Adapting Prediction Error Estimates for Biased Complexity Selection in High-Dimensional Bootstrap Samples

Harald Binder & Martin Schumacher

Universität Freiburg i. Br.

Nr. 100 December 2007

Zentrum für Datenanalyse und Modellbildung

Universität Freiburg

Eckerstraße 1

D–79104 Freiburg im Breisgau

und

Institut für Medizinische Biometrie und Medizinische Informatik

Universitätsklinikum Freiburg

Stefan-Meier-Straße 26

D–79104 Freiburg im Breisgau

binderh@fdm.uni-freiburg.de

ms@imbi.uni-freiburg.de

Abstract

The bootstrap is a tool that allows for efficient evaluation of prediction performance of statistical techniques without having to set aside data for validation. This is especially important for high-dimensional data, e.g. arising from microarrays, because there the number of observations is often limited. The statistical technique to be evaluated has to be applied to every bootstrap sample in the same manner it would be used on new data. This includes selection of complexity, e.g. the number of boosting steps for gradient boosting algorithms. Using the latter, we demonstrate in a simulation study that complexity selection in conventional bootstrap samples, drawn with replacement, is biased in many scenarios. This results in models that are much more complex compared to models selected in the original data. Potential remedies for this complexity selection bias, such as alternatively using a fixed level of complexity or of using sampling without replacement are investigated and it is shown that the latter works well in many settings. We focus on high-dimensional binary response data, with bootstrap .632+ estimates of the Brier score for performance evaluation, and censored time-to-event data with .632+ prediction error curve estimates. The latter, with the modified bootstrap procedure, is then applied to an example with microarray data from patients with diffuse large B-cell lymphoma.

Keywords: Bootstrap; Complexity bias; High-dimensional data; Prediction performance.

1 Introduction

When high-dimensional data, arising e.g. from techniques such as microarrays, is to be used for improved judgement of prognosis for patients, statistical techniques have to be employed that allow for building prediction models from such data. Prediction performance of a statistical model fitted to one data set is often evaluated using a separate validation set. Evaluating prediction performance on the training data would

result in overoptimism. The problem however is that, especially for techniques such as microarrays, where each single observation is potentially very expensive to obtain, it is not advisable to reduce the size of the training data by setting aside data for validation.

The bootstrap approach (Efron, 1983) provides an attractive alternative for evaluating prediction performance, especially when employing the .632+ correction proposed by Efron and Tibshirani (1997). Bootstrap samples are generated by drawing observations from the original data and a predictive model is fitted in each of these samples. Prediction performance is then evaluated using the observations not included in the respective bootstrap samples. The general properties of the bootstrap for prediction error estimation and how it compares to alternatives, such as cross-validation, have already been examined for applications with high-dimensional data (Lusa et al., 2006; Molinaro et al., 2005; Jiang and Simon, 2007) and in other bioinformatics contexts (Fu et al., 2005). However, one limitation of the existing studies is, that they employ only very simple approaches for building prediction models, where either no complexity parameter has to be chosen or such parameters are set to fixed values. For example, Molinaro et al. (2005) reduce the number of features to 20 (by univariate t-tests) before applying model fitting techniques. As will be seen, this avoids a complexity selection bias that can occur in bootstrap samples, but fails to provide guidance when selection of a model complexity parameter is wanted.

One modern class of statistical techniques, that requires selection of a complexity parameter, are gradient boosting algorithms (Friedman, 2001; Bühlmann and Yu, 2003). They can be used to estimate parameters in a high-dimensional setting, e.g. for generalized linear, generalized additive, and survival models, by stepwise maximization of a loss function. For deriving a final predictive model, the number of boosting steps to be used has to be decided on. This is typically performed by automatic techniques, such as

cross-validation. Therefore, to obtain realistic bootstrap estimates of prediction performance, it is important to repeat this complexity selection step in every single bootstrap sample. Otherwise the resulting estimates may be overoptimistic (Simon et al., 2003; Dupuy and Simon, 2007; Zhu et al., 2008).

However, it is still unclear how well automatic complexity selection works in high-dimensional bootstrap samples. The results of Steck and Jaakkola (2003) indicate that in bootstrap samples drawn with replacement a larger level of complexity is selected compared to the original data, when model selection criteria such as AIC or BIC are used. This is contrary to what one would expect, as bootstrap samples contain less information compared to the original data. Intuitively this would correspond to selection of a smaller level of model complexity. There is also a result that indicates a distortion with respect to the distribution of p-values from univariate tests in bootstrap samples drawn with replacement (Strobl et al., 2007). When considering calculation of univariate p-values as a check for including covariates into a predictive model, this also corresponds to a complexity selection bias in bootstrap samples.

For model selection criteria such as AIC or BIC, Steck and Jaakkola (2003) propose a correction, that leads to approximately unbiased model selection in their application. As an more general alternative, that applies for all kinds of automatic model selection techniques, they suggest to use bootstrap samples drawn without replacement. This is also what is used by Strobl et al. (2007) for bias correction. A further alternative would be to employ parametric bootstrap techniques (see Liao and Chin, 2007, for example), but we will not investigate this option in the present paper.

In the following we will systematically investigate the extent of the complexity selection bias for predictive models fitted to high-dimensional bootstrap samples. Specifically, we are interested in how this bias affects bootstrap-based prediction error estimates. In Section 2, bootstrap techniques for prediction error estimation, specifically .632+ estimates,

and error measures for binary response data and censored time-to-event settings will be considered. Section 3 then presents a simulation study for evaluating a potential bias with respect to error estimates for these two types of response. This includes a short overview of gradient boosting techniques, which will be employed for fitting predictive models. The resulting recommendations for avoiding bias in prediction error estimates will then be illustrated with microarray survival data from patients with diffuse large B-cell lymphoma in Section 4. Concluding remarks are given in Section 5.

2 Prediction error estimation

2.1 Bootstrap .632+ estimates

In the following we shortly review bootstrap .632+ prediction error estimates, closely following the notation in Efron and Tibshirani (1997) and Gerds and Schumacher (2007). The aim is to evaluate the prediction performance of a procedure fitted to given training sample observations $\mathbf{x} = \{x_1, \ldots, x_n\}$, which are a random sample, where a single observation $x_i = (y_i, z_i)$ contains an observed response y_i and a covariate vector $z_i = (z_{i1}, \ldots, z_{ip})'$. After fitting some statistical model a risk prediction rule $r_{\mathbf{x}}(z)$ can be derived, which predicts the response y from covariate information z.

Given some discrepancy function $Q(y_0, r_{\mathbf{x}}(z_0))$, which quantifies the error of the risk prediction rule $r_{\mathbf{x}}(z_0)$ for a new observation $x_0 = (y_0, z_0)$, the quantity of interest is the *true error*

$$Err = E[Q(y_0, r_{\mathbf{x}}(z_0))], \tag{1}$$

where $\mathbf{x}$ and $r_{\mathbf{x}}(z)$ are fixed and the expectation is taken over the distribution of the random quantity $x_0 = (y_0, z_0)$.

Using the *apparent error*

$$\overline{err} = \frac{1}{n} \sum_{i=1}^{n} Q(y_i, r_{\mathbf{x}}(z_i)) \tag{2}$$

as an estimate for the true error (1) will typically result in overoptimism with respect to estimated prediction performance, as the same data is used for fitting and error estimation. A conservative approach is to evaluate the prediction performance on new data. For this purpose often validation data is set aside, i.e., not all available data is used for fitting. However, especially in high-dimensional settings, such as e.g. with microarray data, there are often only relatively few observations available and every observation set aside impairs the fit of the model.

There are basically two alternatives for more realistic prediction error estimates: cross-validation and the bootstrap. Cross-validation partitions the data into folds and evaluates prediction performance on every single fold with models fitted to the data from the remaining folds. In jacknife estimation every fold contains only one observation. Therefore the bias, resulting from not having all observations for fitting, is minimal, but the variability of the estimates is very large. Generally, large variability of the prediction performance estimates is a downside of the cross-validation approach. Therefore we focus on the bootstrap in the following.

The bootstrap for prediction error estimation (Efron, 1983) imitates drawing of a new validation set from the population, by randomly drawing observations from the original data, i.e., the empirical distribution. That way several bootstrap samples are generated. In each of these samples the model under investigation is fitted, including selection of potential complexity parameters. For assessing prediction performance, for every bootstrap sample the mean value of the discrepancy function is evaluated for the observations not in the respective bootstrap sample. This is also called bootstrap cross-validation or out-of-bag estimation. With B bootstrap samples, and b_0 being the number of observations $i \in \mathbf{x}_b^0$ not in bootstrap sample $b = 1, \ldots, B$, the *bootstrap cross-validation* estimate

is

$$\widehat{Err}_{B0} = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{b_0}\sum_{i\in\mathbf{x}_b^0} Q(y_i, r_{\mathbf{x}_b}(z_i)), \tag{3}$$

where $r_{\mathbf{x}_b}(z_i)$ is the risk prediction rule resulting from model fitting in bootstrap sample b.

As will be seen, the way of drawing observations for bootstrap samples, with or without replacement, and the number of "new" observations drawn, critically affects the estimate (3). We consider two variants: bootstrap sampling of n observations with replacement, and drawing of $0.632n$ observations without replacement. The motivation for the fraction 0.632 in the latter variant is, that in bootstrap sampling with replacement, approximately $0.632n$ unique observations will enter into a bootstrap sample. Therefore both variants are expected to result in bootstrap samples with a similar level of information in terms of unique observations. Efron (1983) indicates that also half-sample cross-validation, i.e., drawing $n/2$ observations without replacement, is closely related to bootstrap sampling with replacement, but we did not find this in preliminary experiments for our application.

The bootstrap cross-validation estimate (3) is known to be biased upwards. As a correction, Efron and Tibshirani (1997) propose the .632+ estimate, which is a weighted linear combination of the apparent error (2) and the bootstrap cross-validation estimate (3), i.e.,

$$\widehat{Err}_{.632+} = (1-w)\overline{err} + w\widehat{Err}_{B0}. \tag{4}$$

The weight w is obtained via $w = 0.632/(1 - .368\hat{R})$ from the relative overfit

$$\hat{R} = \frac{\widehat{Err}_{B0} - \overline{err}}{\widehat{NoInf} - \overline{err}},$$

which is based on an estimate of the *no-information error* $NoInf$, which would apply

if z and y were independent. This quantity is estimated by

$$\widehat{NoInf} = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{k=1}^{n} Q(y_i, r_{\mathbf{x}}(z_k)).$$

2.2 Measures of prediction error

For binary response data the response y takes only two values, 0 and 1, and risk prediction rules can be obtained from fitted models via the predicted probabilities $\hat{p}(z) = P(y = 1|z)$, i.e., $r_{\mathbf{x}}(z) = \hat{p}(z)$. When some cutoff on the latter is used, to arrive at actual class predictions, estimates of misclassification rate can be obtained. However, often the predicted probability is more interesting than class assignment. An example, where these two objectives lead to different results, is seen for boosting techniques, which seem to be resistant to overfitting when only misclassification rates are considered, but at the same time show strong signs of overfitting when the predicted probabilities are inspected Bühlmann and Yu (2003). While misclassification rate will be rather stable as long as the behavior near the classification boundary stays the same, some error measure that focusses more on the predicted probabilities will also be sensible to changes far away from that boundary. As we prefer the latter, the Brier score (Brier, 1950), with corresponding discrepancy function

$$Q_{Brier}(y_0, \hat{p}(z_0)) = (y_0 - \hat{p}(z_0))^2, \tag{5}$$

will be used for estimating prediction performance in the following.

When the response of interest is an event time, there often are censored observations, i.e., an observation $x_i = (t_i, \delta_i, z_i)$ comprises not only of an observed time t_i and a covariate vector z_i, but also of a censoring indicator δ_i. Given survival time T_i and censoring time C_i (with $t_i = \min(T_i, C_i)$), the latter is defined by $\delta_i = I(T_i \leq C_i)$, where $I()$ is an indicator function that takes value 1 if its argument is true, and 0 otherwise.

Denoting by $Y_i(t)$ the state of individual i at time t, with value 1 if no event has occurred yet, and 0 otherwise, Gerds and Schumacher (2007) suggest to employ a time dependent discrepancy function

$$Q(t; Y_0(t), r_{\mathbf{x}}(t; z_i)) = (Y_0(t) - r_{\mathbf{x}}(t; z_i))^2$$

for bootstrap prediction error estimation. This means that the Brier score (5) is evaluated at time t for a risk prediction rule $r_{\mathbf{x}}(t; z_i)$, that returns the predicted probability of still being free of an event at time t, given the covariate information in z_i.

Tracking the true error (1) for this discrepancy function over time results in prediction error curves. For consistent estimates of the latter based on actual data, weights that account for censoring have to be introduced (Graf et al., 1999; Gerds and Schumacher, 2006). For example, the *apparent prediction error curve* is obtained by

$$\overline{err}(t) = \frac{1}{n} \sum_{i=1}^{n} (Y_i(t) - r_{\mathbf{x}}(t; z_i))^2 W(t, x_i)$$

with inverse probability of censoring weights

$$W(t, \hat{G}, x_i) = \frac{I(T_i \leq t, T_i \leq C_i)}{\hat{G}(T_i-)} + \frac{I(T_i > t)}{\hat{G}(t)},$$

where $\hat{G}(t)$ denotes a uniformly consistent estimate of the conditional probability of being uncensored at time t, given the history. For simplicity, we assume in the following that the censoring mechanism is independent of the survival and the history, and therefore, like in Graf et al. (1999), the Kaplan-Meier estimate can be used for $\hat{G}(t)$.

Adapting the .632+ estimate (4) by tracking the estimated prediction error over time and employing inverse probability of censoring weights, *bootstrap .632+ prediction error curve estimates* are obtained. For more details see Gerds and Schumacher (2007); Schumacher

et al. (2007).

3 Simulation study

3.1 Data structure

Two types of response, binary and censored time-to-event, are considered in the following. As we are mainly interested in a potential bootstrap complexity bias in high-dimensional settings, a large, varying number of covariates $p \in \{200, 1000, 5000\}$ is considered, with a fixed number of $n = 100$ observations for binary response settings and $n = 200$ for censored time-to-event settings.

For the structure of the covariates, an uncorrelated and a correlated scenario is used. In the uncorrelated scenario, each covariate $z_{ij}, i = 1, \ldots, n, j = 1, \ldots, p$, is drawn from a $N(0, 1)$ normal distribution, i.e., there is zero covariance between covariates. The correlated scenario, which is similar to the one employed in Bair and Tibshirani (2004), mimics microarray data structure. For each covariate j an error term $\epsilon_{ij} \sim N(0, 1)$ is generated and the covariate values are determined by

$$z_{ij} = \begin{cases} -1 + \epsilon_{ij} & \text{if } i \leq 0.5n, j \leq 0.05p \\ 1 + \epsilon_{ij} & \text{if } i > 0.5n, j \leq 0.05p \\ 1.5 \cdot I(u_{i1} < 0.4) + \epsilon_{ij} & \text{if } 0.05p < j \leq 0.1p \\ 0.5 \cdot I(u_{i2} < 0.7) + \epsilon_{ij} & \text{if } 0.1p < j \leq 0.2p \\ 1.5 \cdot I(u_{i3} < 0.3) + \epsilon_{ij} & \text{if } 0.2p < j \leq 0.3p \\ \epsilon_{ij} & \text{if } j > 0.3p \end{cases},$$

where u_{ij} are uniform random variables on the interval $[0; 1]$ and $I()$ is an indicator function that takes value 1 if its argument is true, and 0 otherwise. This results in correlations of about 0.50 for $j \leq 0.05p$, 0.35 for $0.05p < j \leq 0.1p$, 0.05 for $0.1p < j \leq$

$0.2p$, 0.32 for $0.2p < j \leq 0.3p$, and no correlation otherwise.

The linear predictor η_i is determined by

$$\eta_i = z_i'\beta \quad i = 1, \ldots, n,$$

where $z_i = (z_{i1}, \ldots, z_{ip})'$ are the covariate vectors and $\beta = (\beta_1, \ldots, \beta_p)'$ is the true parameter vector with

$$\beta_j = \begin{cases} c_e & \text{if } j \cdot 200/p \in \{1, 3, 5, 7, 9\} \\ -c_e & \text{if } j \cdot 200/p \in \{2, 4, 6, 8, 10\} \\ 0 & \text{otherwise} \end{cases} .$$

This corresponds to 10 informative covariates, where the constant c_e determines the amount of information in the data. We investigate three settings with "weak", "medium", and "strong" effect. The c_es are chosen such that prediction performance in uncorrelated scenarios and in scenarios with correlated covariates is approximately equal. For the correlated covariate scenarios this corresponds to $c_e \in \{0.075, 0.1, 0.15\}$ for a binary response and to $c_e \in \{0.05, 0.075, 0.1\}$ for censored time-to-event data. For scenarios with uncorrelated covariates a "strong" information setting is difficult to obtain, as all the information has to be contained in 10 covariates, while in a correlated setting a whole group of correlated covariates can carry information. Therefore in the former we investigate only "weak" and "medium" effects, corresponding to $c_e \in \{1, 2\}$ for binary response data and $c_e \in \{0.5, 1\}$ for censored time-to-event settings.

For the binary response setting, responses $y_i, i = 1, \ldots, n$, are generate from Binomial distributions $y \sim B(\mu_i, 1)$, where $\mu_i = \exp(z_i'\beta)/(1 + \exp(z_i'\beta))$. For time-to-event data, survival times T_i are determined by a Cox-exponential model (see Bender et al., 2005,

for example), i.e.,

$$T_i = -\frac{U_i}{\lambda \exp(z_i'\beta)} \ , \ \ i = 1, \ldots, n,$$

where U_i is generated from a uniform distribution over the interval $[0; 1]$ and $\lambda = 0.1$, corresponding to a mean baseline survival time of 10. Censoring times C_i are determined by $C_i = -U_{Ci}/\lambda_C, i = 1, \ldots, n$, where again U_{Ci} is generated from a uniform distribution over the interval $[0; 1]$ and $\lambda_C = 0.1$. The latter results in censoring for about 50% of the observations. The observed times are then determined by $t_i = \min(T_i, C_i)$ and the censoring indicator is obtained by $\delta_i = I(T_i \leq C_i)$.

For each response type and each combination of the number of covariates p, correlation structure, and effect size, 50 data sets and corresponding tests sets of size $n_{new} = 1000$, for determining true prediction performance, are generated.

3.2 Fitting procedure

For fitting in the binary response setting, the response y_i is assumed to be from a Binomial distribution $y_i \sim B(p(z_i), 1)$, where $p(z_i)$ follows the model

$$p(z_i) = h(\eta_i) = h(F(z_i; \beta))$$

with response function $h(\eta) = \exp(\eta)/(\exp(\eta) + \exp(-\eta))$. The function $F(z; \beta)$, which has to be estimated, is taken to depend on a parameter vector β. For censored time-to-event data we use a Cox proportional hazards model

$$\lambda(t|z_i) = \lambda_0(t) \exp(F(z_i; \beta))$$

with baseline hazard $\lambda_0(t)$, also requiring estimation of the function $F(z; \beta)$, depending on a parameter vector β.

For parameter estimation we use a gradient boosting approach (Friedman, 2001; Bühlmann and Yu, 2003). In general form, such algorithms estimate a function $F(z;\beta)$ by minimizing expected loss $E[L(y, F(z;\beta))]$, where $L(y, F(z;\beta))$ is a loss function, that takes fitted functions $F(z;\hat{\beta})$ and responses y (or (t,δ), for censored time-to-event data) as its arguments. This is performed in $k = 1,\ldots,m$, boosting steps by determining in each step the negative gradient

$$-g_i^{(k)} = -\left[\frac{\partial L(y_i, F(z;\beta))}{\partial F(z;\beta)}\right]_{F(z;\beta)=F^{(k-1)}(z;\hat{\beta}^{(k-1)})} \quad i = 1,\ldots,n,$$

with respect to the current estimate $F^{(k-1)}(z;\hat{\beta}^{(k-1)})$ at the observations, and using a weak learner to obtain a fit $f^{(k)}(z;\hat{\gamma}^{(k)})$ by estimating a parameter (vector) $\gamma^{(k)}$, i.e., the negative gradient is taken to be the response. The overall estimate is then updated via

$$F^{(k)}(z;\hat{\beta}^{(k)}) = F^{(k-1)}(z;\hat{\beta}^{(k-1)}) + \delta f^{(k)}(z;\hat{\gamma}^{(k)}),$$

where δ is some small value providing for cautious updates.

When for $F(z;\beta)$ the simple linear form $F(z;\beta) = z'\beta$, with parameter vector $\beta = (\beta_1,\ldots,\beta_p)'$, is used and in each boosting step k only one element of this parameter vector is updated by choosing between per-covariate updates $f_j^{(k)}(z;\gamma_j^{(k)}) = \gamma_j^{(k)} z_j$, componentwise boosting is obtained. This results in sparse estimates, where only few elements of an estimates parameter vector $\hat{\beta}$ will have a value unequal to zero (Bühlmann, 2006). The resulting fits will be similar to those from path algorithms, which estimate the parameter vector by penalizing the L_1 norm (Park and Hastie, 2007).

We will use componentwise boosting in the following, more specifically the implementation in the R package "mboost", described in Hothorn and Bühlmann (2006). For

binary response settings the loss function employed is the deviance

$$L(y, F(z;\beta)) = -2(y_i \log(p(z)) + (1-y)\log(1-p(z))),$$

i.e., minus two times the log-likelihood. For censored time-to-event settings the negative Cox partial log-likelihood is used as a loss function (Ridgeway, 1999).

The essential parameter, that determines model complexity for gradient boosting approaches, is the number of boosting steps m. While in some special cases (e.g. in continuous response settings) model selection criteria such as AIC or BIC are available for selecting this parameter, this is not the case for gradient boosting for the Cox model. We therefore employ 5-fold cross-validation, with respect to misclassification rate in binary response examples and with respect to the partial log-likelihood for censored time-to-event data. This procedure is applied in each of the 50 original samples for each setting. For prediction error estimation we use $B = 100$ bootstrap samples. As there are two variants, sampling with and without replacement, this amounts to $2 \times 50 \times 100$ bootstrap samples for each setting, where gradient boosting together with cross-validation is applied.

3.3 Results

3.3.1 Binary response setting

Table 1 shows the median number of boosting steps selected by 5-fold cross validation for binary response models in various settings. The number of boosting steps selected for scenarios without correlation is considerable larger compared to correlated scenarios. The reason for this might be that in the former it takes many boosting steps to build up adequate estimates of the large true parameter values, as there will always be some boosting steps that accidentally target other covariates. In contrast, in a setting with

Table 1: Median number of boosting steps selected by 5-fold cross-validation for binary response gradient boosting in 50 original samples and 50×100 bootstrap samples drawn with replacement and without replacement for scenarios with a varying number of covariates p, with (cor) and without correlation (uncor), and with varying effects sizes.

effect	p	original sample		bootstrap w.repl.		without replacement	
		uncor	cor	uncor	cor	uncor	cor
weak	200	65.5	6	295	183	20	6
	1000	17.5	9	184	172.5	4	6
	5000	12	8	154	149	3	5
medium	200	135	16	362	222	38	12
	1000	46	10.5	210	181	7	8
	5000	9.5	12	159	167	3	9
strong	200	-	29	-	254	-	20
	1000	-	21.5	-	215	-	19
	5000	-	23.5	-	196	-	21

a large number covariates that are correlated with the covariates with non-zero true parameter value, every boosting step that targets one of the former extracts at least some information.

Comparing the number of boosting steps selected in bootstrap samples drawn with replacement to that selected in the original data, a clear tendency towards a larger number of boosting steps is seen. This bootstrap complexity bias is present regardless of the number of covariates, the correlation structure, and the effect size. So the complexity bias seen by Steck and Jaakkola (2003), when selecting the complexity of fitted graphs by AIC or BIC, is also seen for linear predictor binary response models fitted by gradient boosting, with complexity selected by cross-validation. Therefore the number of boosting steps selected in bootstrap samples drawn with replacement is no good estimate for the number of boosting steps selected in the original data. The number of boosting steps selected by cross-validation in bootstrap samples drawn without replacement is seen to be systematically smaller compared to the original data. So the former also is no good estimate for the latter. However, this is not our objective.

A second quantity that indicates model complexity, besides the number of boosting

Table 2: Median number of non-zero parameter estimates (par) resulting from 5-fold cross-validation for binary response gradient boosting, mean hit rates (hit), and mean false alarm rates (false) in 50 original samples and 50×100 bootstrap samples drawn with replacement and without replacement for scenarios with a varying number p of correlated covariates and varying effects sizes.

effect	p	original sample			bootstrap w.repl.			without replacement		
		par	hit	false	par	hit	false	par	hit	false
weak	200	4	0.216	0.019	32	0.287	0.147	3	0.135	0.022
	1000	6.5	0.080	0.009	39	0.105	0.037	4	0.052	0.007
	5000	6.5	0.024	0.002	41	0.026	0.008	4	0.012	0.001
medium	200	7	0.384	0.032	32	0.405	0.144	6	0.273	0.028
	1000	7	0.132	0.010	39	0.147	0.037	5	0.086	0.007
	5000	9.5	0.046	0.002	42	0.048	0.008	7	0.029	0.002
strong	200	11	0.510	0.041	31	0.504	0.132	9	0.406	0.033
	1000	13	0.242	0.011	37	0.238	0.034	11	0.181	0.011
	5000	17	0.098	0.003	40	0.090	0.008	14	0.067	0.003

steps, is the number of non-zero elements of the estimated parameter vectors, i.e., the number of covariates deemed influential. The median number of such parameters is given in Table 2 for the scenarios with correlated covariates, together with mean hit rates and false alarm rates with respect to identification of influential covariates. Similar to the selected number of boosting steps, the number of non-zero parameters fitted in bootstrap samples drawn with replacement is much larger compared to the original data. As expected, the median number of parameters selected in bootstrap samples drawn without replacement is smaller compared to the original samples. Note also that the number of non-zero parameters is similar for all levels of information, when using bootstrap samples drawn with replacement. Only in the original samples and in samples drawn without replacement the number of non-zero parameters seems adapts to the amount of information in the data. This is a further aspect where automatic model building behaves differently in bootstrap samples drawn with replacement.

The bias towards selecting more complex models in bootstrap samples drawn without replacement is also seen from the mean hit rates in Table 2. In several instances this

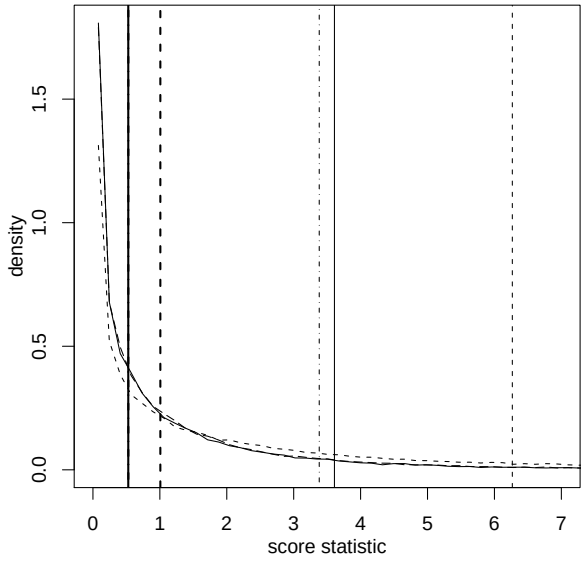

Figure 1: Density of the univariate, per-covariate score statistic obtained from 50 repetitions with a binary response and $p = 1000$ standardized, correlated covariates with medium effect in the original data (solid curve), bootstrap samples drawn with replacement (dashed curve), and bootstrap samples drawn without replacement (dash-dotted curve). Median and 90% quantiles are indicated by thick and thin vertical lines respectively.

quantity is larger in the bootstrap samples compared to the original data. As the former contain less information, this obviously comes at the price of increased false alarm rates. The latter are consistently larger, at least by a factor of 3, even when the hit rates are similar to the original data. In contrast, in bootstrap samples drawn without replacement the mean false alarm rates are smaller compared to the original data in almost all scenarios, indicating more cautious model selection. Roughly extrapolating the false alarm rates towards a situation where the hit rates would be similar to that from bootstrap samples drawn with replacement, it seems that the overall covariate identification performance might be better in bootstrap samples drawn without replacement compared to sampling with replacement.

One potential source of the complexity bias, i.e., the bias with respect to the number

of boosting steps as well as with respect to the number of covariates with non-zero coefficients, is the change of correlation structure between the covariates and the response due to bootstrap sampling. Figure 1 shows the density for the univariate, per-covariate score statistic (which is closely related to the correlation between covariates and the response) for the scenario with $p = 1000$ correlated covariates and a medium level of information. While the density resulting from bootstrap sampling without replacement (dash-dotted curve) almost coincides with the density obtained from the original data (solid curve), the density for bootstrap samples drawn with replacement (dashed curve) is distinctly different. It has considerably more mass in the tail, probably resulting from artificial covariate-response relationships introduced by replicated observations.

We performed a small simulation study (not reported here) where covariates were generated to follow densities such as seen in Figure 1. There it could be seen that the density of the score statistic determines the selected model complexity, even when there are no replicated observations. So it seems that the larger mass in the tail directly results in a larger number of boosting steps and in more complex models being selected. This also means that the complexity bias can not (only) be due to the potential overlap of training folds and test folds with respect to observations in cross-validation, which is induced by bootstrap sampling with replacement.

The main target of the present study is a potential bootstrap bias with respect to prediction error estimation. To investigate in which way such a potential bias depends on the sampling and the complexity selection scheme, three variants of .632+ estimates are considered. For estimates based on bootstrap samples drawn with replacement, the number of boosting steps either is selected in each bootstrap sample separately, or the same, fixed number of boosting steps is used, which is determined by cross-validation in the original data. Our initial motivation for considering the latter complexity selection scheme had been to quantify the overoptimism incurred by leaving out the model building

step of complexity selection (Simon et al., 2003). However, preliminary experiments indicated that reasonable prediction error estimates might nevertheless be obtained. Therefore we include this variant as a competitor in its own right. For bootstrap samples drawn without replacement the number of boosting steps, i.e. model complexity, is determined in every single bootstrap sample by cross-validation.

With $\widehat{Err}_{.632+}$ being the .632+ estimate of the Brier score, and Err being the true Brier score, the relative bias is obtained via

$$\text{RelBias} = \frac{\widehat{Err}_{.632+} - Err}{Err}.$$

Table 3 shows the mean relative bias for various scenarios. For each of these the .632+ variant with the smallest mean relative bias is indicated by boldface printing. The largest relative bias is seen for .632+ estimates based on bootstrap samples drawn with replacement, with complexity selected in every bootstrap sample. The complexity selection bias seen in Tables 1 and 2 seems to directly translate into a large bias with respect to prediction error estimation. In contrast, the estimates based on bootstrap samples drawn without replacement exhibit a much smaller, more reasonable bias, which only in 3 scenarios is on average larger than 5% of the true error. So the smaller amount of model complexity selected does not seem to be a disadvantage.

The small bias of the .632+ estimates obtained with a fixed number of boosting steps, in bootstrap samples drawn with replacement, is surprising, as not all model building steps are performed in every bootstrap sample. While for uncorrelated covariates the fixed complexity estimates even seem to be on par with the estimates obtained from sampling without replacement, the latter perform better in the correlated scenarios. As these are more realistic, at least for microarray applications, the approach without replacement should be preferred. On theoretical grounds it also has the advantage that it does not omit model building steps.

Table 3: Mean relative bias of .632+ estimates for the Brier score (with standard errors in parentheses), for binary response models fitted by gradient boosting, based on bootstrap samples drawn with replacement, with the number of boosting steps either selected in every bootstrap sample (w. repl.) or taken to be the same as in the original data (original step), and based on bootstrap samples drawn without replacement (w/o repl.), for scenarios with a varying number of covariates p, with (cor) and without correlation (uncor), and with varying effects sizes. The smallest mean bias in each scenario is printed in boldface.

effect	p	w. repl.	original step	w/o repl.
uncorrelated covariates				
weak	200	0.149 (0.023)	**0.052** (0.019)	0.072 (0.019)
	1000	0.050 (0.016)	0.002 (0.013)	**0.001** (0.012)
	5000	0.038 (0.010)	**-0.001** (0.009)	-0.012 (0.010)
medium	200	0.176 (0.030)	**0.104** (0.027)	0.150 (0.029)
	1000	0.103 (0.019)	0.036 (0.016)	**0.035** (0.015)
	5000	0.090 (0.014)	0.033 (0.008)	**0.025** (0.008)
correlated covariates				
weak	200	0.058 (0.010)	**-0.006** (0.012)	0.008 (0.012)
	1000	0.065 (0.013)	-0.015 (0.011)	**-0.002** (0.012)
	5000	0.062 (0.012)	-0.005 (0.012)	**0.003** (0.012)
medium	200	0.109 (0.021)	-0.030 (0.016)	**-0.011** (0.017)
	1000	0.119 (0.016)	**0** (0.011)	0.012 (0.012)
	5000	0.080 (0.019)	-0.025 (0.016)	**-0.011** (0.016)
strong	200	0.099 (0.025)	-0.031 (0.019)	**-0.017** (0.019)
	1000	0.052 (0.023)	-0.049 (0.016)	**-0.031** (0.017)
	5000	**-0.014** (0.032)	-0.080 (0.026)	-0.066 (0.026)

Table 4: Median number of boosting steps selected by 5-fold cross-validation for Cox model gradient boosting in 50 original samples and 50×100 bootstrap samples drawn with replacement and without replacement for scenarios with a varying number of covariates p, with (cor) and without correlation (uncor), and with varying effects sizes.

effect	p	original sample		bootstrap w.repl.		without replacement	
		uncor	cor	uncor	cor	uncor	cor
weak	200	431	34	986	481	216	28
	1000	302	33.5	754.5	571.5	72	24
	5000	108	36.5	611	496	22	22
medium	200	972.5	58.5	1893	510	715	48
	1000	800.5	48.5	1203	589.5	354	42
	5000	532	49	745	515	102	40
strong	200	-	74.5	-	550	-	66
	1000	-	69.5	-	605	-	59
	5000	-	63	-	521	-	57

3.3.2 Censored time-to-event setting

In this section we will investigate the performance of .632+ bootstrap prediction error curve estimates in censored time-to-event settings. Table 4 shows the median number of boosting steps selected in the various scenarios by 5-fold cross-validation. The results are very similar to the binary response settings. In the uncorrelated scenarios a much larger number of boosting steps seems to be required for similar prediction performance. Again, using bootstrap samples drawn without replacement results in a number of boosting steps that is considerably larger compared to the number of steps chosen in the original data. Also, the number of boosting steps chosen in bootstrap samples drawn without replacement is seen to be smaller compared to the latter. So, the bootstrap complexity bias described by Steck and Jaakkola (2003) and seen for the binary response examples, also seems to be present for high-dimensional Cox survival models.

Figure 2 shows the mean .632+ prediction error curve estimates for scenarios with $p = 1000$ correlated covariates with a weak (left panel) or a medium effect (right panel). In both scenarios the true prediction error (solid curve) is below the Kaplan-Meier bench-

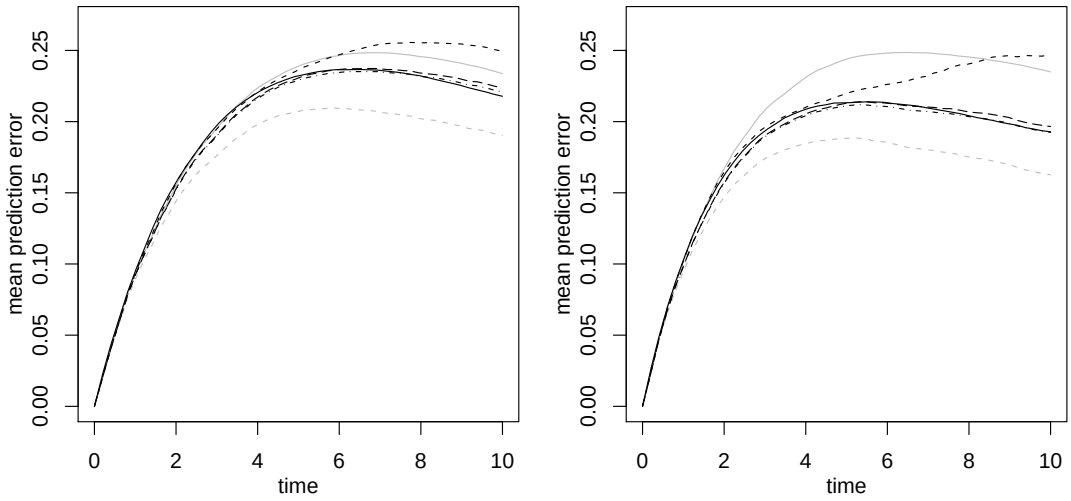

Figure 2: Mean .632+ prediction error curve estimates for Cox survival models fitted by gradient boosting, based on bootstrap samples drawn with replacement, with the number of boosting steps either selected in every bootstrap sample (dashed curves) or taken to be the same as in the original samples (long dashed curves), and based on bootstrap samples drawn without replacement (dash-dotted curves), for scenarios with $p = 1000$ correlated covariates with small (left panel) and medium (right panels) effect size. The mean true prediction error (solid black curves), apparent error (dashed grey curves), and the Kaplan-Meier benchmark (solid grey curves) are given as a reference.

mark (solid grey curve), i.e., at least some information can be extracted. The .632+ estimate based on bootstrap samples drawn with replacement in combination with a fixed number of boosting steps, selected in the original data, as well as the estimate based on bootstrap samples drawn without replacement closely track the true prediction error. In contrast, the estimate based on bootstrap samples drawn with replacement and a variable number of boosting steps considerably overestimate the prediction error. For example, for the scenario with weak covariate effect (left panel) the latter estimate would even wrongly suggest that gradient boosting performs worse than the Kaplan-Meier benchmark. For the example with a medium covariate effect (right panel), the bootstrap complexity bias results not only in a simple upwards shift, but in a distortion of the shape of the true prediction error curve. So a simple correction afterwards, e.g. by modifying the weights in (4), does not seem to be possible.

For a comparison beyond visual inspection of (mean) prediction error curve estimates, we consider the integrated differences (up to time 10) between the true prediction error a .632+ estimate. The integrated difference is divided by the area under the true prediction error curve to arrive at a measure of relative bias. A positive value of this quantity indicates that on average the prediction error is overestimated. Table 5 shows the mean relative bias for the three types of .632+ estimates under consideration in various scenarios.

The .632+ estimate based on bootstrap samples drawn with replacement, in combination with a variable number of boosting steps selected in every bootstrap samples, consistently performs worst for prediction error curve estimation. Even the difference to the approach showing the second to best performance is seen to be very large in most scenarios. While in the binary response scenarios there did not seem to be a systematic effect of p on the size of the (relative) bias, there is a clear pattern for the censored time-to-event setting. The bias is larger for a larger number of covariates (regardless of whether these

Table 5: Mean relative bias of .632+ prediction error curve estimates (with standard errors in parentheses), for Cox survival models fitted by gradient boosting, based on bootstrap samples drawn with replacement, with the number of boosting steps either selected in every bootstrap sample (w. repl.) or taken to be the same as in the original data (original step), and based on bootstrap samples drawn without replacement (w/o repl.), for scenarios with a varying number of covariates p, with and without correlation, and with varying effects sizes. The smallest mean relative bias in each scenario is printed in boldface.

effect	p	w. repl.	original step	w/o repl.
uncorrelated covariates				
weak	200	0.062 (0.005)	0.033 (0.005)	**0.007** (0.005)
	1000	0.072 (0.008)	0.048 (0.007)	**0.006** (0.008)
	5000	0.065 (0.007)	0.053 (0.006)	**0.021** (0.006)
medium	200	0.094 (0.009)	0.050 (0.008)	**0.011** (0.007)
	1000	0.119 (0.008)	0.087 (0.007)	**0.007** (0.007)
	5000	0.115 (0.009)	0.103 (0.007)	**0.004** (0.007)
correlated covariates				
weak	200	0.038 (0.006)	0.004 (0.005)	**-0.003** (0.005)
	1000	0.051 (0.007)	**-0.004** (0.006)	-0.011 (0.006)
	5000	0.074 (0.006)	**-0.006** (0.006)	-0.014 (0.007)
medium	200	0.045 (0.007)	**0.002** (0.006)	-0.009 (0.006)
	1000	0.089 (0.007)	**-0.005** (0.006)	-0.015 (0.006)
	5000	0.112 (0.008)	**-0.018** (0.007)	-0.029 (0.007)
strong	200	0.049 (0.007)	**-0.004** (0.007)	-0.014 (0.007)
	1000	0.098 (0.008)	**-0.014** (0.007)	-0.025 (0.006)
	5000	0.124 (0.009)	**-0.027** (0.007)	-0.038 (0.007)

are correlated or uncorrelated). Therefore prediction error curve estimation in high-dimensional settings seems to be particularly affected. The bias also is seen to be larger when there is more information in the data.

Comparing the estimates based on fixed complexity to estimates based bootstrap samples drawn without replacement, the situation is also different to the binary response setting. While there the former approach was sometimes found to be superior, in the survival setting both approaches are very close. The differences seem to depend on the specific kind of scenario. The .632+ estimates based on sampling without replacement display a tendency towards underestimating the prediction error, albeit only by a small amount, while the fixed complexity estimates tend to overestimate. Generally, both approaches seem to result in reasonable estimates of the true prediction error curve.

4 Application

In the following we are going to illustrate the three types of .632+ prediction error curve estimates, using microarray survival data from 240 patients with diffuse large B-cell lymphoma (DLBCL) (Rosenwald et al., 2002), where 7399 microarray features are available. The event of interest, death, occurred for 57% of the patients during follow up, where median follow up was 2.8 years. For an overview of analyses targeting this data set see Segal (2006).

While the simulation study focussed on gradient boosting for model fitting, the reported complexity selection bias can also be seen when other model fitting procedures are applied to high-dimensional data. Therefore, in addition to gradient boosting, Cox survival models are fitted by the CoxPath procedure (Park and Hastie, 2007) and the PC-PCR approach (Li and Gui, 2004). For preprocessing of the data and details of the procedures in this specific application see Schumacher et al. (2007). We also have a limited set of

simulation results for the latter two procedures, but do not report these here.

The number of boosting steps selected in the original data by 5-fold cross-validation for gradient boosting is 93. The median number selected in bootstrap samples drawn with replacement is 625.5, without replacement it is 58.5. This is consistent with the complexity bias seen in the simulation study. For the CoxPath procedure model complexity is described by the degrees of freedom of a fitted model. Using 5-fold cross-validation, a model with 54 degrees of freedom is obtained from the original data. The median of the degrees of freedom in bootstrap samples drawn with replacement is 66, without replacement it is 12.5. This indicates that also for the CoxPath procedure, bootstrap sampling with replacement results in more complex models, while in samples drawn without replacement less complex models are chosen, compared to the original data. Similarly, for the PC-PCR approach, which has the number of partial model components as its complexity parameter, 3 components are chosen in the original data, while the median is 4 components using bootstrap with replacement, and 3 without replacement. For the latter approach a p-value criterion (as suggested by Li and Gui, 2004), instead of cross-validation, is used for complexity selection. As nevertheless a complexity selection bias occurs, this indicates that this bias is not tied to a specific complexity selection procedure. While Steck and Jaakkola (2003) report a bias for complexity selection criteria such as AIC or BIC, in the simulation study we found it to be also present when employing cross-validation, and in the present example data it is now also seen to be present when using a p-value criterion. This is no surprise, as the change in dependency structure introduced by bootstrap sampling with replacement, seen e.g. from Figure 1, is expected to affect all kinds of complexity selection procedures.

Given that the complexity selection bias is seen in the present example for all three model fitting procedures, effects similar to the simulation study are also expected with respect to estimates of prediction error. Figure 3 shows the .632+ prediction error curve

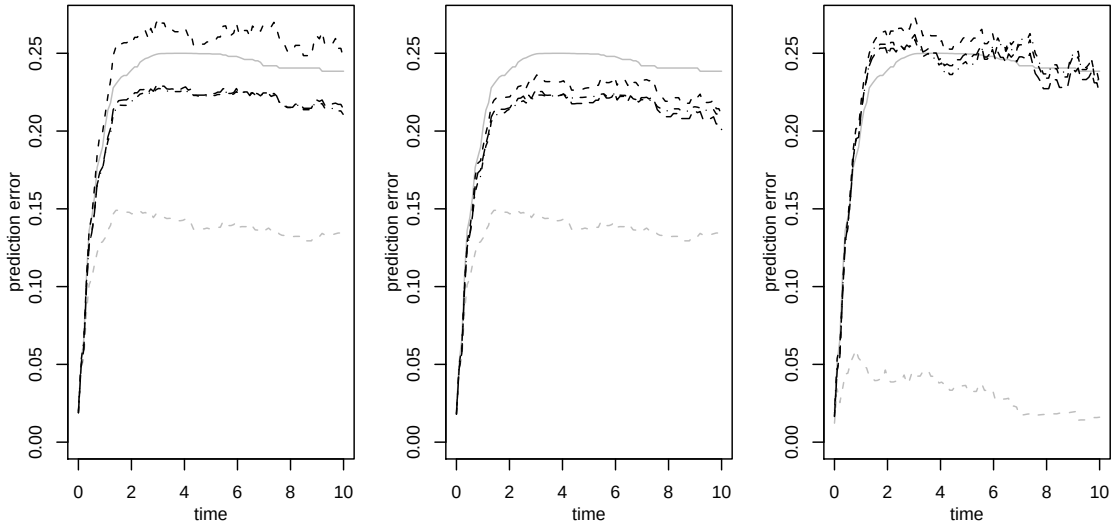

Figure 3: .632+ prediction error curve estimates for Cox survival models fitted to large B-cell lymphoma microarray survival data by gradient boosting (left panel), the CoxPath procedure (middle panel), and the PC-PCR approach (right panel), based on bootstrap samples drawn with replacement, with model complexity either selected in every single bootstrap sample (dashed curves) or taken to be the same as in the original data (long dashed curves), and based on bootstrap samples drawn without replacement (dash-dotted curves). The apparent error (dashed grey curves) and the Kaplan-Meier benchmark (solid grey curves) are given as a reference.

estimates for all model fitting approaches, based on bootstrap with replacement, using variable (dashed curves) or fixed complexity (long dashed curves), and based on bootstrap samples drawn without replacement (dash-dotted curves). For all three procedures the estimates based on bootstrap samples drawn without replacement are below the estimates based on bootstrap samples drawn with replacement. Based on the simulation results in Section 3.3.2, the former are believed to be more accurate. For gradient boosting the effect of the complexity selection bias seems to be especially large and problematic. The estimate based on bootstrap sampling with replacement would indicate that the model performs worse than the Kaplan-Meier benchmark, which does not use any covariate information. In contrast, the estimate based on bootstrap samples drawn without replacement probably correctly indicates that gradient boosting manages to extract information from the data. The estimates based on a fixed level of model complexity in bootstrap samples are close to the estimates based on bootstrap samples drawn without replacement. This also is in line with the results of the simulation study, where both estimates closely tracked the true prediction error.

5 Concluding remarks

Up to now a potential model complexity selection bias in (high-dimensional) bootstrap samples hardly seems to have been investigated and only scarcely appears in the literature. In the present study, we systematically investigated this bias, using gradient boosting techniques for model fitting. It was demonstrated that in bootstrap samples drawn with replacement in many instances models are selected that have much larger complexity compared to models selected in the original data. While it is already known that such an effect occurs when model complexity is selected by criteria such as AIC or BIC, we consistently found it here also when employing cross-validation.

In the present setting the goal was not to estimate model complexity per se, but to get good estimates of prediction error, based on the bootstrap .632+ technique. For the Brier score in binary response settings and prediction error curves in a censored time-to-event setting, the complexity selection bias was shown to result in a severe bias for .632+ estimates. As an alternative we investigated bootstrap sampling without replacement. This avoids replicated observations, which seem to be a source of the complexity bias, probably due to the change in dependency structure they induce. The proposed sampling scheme, where $0.632n$ observations are drawn without replacement, was seen to result in improved, reasonable estimates of the true Brier score and of prediction error curves in high-dimensional settings.

Bootstrap sampling without replacement is closely related to cross-validation, where the data is repeatedly split into k folds, and models are fitted using the observations from $k-1$ folds and evaluated on the remaining fold. However, while for cross-validation typical values for k are 5, 10, or n (corresponding to the jackknife), the approach proposed in the present paper roughly corresponds to $k = 3$. The choice of using samples with $0.632n$ observations was made such that these samples contain approximately the same number of unique observations, and therefore the same amount of information, as bootstrap samples drawn with replacement. This close correspondence was sought, because we still wanted to apply the .632+ correction, which was developed for sampling with replacement. While it was seen that this indeed resulted in good estimates of the true prediction error, for cross-validation with arbitrary values of k it is unclear how to correct for a potential bias of the resulting estimates.

We are aware that the theoretical basis for using the .632+ correction together with bootstrap samples of size $0.632n$, drawn without replacement, is weak. However, for the theoretically more sound samples of size $n/2$, which Efron (1983) suggests to be similar to the bootstrap with replacement, a larger bias was seen in preliminary experiments. So,

our main justification for the proposed approach is its good performance in the simulation study. Naturally, this implies that the performance for scenarios not addressed in the simulation design is unclear. This is e.g. the case for scenarios where the number of covariates is smaller than the number of observations.

Improved estimates could also be obtained, when model complexity was not selected in every single bootstrap sample, but a fixed level of complexity, selected in the original data, was used. This is very surprising, as one essential step of model building is omitted in the bootstrap samples. While it would be expected that this results in underestimation of prediction error (Simon et al., 2003), this was not seen to be the case here. One explanation might be, that not the actual list of selected covariates was transferred from the original data to the bootstrap samples, but only the level of complexity, therefore only correcting for the complexity bias, without passing on too much information. This approach is tempting, as it is computationally much less demanding compared to approaches where complexity selection has to performed in every single bootstrap sample. Nevertheless, we recommend using the approach based on bootstrap sampling without replacement, as it does not omit model building steps and therefore has a better theoretical foundation.

While we also investigated .632+ estimates of misclassification rate and of the area under the ROC curve, we did not report results on these. There, effects of the complexity selection bias could also be seen, but due to the coarseness of these error measures, correction by a modified bootstrap sampling scheme did not work as well as for estimates of the Brier score. In the present study we also did not explore how large the number of covariates has to be for the model complexity bias to appear, as we were mainly interested in high-dimensional settings, that arise e.g. in the analysis of microarray data. Furthermore, we restricted attention to models with linear predictors. The complexity selection bias may for example also be a problem for additive models which are estimated via

splines with a large number of basis functions. This requires further investigation.

Finally, it was demonstrated with in an application example microarray survival data, how .632+ prediction error curve estimates, based on bootstrap samples drawn without replacement, can also be used for judging the performance of procedures other than gradient boosting. There the complexity selection bias was also seen for the CoxPath and the PC-PCR procedure when sampling with replacement was employed. The similarity to the results from the simulation study leads us to expect, that also for the latter (and even more) procedures .632+ prediction error curve estimates based on sampling without replacement will be reasonable. Therefore they may present a useful addition to the statistician's toolbox.

Acknowledgements

We gratefully acknowledge support from Deutsche Forschungsgemeinschaft (DFG Forschergruppe FOR 534).

References

Bair, E. and Tibshirani, R. (2004). Semi-supervised methods to predict patient survival from gene expression. *PLoS Biology*, 2(4):0511–0522.

Bender, R., Augustin, T., and Blettner, M. (2005). Generating survival times to simulate Cox proportional hazards models. *Statistics in Medicine*, 24(11):1713–1723.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3.

Bühlmann, P. (2006). Boosting for high-dimensional linear models. *The Annals of Statistics*, 34(2):559–583.

Bühlmann, P. and Yu, B. (2003). Boosting with the L2 loss: Regression and classification. *Journal of the American Statistical Association*, 98:324–339.

Dupuy, A. and Simon, R. M. (2007). Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *Journal of the National Cancer Institute*, 99(2):147–157.

Efron, B. (1983). Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331.

Efron, B. and Tibshirani, R. (1997). Improvements on cross-validation: The .623+ boostrap method. *Journal of the American Statistical Association*, 92(438):548–560.

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

Fu, W. J., Carroll, R. J., and Wang, S. (2005). Estimating misclassification error with small samples via bootstrap cross-validation. *Bioinformatics*, 21(9):1979–1986.

Gerds, T. A. and Schumacher, M. (2006). Consistent estimation of the expected Brier score in general survival models with right-censored event times. *Biometrical Journal*, 48(6):1029–1040.

Gerds, T. A. and Schumacher, M. (2007). Efron-type measures of prediction error for survival analysis. *Biometrics*, 63(4):1283–1287.

Graf, E., Schmoor, C., Sauerbrei, W., and Schumacher, M. (1999). Assessment and comparison of prognostic classification schemes for survival data. *Statistics in Medicine*, 18:2529–2545.

Hothorn, T. and Bühlmann, P. (2006). Model-based boosting in high dimensions. *Bioinformatics*, 22(22):2828–2829.

Jiang, W. and Simon, R. (2007). A comparison of bootstrap methods and an adjusted bootstrap approach for estimating the prediction error in microarray classification. *Statistics in Medicine*, 26(29):5320–5334.

Li, H. and Gui, J. (2004). Partial Cox regression analysis for high-dimensional microarray gene expression data. *Bioinformatics*, 21 (Suppl. 1):i208–i215.

Liao, J. G. and Chin, K.-V. (2007). Logistic regression for disease classification using microarray dara: Model selection in a large p and small n case. *Bioinformatics*, 23(15):1945–1951.

Lusa, L., McShane, L., Radmacher, M. D., Shih, J. H., Wright, G. W., and Simon, R. (2006). Appropriateness of some resampling-based inference procedures for assessing performance of prognostic classifiers derived from microarray data. *Statistics in Medicine*, 26(5):1102–1113.

Molinaro, A. M., Simon, R., and Pfeiffer, R. M. (2005). Prediction error estimation: A comparison of resampling methods. *Bioinformatics*, 21(15):3301–3307.

Park, M. Y. and Hastie, T. (2007). L_1-regularization path algorithms for generalized linear models. *Journal of the Royal Statistical Society B*, 69(4):659–677.

Ridgeway, G. (1999). The state of boosting. *Computing Science and Statistics*, 31:172–181.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyna, R. D., Muller-Hermelink, H. K., Smeland, E. B., and Staudt, L. M. (2002). The use of molecular profiling to predict survival after chemotherapy for diffuse large-B-cell lymphoma. *The New England Journal of Medicine*, 346(25):1937–1946.

Schumacher, M., Binder, H., and Gerds, T. A. (2007). Assessment of survival prediction models based on microarray data. *Bioinformatics*, 23(14):1768–1774.

Segal, M. (2006). Microarray gene expression data with linked survival phenotypes: Diffuse large-B-cell lymphoma revisited. *Biostatistics*, 7(2):268–285.

Simon, R., Radmacher, M. D., Dobbin, K., and McShane, L. M. (2003). Pitfalls in the use of DNA microarray data for diagnostic and prognostic classification. *Journal of the National Cancer Institute*, 95(1):14–18.

Steck, H. and Jaakkola, T. (2003). Bias-corrected bootstrap and model uncertainty. In *Advances in Neural Information Processing Systems 16*.

Strobl, C., Boulesteix, A.-L., Zeileis, A., and Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8(25).

Zhu, J. X., McLachlan, G. J., Ben-Tovim Jones, L., and Wood, I. A. (2008). On selection biases with prediction rules formed from gene expression data. *Journal of Statistical Planning and Inference*, 138(2):374–368.