid, and R. Viviani, "Predicting CYP2D6 phenotype from resting brain perfusion images by gradient boosting," *Psychiatry Research: Neuroimaging*, vol. 259, pp. 16–24, 2017.

[91] M. Feilke, B. Bischl, V. J. Schmid, and J. Gertheiss, "Boosting in nonlinear regression models with an application to DCE-MRI data," *Methods of Information in Medicine*, vol. 55, no. 1, pp. 31–41, 2016.

[92] M. Pybus, P. Luisi, G. M. Dall'Olio et al., "Hierarchical boosting: a machine-learning framework to detect and classify hard selective sweeps in human populations," *Bioinformatics*, vol. 31, no. 24, pp. 3946–3952, 2015.

[93] K. Lin, H. Li, C. Schlötterer, and A. Futschik, "Distinguishing positive selection from neutral evolution: Boosting the performance of summary statistics," *Genetics*, vol. 187, no. 1, pp. 229–244, 2011.

[94] C. Truntzer, E. Mostacci, A. Jeannin, J.-M. Petit, P. Ducoroy, and H. Cardot, "Comparison of classification methods that combine clinical data and high-dimensional mass spectrometry data," *BMC Bioinformatics*, vol. 15, no. 1, article 385, 2014.

[95] J. W. Messner, G. J. Mayr, and A. Zeileis, "Nonhomogeneous Boosting for Predictor Selection in Ensemble Postprocessing," *Monthly Weather Review*, vol. 145, no. 1, pp. 137–147, 2017.

[96] A. Mayr, M. Schmid, A. Pfahlberg, W. Uter, and O. Gefeller, "A permutation test to analyse systematic bias and random measurement errors of medical devices via boosting location and scale models," *Statistical Methods in Medical Research*, vol. 26, no. 3, pp. 1443–1460, 2017.

[97] F. Faschingbauer, U. Dammer, E. Raabe et al., "A new sonographic weight estimation formula for small-for-gestational-age fetuses," *Journal of Ultrasound in Medicine*, vol. 35, no. 8, pp. 1713–1724, 2016.

[98] J. Schäfer, J. Young, E. Bernasconi et al., "Predicting smoking cessation and its relapse in HIV-infected patients: the swiss HIV cohort study," *HIV Medicine*, vol. 16, no. 1, pp. 3–14, 2015.

[99] M. Melcher, T. Scharl, M. Luchner, G. Striedner, and F. Leisch, "Boosted structured additive regression for," *Biotechnology and Bioengineering*, vol. 114, no. 2, pp. 321–334, 2017.

[100] P. Bahrmann, M. Christ, B. Hofner et al., "Prognostic value of different biomarkers for cardiovascular death in unselected older patients in the emergency department," *European Heart Journal: Acute Cardiovascular Care*, vol. 5, no. 8, pp. 568–578, 2016.

[101] D. Pattloch, A. Richter, B. Manger et al., "Das erste Biologikum bei rheumatoider arthritis: einflussfaktoren auf die Therapieentscheidung," *Zeitschrift für Rheumatologie*, vol. 76, no. 3, pp. 210–218, 2017.

[102] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016*, pp. 785–794, August 2016.

[103] B. Hofner, J. Müller, and T. Hothorn, "Monotonicity-constrained species distribution models," *Ecology*, vol. 92, no. 10, pp. 1895–1901, 2011.

[104] B. Hofner, T. Kneib, and T. Hothorn, "A unified framework of constrained regression," *Statistics and Computing*, vol. 26, no. 1-2, pp. 1–14, 2016.

[105] B. Hofner and A. Smith, "Boosted negative binomial hurdle models for spatiotemporal abundance of sea birds," in *Proceedings of the 30th International Workshop on Statistical Modelling*, pp. 221–226, 2015.

[106] S. Friedrichs, J. Manitz, P. Burger et al., "Pathway-based kernel boosting for the analysis of genome-wide association studies," *Computational and Mathematical Methods in Medicine*, vol. 2017, 17 pages, 2017.