

Kehl, Victoria; Ulm, Kurt

**Working Paper**

## Responder Identification in Clinical Trials with Censored Data

Discussion Paper, No. 311

**Provided in Cooperation with:**

Collaborative Research Center (SFB) 386: Statistical Analysis of discrete structures - Applications in Biometrics and Econometrics, University of Munich (LMU)

*Suggested Citation:* Kehl, Victoria; Ulm, Kurt (2003) : Responder Identification in Clinical Trials with Censored Data, Discussion Paper, No. 311, Ludwig-Maximilians-Universität München, Sonderforschungsbereich 386 - Statistische Analyse diskreter Strukturen, München, <https://doi.org/10.5282/ubm/epub.1692>

This Version is available at:

<https://hdl.handle.net/10419/23869>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

# Responder Identification in Clinical Trials with Censored Data

Victoria Kehl, Kurt Ulm

Institut for Medical Statistics and Epidemiology  
Technical University – Munich  
Ismaningerstr. 22  
81675 Munich  
Germany  
Tel. +49-89-4140-4348  
Fax: +49-89-4140-4973  
E-mail: [victoria.kehl@imse.med.tu-muenchen.de](mailto:victoria.kehl@imse.med.tu-muenchen.de)

This research was supported by grant UL 94/11-1 of the German Research Community (DFG) and by DFG's Special Research Areas (SFB) 386, project B7.

## Summary

We present a newly developed technique for identification of positive and negative responders to a new treatment which was compared to a classical treatment (or placebo) in a randomized clinical trial. This bump-hunting-based method was developed for trials in which the two treatment arms do not differ in survival overall. It checks in a systematic manner if certain subgroups, described by predictive factors do show difference in survival due to the new treatment. Several versions of the method were discussed and compared in a simulation study. The best version of the responder identification method employs martingale residuals to a prognostic model as response in a stabilized through bootstrapping bump hunting procedure. On average it recognizes 90% of the time the correct positive responder group and 99% of the time the correct negative responder group.

Keywords: responder identification, bump hunting, predictive factors

## 1. Introduction

The term *response* up to now appears only in clinical trials, in which surrogate markers are used to describe the effect of a treatment when that effect is other than to prevent an event. For example in oncology, the desired effect of a treatment may be reduction of tumor size, whereas the outcome of interest (called event) may be death. Then a *responder* is a patient who experienced tumor reduction or complete remission and a *non-responder* is a patient whose tumor did not change or grew. Notice, that does not necessarily mean that responders lived longer. No definition of responder is currently available for trials in which the effect of treatment is to prevent an event (e.g. mortality).

The classical definition of *responder* is altered in this research in order to fit the more general clinical trial situation, in which the wished effect of a treatment is increasing the event-free period of the treated patient. We define ***positive responders*** to be patients under the new treatment, who benefit from it. Their benefit is manifested in the fact that their survival time is longer than that of patients with the same characteristics (predictive factors), randomized in the classical treatment group. We define ***negative responders*** to be patients under the new treatment who are harmed by it. Their survival time is shorter than that of a similar, described by predictive factors, group of patients under the classical treatment. Consequently, ***non-responders*** would be patients who are neither positive nor negative responders. Their survival time does not differ from similar patients under the classical treatment. We are interested in identifying responders – both positive and negative.

Let us concentrate on the following common clinical trial situation in which the ability of a new treatment to prevent an event is tested. Patients are randomized into two groups: one receiving the classical treatment (or placebo) and the other receiving the new treatment. Not rarely, the outcome of such trials shows no difference in the survival probabilities of the two treatment groups. But still, it could happen that certain subgroups of patients show improved survival under the new treatment, while others appear to suffer from it. Exactly this situation appeared in the European Myocardial Infarction Amiodarone Trial (EMIAT) [1], which inspired this research. For an

application of the proposed responder identification method on the EMIAT data set, please refer to [2].

Suppose the survival time of a patient in the new treatment group of a clinical trial is greater than the overall survival time. There can be three reasons for this phenomena: (i) chance – we cannot predict or account for occurrence by chance in a model, (ii) the patient has a prognosis better than the average, due to the specific *prognostic factors* that he enjoys, (iii) the new therapy is really working. Note that prognostic factors influence the survival time **independently** of treatment. One can account for prognostic factors, provided that they have been measured, by developing a prognostic model on the classical treatment (or placebo) group. We have chosen to use the well known Cox-PH model for that purpose. Notice, that the factors in this model would be prognostic in the real sense of the term only if they are found on a placebo arm. If the new treatment is tested against a classical treatment, the factors would be "prognostic" only with respect to the classical treatment and not in general. To avoid confusion, for the rest of this paper we will call both factor types prognostic.

Our goal in this research was to find a method for identifying patients with special reactions to the new treatment (those could be positive as well as negative), which are different from the whole patient population and cannot be explained by prognostic factors (iii). In such cases *predictive factors* are responsible for the difference in survival. Note, that a factor can have both prognostic and predictive power, if its prognostic value is different in the two treatment groups.

## 2. The Classical Approach

Up to now, the classical and the only structured approach for responder identification in clinical trials has been the Cox-PH model including interaction terms between the treatment and some or all of the covariates [3]. In our setting the following version of the Cox-PH model would be fitted on the entire data set:

$$\lambda(t, x_i, z_i) = \lambda_0(t) \cdot e^{\overbrace{\beta'_x \cdot x_i}^{\text{prognostic}} + \overbrace{\beta'_1 \cdot z_i \cdot \text{treat} + \beta'_2 \cdot z_i + \beta_T \cdot \text{treat}}^{\text{predictive}}}$$

where  $i$  is a patient identifier,  $i = 1, \dots, n$ ,  $x$  is a vector of *prognostic* factors,  $\beta_x$  is a vector of coefficients of the prognostic factors,  $z$  is a vector of *predictive* factors ( $z \subset x$  is possible),  $\beta_z$  is a vector of coefficients of the predictive factors (to avoid double appearance,  $\beta_z[i] = 0$  for  $z_i \subset x$ ),  $treat$  is the factor indicating treatment group (0 = classical, 1 = new),  $\beta_T$  is the coefficient of the treatment indicator,  $\beta_I$  is a vector of coefficients of the interaction terms.

If a certain predictive factor interaction term shows to be adding information to the model, this should be interpreted as follows. If the coefficients in the predictive part of the model are such, that the presence of factor  $z_i$  in the model *increases the hazard* of patients having that factor and taking the new treatment, we can say that  $z_i$  is a predictive factor and patients having this characteristic are ***negative responders*** of the new treatment. Naturally, if the coefficients in the predictive part of the model lead to *reduction of the hazard* in the presence of factor  $z_i$ , then  $z_i$  would be a predictive factor which defines the ***positive responder*** group.

The problem with this method is, that in order for it to recognize a combination of factors as predictive, this particular combination has to be present in the model as interaction. Even assuming that the interaction between the factors is linear, the order of the interaction term is unknown. If two predictive factors and factor treatment should show interaction, one needs to consider all possible interaction terms of up to third order, in order to give a chance of a covariate selection procedure to choose the right combination. The number of possible interaction terms to be considered grows rapidly as the number of factors grows. It is also known, that the power of stepwise variable selection procedures decreases as the number of variables (variable combinations) increases. For that reason, up to now in practice researchers have always used previous knowledge about the factors in the study in order to do such subgroup analysis (see [4] & [5] for examples).

Considering the limitations of this approach, it is clear, that a new more involved exhaustive method is needed.

### 3. The Responder Identification Method for Censored Data

We suggest the following strategy for positive and negative responder identification:

#### **Algorithm 1**

##### ***Responder Identification for Censored Data:***

1. Develop a good prognostic model (e.g. Cox-PH model) on the classical treatment arm of the data.
2. Apply the prognostic model together with its estimated coefficients and baseline hazard to the new treatment group.
3. Calculate residuals of the prognostic model for outlier identification in the new treatment group. Patients who are not well predicted (outliers in the residuals) would be candidates for responders.
4. Using the residuals as a response, develop a model on the new treatment group which describes extreme regions of the residuals.
5. Identify the groups of patients in the classical treatment group, who correspond to the groups with extreme residuals in the new treatment arm, i.e. divide the classical treatment space in the same way as the new treatment space and consider the regions which were identified as extreme in the new treatment space.
6. Compare the survival curves of each classical-new treatment pair of extreme regions (log rank test). **If there is a significant difference in survival, the group with extreme positive residuals would identify negative responders and the extreme negative residuals – positive responders.** Also, the factors involved in the description of the regions will be predictive.

The best version of the responder identification method (as of our simulation study) is shown schematically in the flow-chart of figure 1.

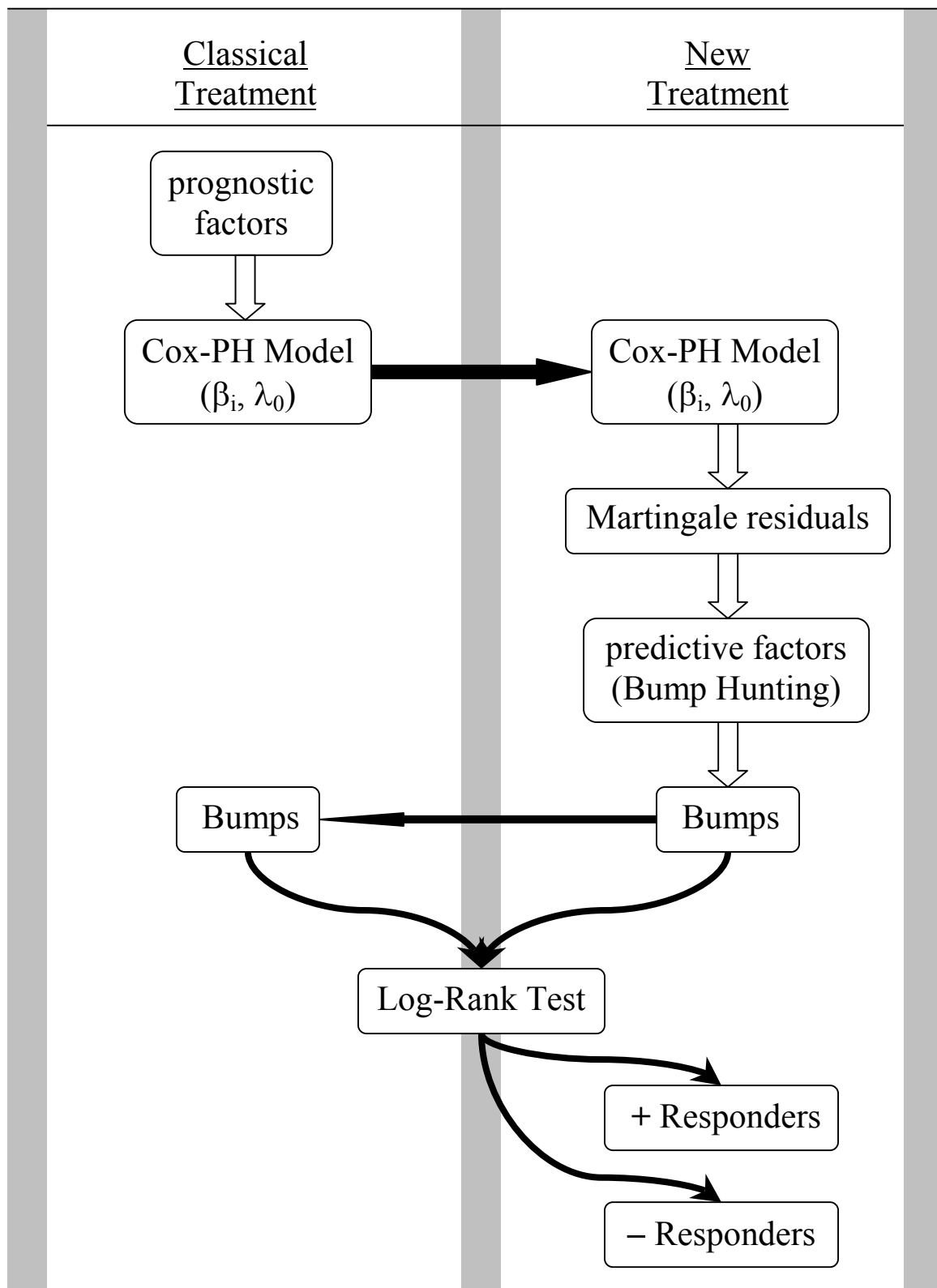


Figure 1. Flow diagram of the responder identification algorithm.



## 4. Options in the Responder Identification Method

### 4.1 The Residuals

In step 3 of the responder identification method we need residuals to the Cox-PH model which correspond to data points and are not explicitly connected to single prognostic factors contained in the model. Such residuals would be able to identify outlying points with poorly predicted individual outcomes by the prognostic model. Those points can be used for predictive factor identification and, ultimately, responder identification purposes. We suggest the use of martingale or deviance residuals despite their bad distributional properties because of their interpretability and since no special distributional properties are required in the responder identification method. The log-odds and normal deviate residuals suggested by Nardi & Schemper [6] can be used as well if interpretability in the form "expected – predicted" is of no interest.

Specifically for the Cox-PH model, the definition of martingale residuals reduces to:

$$\hat{M}_i = \delta_i - \hat{\Lambda}_0(t_i) \cdot e^{\hat{\beta} \cdot Z_i} = \delta_i - \hat{\Lambda}_i(t_i, Z_i),$$

where  $t_i$  is the observation time and  $\delta_i$  is the final status for subject  $i$  [7]. Notice that since the status can take only values of 0 or 1 and the hazard is always non-negative, the martingale residual for the Cox-PH model takes values only in the interval  $(-\infty, 1]$ .

The deviance residual [8], is the signed square root of the deviance. The  $\ln(\cdot)$  function inflates martingale residuals close to 1 and the square root contracts the large negative values. It is zero if and only if  $\hat{M}_i = 0$ . For the Cox-PH model, the deviance residual simplifies to:

$$d_i = \text{sgn}(\hat{M}_i) \cdot \sqrt{-2 \cdot [\hat{M}_i + \delta_i \cdot \ln(\delta_i - \hat{M}_i)]}.$$

For residual interpretation purposes the censoring indicator can be thought of as a classification rule, which places patients into either the low or the high hazard group. This results in only a few possible scenarios for the residuals:

1. As by most other residuals, values of the martingale and deviance residuals around zero reflect good fit of the model. In our situation this can be achieved if  $\delta_i = 1$  and  $\hat{\Lambda}_i \approx 1$ , which means that the  $i^{\text{th}}$  patient with an event was predicted to be at high risk, or if  $\delta_i = 0$  and  $\hat{\Lambda}_i \approx 0$ , which means that the  $i^{\text{th}}$  patient was censored and predicted to be at low risk. Those are candidates for ***non-responders***.
2. Large values of any residuals are a sign of bad fit of the prognostic model and here – a possible sign of existing predictive factors. Values of the martingale residuals close to 1 can be achieved only if  $\delta_i = 1$  and  $\hat{\Lambda}_i \approx 0$ , i.e. the  $i^{\text{th}}$  patient was predicted to be at low risk but he/she had an event. Such patients are candidates for ***negative responders***. Their deviance residuals will have values even larger than those of their martingale residuals (see figure 2).
3. Large negative values are also sign of a bad fit. Large negative values of the residuals are achieved if  $\delta_i = 0$  and  $\hat{\Lambda}_i > 0$ , i.e. the  $i^{\text{th}}$  patient was predicted to be at high risk but he/she was censored (i.e. did better than expected from the prognostic model). Such patients are candidates for ***positive responders***. A large negative residual is also possible for patients who had an event and extremely large predicted hazard rate. Notice, that even though the patient experiences an event, he/she would still be candidate for a positive responder, since in order to have such a large hazard rate, he/she must have had a much longer event-free period than expected.

The general relationship between size of the predicted hazard and resulting residual (martingale and deviance) is plotted in figure 2 for censored and uncensored cases, which is helpful in illustrating the above cases.

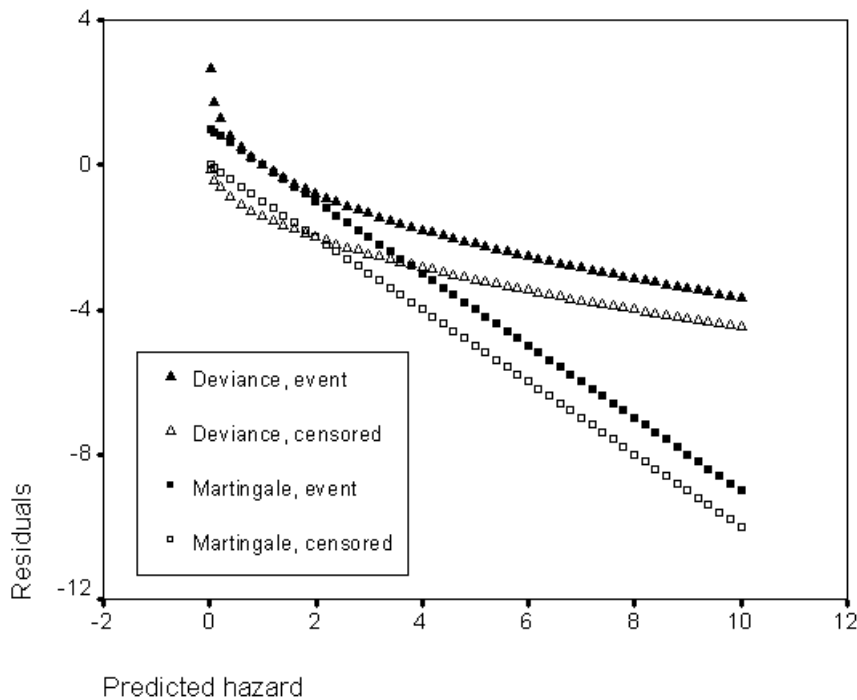


Figure 2: *Relationship between the predicted hazard and the martingale and deviance residuals for the event and censored (always non-positive) cases. The relationship between predicted hazard and martingale residual is linear, whereas deviance residuals transform that relationship.*

## 4.2 The Predictive Model

We suggest the use of a bump hunting model in step 4 of the responder identification method. Regression trees can be used alternatively as a predictive model.

### 4.2.1 Regression Trees

We start with the better known regression tree model [9]. As mentioned earlier, the responder identification idea is based on finding patients in the new treatment group, who are badly predicted by the prognostic model. We have already shown that martingale and deviance residuals are suitable for this purpose. Since regression tree models split the input space into regions, which are described by a part of the input

variables (the predictive factors), and the size of the output variable in each region is predicted, we can use regression trees for responder identification. The tree would be built on the new treatment arm of the data and the residuals to the prognostic model would be used as an output variable. The hope is that one or more of the final regions of the tree model would have much larger or much smaller mean of the residuals in them, than the average for the input space ( $\approx 0$ ).

We have used the regression tree model as introduced by Breiman et al [9] and as implemented in S-plus 4.5 for Windows, where the optimal cutpoints are chosen by minimizing the sum of the averages of the response variable in the resulting regions over all available splitting variables  $x_j$  and splitting points  $t_s$ .

When using a regression tree instead of a bump model, one should change the responder identification algorithm as follows:

## **Algorithm 2**

### ***Responder Identification with Regression Trees:***

Steps 1 through 3 as in algorithm 1 above.

4. Develop a regression tree model on the new treatment group, using the martingale residuals of the prognostic model as response.

*Note: A tree model describes the whole input space. We are interested only in extreme regions, i.e. end nodes with patients who have large positive or large negative residuals.*

5. Order all end nodes by size of the mean response in them.
6. Split the classical treatment group into subgroups as defined by the end nodes of the regression tree model.
7. Start with the largest in absolute value (by mean of response) negative end node in the new treatment arm. Compare the survival curves of the new and classical treatment patients identified with that node. Calculate the p-value of the log-rank statistic ( $p(\text{LR})$ ).
8. Add all patients contained in the next largest negative end node (from the list in 5.) to the previously considered group of new treatment patients. Calculate  $p(\text{LR})$  for the identified groups in the new and classical treatment arms.
9. Repeat step 8 while  $p(\text{LR})$  decreases or until there are no more negative end nodes<sup>1</sup>.
10. The last combination of end nodes defines the set of positive responders. The factors involved in defining it are predictive.
11. Repeat steps 7 through 9 for the non-negative end nodes from the list in 5., starting with the largest by mean of response.
12. The last combination of non-negative end nodes defines negative responders. The factors involved in defining it are predictive.

---

<sup>1</sup> For data which shows no initial survival difference between the two treatment arms.

#### 4.2.2 The Bump Hunting Model

Although not originally created for responder identification, bump hunting as proposed by Friedman & Fisher [10] seems to be tailor made for that purpose. Since this method is not as well known as regression trees, we will briefly summarize its algorithm. Bump hunting is a type of greedy optimization algorithm equipped with patience, which stresses interpretability of the resulting model. It optimizes the average of the output variable while choosing a series of input variables and corresponding cutpoints in the following way.

During the model construction process, just as in regression trees, bump hunting looks for rectangular regions called boxes, but not by minimizing the sum of the averages of the (two) new regions into which the current space is split. Bump hunting "peels off" a certain percentage of the data while optimizing the response average of the elements left in the box. At each peeling step, a variable and a peel-off value is chosen, which together define a border so that the data points left in the region have the largest mean of the output variable:  $\max_{x_j, t_j} \bar{y}_B$ , where  $B$  is the box resulting from a peeling at variable  $x_j$  and peeling point  $t_j$  (if a minimal region is sought, one can maximize the negative average). The **top-down-peeling** process stops when a minimum number of elements in the box is reached. Since peeling is a greedy process, the average of the response variable in the box can often be improved by "pasting" back some of the data to the box. The **bottom-up-pasting** process stops when the average in the box can no longer be improved. In general, when pasting is possible, the new box does have a larger mean of the response variable, but that rarely has a dramatic effect [10]. Two parameters need to be specified in the box construction algorithm: **peeling quantile  $\alpha$**  and **minimal support  $\beta_0$** . The peeling quantile determines the percentage of data points excluded (peeled away) from the current box at each peeling step. Friedman and Fisher [10] suggest values of  $\alpha$  between 0.05 and 0.1, which results in the removal of 5% to 10% of the data at each step. The minimal support is a threshold parameter, which determines the minimal size of the final box. The choice of the minimal box support involves statistical and domain of application dependent considerations. The development of a box mean (i.e. mean of the target variable for data points in the box) with respect to support  $\beta$  can be observed with the help of the box construction **trajectory**. The trajectory allows one

to visually choose an optimal  $\beta_0$ . Figure 3 shows an example of a trajectory, constructed with  $\alpha = 0.1$  where the mean of the response variable is maximized. One can observe how the mean grows from the mean of the whole data set (0) to about 0.75. The points on the trajectory represent the consecutively chosen borders. The trade off between support and mean in the growing box is clearly visible. Notice, maximization is done only in the direction of mean response. Multivariate optimization which includes both mean of response and support is not performed in the original bump hunting algorithm.

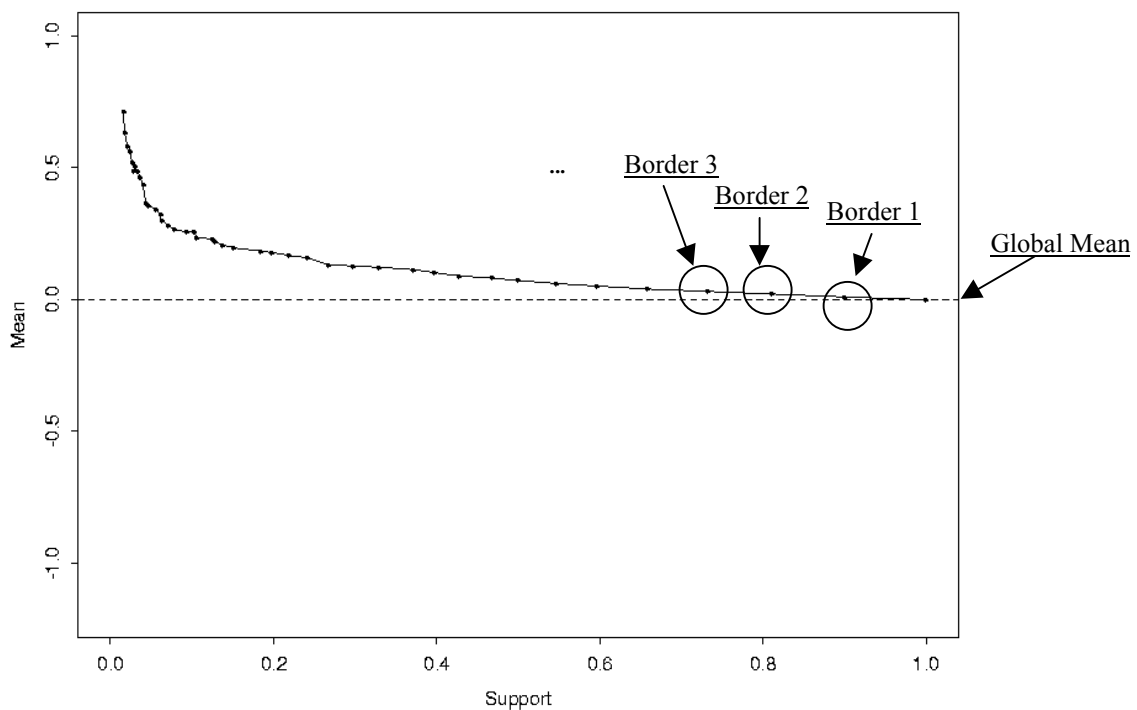


Figure 3: Trajectory – visualization of the box-building process ( $\alpha = 0.1$ ).

If the input factors are *categorical*, that changes the peeling procedure in the following manner: (i) binary variables allow only one peeling point, which splits the data into two parts, (ii) variables with more than two categories are treated the same way as in CART: peeling points are defined in such manner, that any category can be peeled off regardless of order, (iii) continuous variables are often categorized. In this case, to preserve the order among categories, we suggest artificially entering the categorized input variable as "continuous" in the bump hunting algorithm.

### 4.2.3 Stabilized Bump Hunting

Bump models are rather unstable, due to their hierarchical nature. There are two general ways of assuring that a model is "good": validation and stabilization. Harrell, Lee, & Mark [11] summarize the procedures for performing external validation and three types of internal validation: data splitting, cross-validation, and bootstrapping. The external and the first two internal validation methods require abundance of appropriate data, which one rarely has. Bootstrapping uses the entire data set in the model building process and then calculates some goodness-of-fit statistic on a large number of bootstrap samples taken from the original data. There is no known goodness-of-fit statistic for the bump model, therefore, bootstrapping as a validation procedure is not directly applicable for bump models. The validation choices left are external validation or internal validation involving data splitting. Both are not always possible.

If there is no direct way to validate a bump model, one should at least reduce the variability of the bump hunting model resulting from small changes in the data – the so called *stabilizing*.

Bootstrapping can be used as a stabilization procedure during the model building process (see [12] & [13]). One can use bootstrap samples of the original data in order to estimate the model coefficients or to choose stronger predictors. Bootstrapping, in all of its shapes and forms improves or qualifies the predictive capacity of the model.

We stabilize the bump hunting model in two ways: (i) by categorization of all continuous predictors and (ii) by bootstrapping at each border selection step. Categorization of all continuous predictors needs to be done in order to reduce peel-off point variation. This limits somewhat the power of bump hunting, since it restricts the peeling process, but it is a necessary preliminary step for the bootstrapped bump hunting. In order to stabilize the border selection method, we choose each border (i.e. predictor-restriction combination) after considering all borders chosen from  $n$  bootstrap samples. We fix a border and proceed to the next one only if it was chosen in the majority of the bootstrap samples.



Bootstrapping in (ii) is performed according to algorithm 3. Note, that predictor and border are not equivalent terms. One predictor may appear with different restrictions in different bootstrap samples. We are only interested in the border frequency as a combination of predictor and constraint.

Naturally, any stopping criteria which considers only one step at a time is easily implemented, but in general nearsighted. An alternative is to look several steps ahead before a stopping decision is made, since a seemingly "bad" border can lead to a "good" one and result in a better model. We choose not to do this in the following simulation study in order to fully automate the software implementation and reduce computation time.

### **Algorithm 3**

#### ***Stabilized Bump Hunting Algorithm***

1. Set the  $p$ -value of the log-rank statistic to 1 ( $p(LR) = 1$ ) and let  $T$  be the set of all patients in the new treatment arm.
2. Take  $n$  bootstrap samples of  $T$  and, using the original bump hunting algorithm, create a trajectory for each one of them, including the original sample.
3. Consider all  $n + 1$  first borders and the associated predictors and choose the one which appears most often. If there is a tie, choose the less restrictive border, i.e. one which results in a box with bigger support when applied to the original data set.
4. Restrict  $T$  using the border from step 3. Calculate the mean response and the support of the resulting box.
5. Apply the rules restricting  $T$  to the classical treatment (or placebo) group and create a set  $P$  of patients under the same restrictions as in  $T$ .
6. Calculate the  $p$ -value of the log-rank statistic ( $p(LR)$ ) for the difference in survival between patients in  $P$  and in  $T$ . If  $p(LR)$  improves<sup>2</sup> from its previous value, return to step 2. If not, stop.

## **5. Simulation Study**

If there were a data set, in which the positive and negative responder groups were known, one could apply the different versions of the algorithm discussed up to now and compare their ability to recognize those groups. Unfortunately, this is not possible in a real life data set, since the actual groups to be identified are not known.

---

<sup>2</sup> The definition of "improves" can be different for different types of data. If initially there is no difference in survival between the new and the classical treatment groups, the  $p$ -value improves when it decreases.

We simulated a survival type data set to resemble a two arm randomized clinical trial with a total of 1000 patients, with no difference in survival between the two treatment groups. A total of seven factors were created: five binomial, one categorical with three levels, and one continuous in order to test the power of the different procedures in dealing with different types of variables. The factors (each was a vector of length 1000) were simulated in the following way. Each value of the *binary factors*  $X1$ ,  $X4$ ,  $X5$ ,  $X6$ , and  $TREAT$  was chosen at random from a binomial distribution with probability  $p = .5$ .  $TREAT = 0$  denotes placebo patients,  $TREAT = 1$  denotes new treatment patients. The *categorical factor*  $X2$  values were chosen at random from the set  $\{0, 1, 2\}$  with corresponding probabilities  $\{.33, .33, .34\}$ . The *continuous factor*  $X3$  had values chosen at random from a normal distribution with mean five and variance two. Follow-up time for this model was simulated to be Weibull distributed with shape parameter equal to two and scale parameter equal to the relative hazard, i.e.  $TIME$  was created to be a vector of length 1000, each component of which was chosen at random from the unique to each patient Weibull distribution, depending on his/her relative hazard. We used the following Cox-PH model with prognostic and predictive parts for defining the hazard:

$$\lambda(t | X) = \lambda_0(t) \cdot e^{\text{prognostic} + \text{predictive}}$$

$$\text{prognostic} = \beta_1 \cdot X1 + \beta_2 \cdot X1 \cdot X3 + \beta_3 \cdot X3$$

$$\text{predictive} = c_{\min} \cdot X4 \cdot X5 \cdot X6 \cdot TREAT + c_{\max} \cdot X22 \cdot X5 \cdot TREAT,$$

where  $c_{\min}$  and  $c_{\max}$  are coefficients in the predictive part,  $\beta_1$ ,  $\beta_2$ , and  $\beta_3$  are coefficients in the prognostic part, and  $X22$  indicates  $X2 = 2$ .

The prognostic coefficient values  $\beta_1 = \ln 3$ ,  $\beta_2 = -(\ln 3)/5$ , and  $\beta_3 = (\ln 3)/10$  ( $\approx 1.0986$ ,  $-.2197$ , and  $.1099$  respectively) simulate an interaction between categorical factor  $X1$  and continuous factor  $X3$ . The interaction is to be interpreted as follows: in absence of factor  $X1$  ( $X1 = 0$ ), increase of factor  $X3$  from 0 to 10 increases the relative hazard from 1 to 3; in presence of factor  $X1$  ( $X1 = 1$ ), increase of factor  $X3$  from 0 to 10 decreases the relative hazard from 3 to 1.

The following pairs of values for  $c_{\min}$  and  $c_{\max}$  were chosen for further investigation: (-2, 2), (-1, 1), and (-.5, .25). Note, that larger absolute values of the coefficients simulate stronger influence of the predictive part of the model on hazard. In addition, since  $c_{\min}$  is always negative, it decreases hazard, i.e. patients with  $X4 = 1$  &  $X5 = 1$  &  $X6 = 1$  would have lower hazard under treatment than under placebo – **this is the simulated positive responder group**. Conversely, since  $c_{\max}$  is always positive, it increases hazard, i.e., patients with  $X2 = 2$  &  $X5 = 1$  would have higher hazard under treatment than under placebo – **this is the simulated negative responder group**.

Censoring was assigned at random and is independent of time. Three different percentages of censoring were considered: 10%, 30%, and 70%.

The above described survival data was simulated in 9 groups with different model coefficients and censoring rates as shown in table 1. The survival curves of the placebo (TREAT = 0) and treatment (TREAT = 1) groups were compared in each simulation group. The overall survival difference between the two treatment arms was not significant at the .05 level in each of the 9 groups, as it was expected by the simulation study design. The simulated responder groups, however, differ in survival between the treatment and placebo groups (as expected).

Table 1: *Simulation groups*

Simulation group #	$c_{\min}$	$c_{\max}$	% censored
1	-2	2	10
2	-1	1	10
3	-.5	.25	10
4	-2	2	30
5	-1	1	30
6	-.5	.25	30
7	-2	2	70
8	-1	1	70
9	-.5	.25	70


## 5.1 The Cox-PH model with treatment interaction term

Simulation groups 1, 5, & 9 were chosen as representative in an attempt to evaluate the power of the most frequently used variable selection process, forward stepwise selection, to identify the simulated Cox-PH model with interactions as "best." Forward selection with likelihood ratio test as model improvement criteria was used with inclusion  $p(\text{Wald}) = .01$  and exclusion  $p(\text{Wald}) = .05$ . Table 2 gives a summary of this investigation. Simulation group 1 has strong simulated treatment effect (easy to detect) and only 10% censoring. Simulation group 5 has medium strength simulated treatment effect and 30% censoring. Simulation group 9 has 70% censoring and slight treatment effect (difficult to detect). A total of 10 data sets were simulated in each group. The null model (no factors) and the correct model likelihood ratios were computed on each data set. Forward stepwise selection was applied on each data set four times: once including all factors  $X_1$  through  $X_6$  and TREAT and all their possible two-way interactions (a total of 28 terms to choose from), once including all single factors and all their up to third order interactions (63 terms), all single factors and all their up to fourth order interactions (98 terms), and finally, all single factors and their up to fifth order interactions (119 terms). The largest interaction term in the correct model is of fourth order. Interactions of up to fifth order were considered in order to check if forward selection including interaction terms of higher than needed order would choose more complicated terms than necessary. This should give a hint on the behavior of the automated selection procedure in a "real life" data set, for which the correct model is unknown.

Table 2: Table of likelihood ratios for the null model, the models found with forward selection when different highest order interactions were present, and the correct model.

Simulation group	Run	Null (df=0)	2 <sup>d</sup> order interaction (df)	3 <sup>d</sup> order interaction (df)	4 <sup>th</sup> order interaction (df)	5 <sup>th</sup> order interaction (df)	Correct (df=6)
1	1	11239	11076 (11)	10976 (8)	10942 (6)	10942 (6)	10942
	2	11110	10941 (9)	10860 (5)	10844 (4)	10844 (4)	10820
	3	11121	10986 (10)	10884 (9)	10860 (5)	10860 (5)	10814
	4	11161	10946 (9)	10893 (6)	10868 (4)	10857 (6)	10822
	5	11094	10966 (9)	10787 (11)	10803 (5)	10803 (5)	10764
	6	11093	10929 (10)	10861 (5)	10819 (5)	10804 (8)	10784
	7	11132	10914 (13)	10801 (12)	<b>10736 (10)</b>	<b>10736 (10)</b>	<b>10761</b>
	8	11103	10919 (11)	10857 (5)	10819 (5)	10819 (5)	10803
	9	11109	10998 (6)	10878 (7)	10828 (7)	10828 (7)	10805
	10	11189	11015 (11)	10912 (12)	<b>10894 (8)</b>	<b>10894 (8)</b>	<b>10903</b>
5	1	8651	8608 (3)	8593 (5)	8581 (5)	8581 (5)	8555
	2	8705	8677 (3)	8633 (6)	8636 (5)	8636 (5)	8632
	3	8441	8422 (2)	8394 (5)	8395 (3)	8382 (5)	8366
	4	8525	8491 (5)	8476 (3)	8472 (3)	8472 (3)	8447
	5	8407	8381 (3)	<b>8321 (9)</b>	8329 (7)	8329 (7)	<b>8324</b>
	6	8412	8387 (3)	8365 (5)	8363 (4)	8363 (4)	8343
	7	8444	8410 (3)	8401 (3)	8384 (4)	8384 (4)	8368
	8	8584	8529 (4)	8522 (3)	8503 (3)	8503 (3)	8472
	9	8428	8397 (3)	8342 (4)	8335 (4)	8335 (4)	8316
	10	8586	8547 (3)	8495 (3)	8486 (3)	8486 (3)	8453
9	1	2740		<b>2722 (3)</b>	<b>2722 (3)</b>	<b>2722 (3)</b>	<b>2724</b>
	2	2474					2455
	3	2614					2600
	4	2643		<b>2627 (2)</b>	<b>2627 (2)</b>	<b>2627 (2)</b>	<b>2631</b>
	5	2845	2829 (3)	2829 (3)	2829 (3)	2829 (3)	2823
	6	2956					2945
	7	2509					2489
	8	2738	2729 (1)	2729 (1)	2729 (1)	2729 (1)	2724
	9	2751					2745
	10	2291					2279

**bold** = models with better LR than the corresponding correct model

 = identical models (valid for the row)

## 5.2 The proposed responder identification method

The three versions of the responder identification method were applied on 200 data sets in each of the nine simulation groups from table 1. Martingale residuals were calculated on half of the data sets in each group. Deviance residuals were calculated on the other

half. Our goals were: (1) to evaluate the prognostic power of the responder identification procedure with regression trees, ordinary and stabilized bump hunting, by comparing the identified through the method groups of responders to the correct groups and (2) to compare the power of identification of the method when martingale and deviance residuals are used as response variables in the predictive models. Step 1 of the responder identification algorithm can be skipped in the simulation study. The prognostic model here is known. It was simulated to contain factors  $X_1$ ,  $X_3$ , and their linear interaction. Table 3 shows the average number of times the entire responder sets were chosen correctly over the 100 data sets in each simulation group, each residual type, and each predictive model.

Table 3: *Number of times the correct responder groups were chosen from 100 simulations for each simulation group, using martingale (MART) or deviance (DEVI) residuals as response in a regression tree (tree), bump hunting (original), or stabilized bump hunting (stabilized) predictive model. MIN denotes search for a minimal region in the residuals (i.e. positive responders); MAX – maximal region (i.e. negative responders).*

sim. group #	MART						DEVI					
	tree		original		stabilized		tree		original		stabilized	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
1	98	0	71	95	97	99	100	65	42	100	72	100
2	73	23	65	98	88	100	94	71	33	100	52	100
3	22	10	47	94	84	99	52	10	26	100	46	100
4	94	0	72	98	99	99	94	81	39	100	78	100
5	76	41	67	97	92	99	76	70	35	99	59	100
6	28	6	52	97	79	100	33	6	22	100	36	100
7	66	60	75	96	97	99	59	52	43	100	78	100
8	51	53	67	94	89	99	53	32	31	100	48	100
9	21	4	56	94	83	98	10	1	26	99	35	100

It turned out, that in the simulated data the p-value of the log-rank statistic never grew insignificant in the stabilized and original bump hunting. We developed the bump hunting models using minimal support of .05 as stopping criteria. Since we knew the correct responder and non-responder groups, we considered just one box per bump and just the first three selected borders. The developed model was considered correct if the first three borders of the maximal box were any permutation of the following:  $X_2 \neq 0$ ,  $X_2 \neq 1$ ,  $X_5 \neq 0$  and the minimal box – any permutation of the following borders:  $X_4 \neq 0$ ,

$X_5 \neq 0$ ,  $X_6 \neq 0$ . Naturally, the correct bumps are usually not known. One does not even know how many boxes each bump has. Fortunately, "real life" data sets are not as clean and ordered as our simulated data, so that the bump growth process can actually be governed by the log-rank statistic, as in the suggested algorithm (see [2]).

## 5.3 Results

### 5.3.1 *Cox-PH model with treatment interaction*

In defense of the forward stepwise selection procedure, one should note, that in most cases (21 out of 24 constructed models) it did not add a fifth order interaction term, but delivered the model chosen from the procedure including up to fourth order interactions (see table 2). Unfortunately, it also chose the correct model only once out of 30 times (run 1, simulation group 1)!

Consider first simulation group 9, the most realistic one. In 6 out of 10 cases the forward selection procedure did not find any significant factors. In the 4 data sets, in which significant factors were found, they were other than the simulated ones (i.e. noise). In simulation groups 1 and 5 (see table 2), the likelihood ratio of the correct model was better than that of the forward selected models for 18 out of 20 data sets. In the cases where  $LR(\text{forward}) < LR(\text{correct})$  the forward selected model contained the correct model and some additional factors. The overall impression is that the Cox-PH model with interactions is not a sensitive enough method for responder identification purposes when the effect of factors and factor combinations on treatment is weak and there is large percentage of censoring in the data. It performs well on data sets with small to moderate percent censoring and strong to moderate treatment effect. The problem with applying this responder identification procedure in praxis is that the correct model is unknown and the forward selection procedure has low power when choosing from a large number of terms.

### 5.3.2 *Algorithm 2 with a regression tree predictive model*

Deviance residuals in general performed about the same or better than martingale residuals in positive responder identification (i.e. negative nodes). The same was true for negative responders (positive nodes), except for data sets with large percentage of



censoring. Overall, the responder identification algorithm using regression trees showed acceptable power of identification for data, in which the groups to be identified were with much larger (or much smaller) hazard than the entire data set (sim. groups 1, 4, & 7 with large predictive coefficients). The results were miserable for the data in sim. groups 3, 6, & 9, where the predictive coefficients were very small. This leads us to the conclusion, that **regression trees are not sensitive enough to be applied in responder identification. Nevertheless, if we had to make a recommendation which residuals to be used as a response factor in CART, we would prefer deviance residuals, as they have acceptable performance at least for the case when censoring is not too large and the responder coefficients are strong (sim. groups 1, 2, 4, & 5). For data with large percentage censoring it is preferable to use martingale residuals.**

### *5.3.3 Algorithms 1 & 3 with bump hunting predictive models*

The results with bump hunting seem to be independent of percent censoring: simulation groups (1, 4, 7), (2, 5, 8), and (3, 6, 9) have similar outcomes across the different methods. Size of the  $c_{\min}$  and  $c_{\max}$  coefficients show effect: larger in absolute value coefficients result in better performance of the different methods. Martingale residuals show to be better suited for positive responder identification (minimal bump) than deviance residuals. For negative responder identification (maximal bump) deviance residuals perform just as well or slightly better than martingale residuals. In all cases where improvement was possible, the stabilized bump hunting algorithm showed much better results than the original algorithm.

Deviance residuals perform excellent in negative responder identification and unsatisfactory in positive responder identification. Their use is not recommended when both responder groups are needed. **The stabilized bump hunting procedure with martingale residuals as response variable delivers excellent results both in positive and negative responder identification, especially if the effect is strong.**

## 5.4 Comparison

We did not formally include the classical Cox-PH model with treatment interactions in this comparison, because simulations with this model were performed only on groups 1,

5, & 9 and only on 10 data sets in each group. The correct model was found on only one data set in the “easiest” simulation group 1. This results in 10%, 0%, & 0% recognition rate for groups 1, 5, & 9 respectively. Then the Cox-PH model with treatment interactions is only better than the regression tree and mantingale residuals version of the responder identification algorithm.

Figure 4 gives a summary of the results of the responder identification algorithm when the regression trees, original, and stabilized bump hunting is employed (see table 3). Note, that comparison between a tree and a bump model can only be made in a very loose sense, since tree models describe the entire space and bump models – just extreme parts of it. In 16 out of the 18 cases with martingale residuals as response, bump hunting was more powerful than regression tree as predictive model in the responder identification algorithm. In the two other cases the results of both models were comparable; regression tree performed slightly better than bump hunting. When deviance residuals were used as a response variable, all three versions of the responder identification method performed unsatisfactory in finding positive responders, especially on data sets with high percentage of censoring. Finding negative responders with deviance residuals and bump hunting (both original and stabilized) was extremely successful. Regression trees performed unsatisfactory in a combination with deviance residuals in the responder identification method when negative responders were sought. **It is, therefore, recommendable that the responder identification method be used with martingale residuals as a response variable in the stabilized bump hunting.**

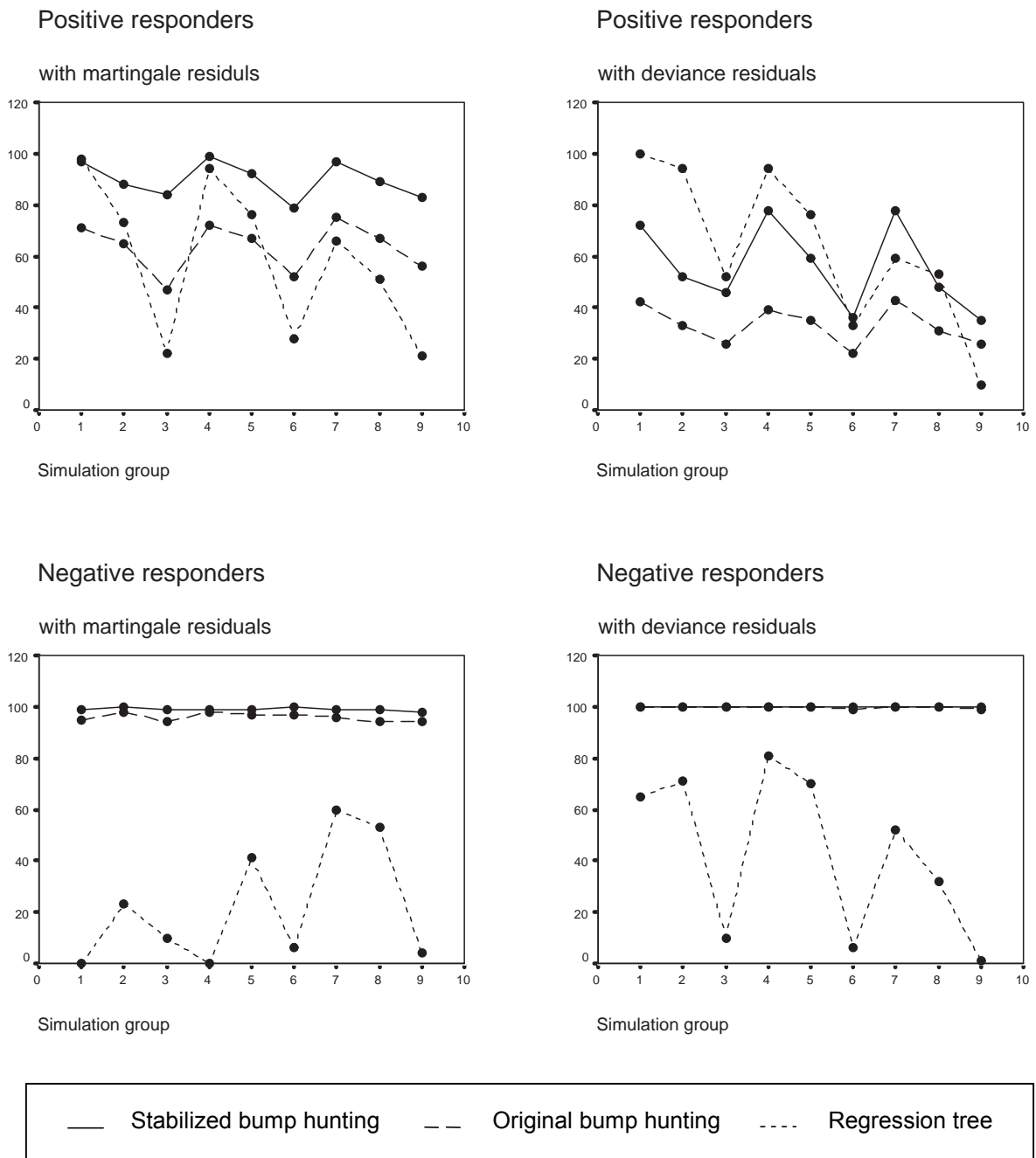


Figure 4: *Number of correctly identified responder groups from 100 simulations in each simulation group.*

## 5.5 Implementation

This simulation study was performed with the help of the readily available statistical packages SPSS and S-PLUS and the programming languages S and C. The Cox model with interactions and the Kaplan-Meier curves were generated using the survival analysis tools in SPSS 10.0. The simulation of all data sets, as well as the bump hunting analysis were done with especially written for the purpose S programs, which run with S-PLUS 4.5. Construction of the bump models was done in S-PLUS for Unix, using the algorithms of Becker [14] called `.boxes`, `.express.boxes`, and `.border.ranking`, which use C subroutines. The part of the simulation study involving regression tree models was done in S-PLUS for Windows. S routines using the S-PLUS tools for regression tree construction were written for that purpose.

## 6. Discussion

The responder identification method was developed with a situation in mind, in which overall the new treatment does not show to be better or worse than the classical treatment (i.e. the survival curves in both treatment arms do not differ significantly). The method needs slight alteration if initial difference in survival is present. Ordinarily, one would use the change in p-value of the log-rank statistic as a stopping criteria in the bump hunting procedure (Algorithm 3, point 6). In the peeling process of bump hunting, one would reduce the new treatment patient arm step by step. If there is a significant difference for the entire population, reducing the group would lead to less and less significant p-values before it eventually reduces the new treatment group to this one special subgroup, for which the p-values become significant again to show difference in survival between the new and the old treatment subgroups in the opposite direction from the initial situation. In those cases, the p-value of the log-rank statistic cannot be used as an automatic stopping criteria and the growth of the bump hunting model needs to be controlled manually. Alternatively, the algorithm can be changed to "intelligently" evaluate the p-values by looking some steps ahead.

As further improvement to the responder identification method for censored data, one may choose to allow for pasting in the bump model growth or looking several steps ahead at the log-rank statistic performance in the predictive model growth. As discussed

earlier, this improves the shortsightedness of a stopping criteria, but unfortunately also complicates the automation of algorithms.

The best version of the responder identification algorithm (algorithm 3) can be modified to include pasting as well. In this case, we would use the minimal support (calculated in the original data set) as stopping criteria of the peeling process instead of the  $p$ -value of the log rank statistic. Pasting borders would also be chosen through bootstrapping. Here we can use both the  $p$ -value of the log rank statistic and the indicator for increase (decrease) of the box mean as stopping parameter.

We limit the stabilized bump hunting procedure to categorical or categorized continuous factors, but we are currently working on changes allowing continuous factors as well.

So far we have only considered the ordinary Cox-PH model as a prognostic model in our algorithms. One may choose to investigate time-varying effects of the prognostic factors before looking for predictive factors. Additional research is needed on the influence the choice of prognostic model exerts on the predictive model outcome.

*A note on application:* We are currently using the best version of the proposed responder identification method as a routine on all appropriate clinical data we have. First results can be seen in [2].

## Bibliography

1. Julian, D.G. et al. Randomised trial of effect of Amiodarone on mortality in patients with left-ventricular dysfunction after recent myocardial infarction: EMIAT. *The Lancet* (1997); **349**: 667-674.
2. Kehl, V. Responder identification in clinical trials. Dissertation, Institute for Statistics, Ludwig-Maximilian University, Munich. Online publication (2002): [http://edoc.ub.uni-muenchen.de/archive/00000590/01/Kehl\\_Victoria.pdf](http://edoc.ub.uni-muenchen.de/archive/00000590/01/Kehl_Victoria.pdf)
3. Schemper, M. Non-parametric analysis of treatment-covariate interaction in the presence of censoring. *Statistics in Medicine* (1998); **7**: 1257–1266.
4. Janse, M.J. et al. Identification of post acute myocardial infarction patients with potential benefit from prophylactic treatment with Amiodarone. A substudy of EMIAT. *European Heart Journal* (1998); **19**: 85–95.
5. Malik, M. et al. Depressed Heart Rate Variability Identifies Postinfarction Patients who Might Benefit From Prophylactic Treatment with Amiodarone. A Substudy of EMIAT . *Journal of the American College of Cardiology*(2000); **35.5**: 1263-1275.
6. Nardi, A. and Schemper, M. New Residuals for Cox Regression and Their Application to Outlier Screening. *Biometrics* (1999); **55**: 523-529.
7. Barlow, W.E. and Prentice, R.L. Residuals for relative risk regression. *Biometrika* (1988); **75**: 65-74.
8. Therneau, T.M., Grambsch, P.M., Fleming, T.R. Martingale-based residuals for survival models. *Biometrika* (1990); **77.1** : 147–160.
9. Breiman, L., Friedman, J.H., Olsen, R.A., Stone, C.J. *Classification & Regression Trees*. Pacific Grove, CA: Wodsworth & Brooks/Cole, advanced books & software, 1984.

10. Friedman, J.H. & Fisher, N.I. Bump hunting in high-dimensional data. *Statistics and Computing* (1999); **9.2**: 123–143.
11. Harrell, F.E., Lee, K.L., Mark, D.B. Tutorial in Biostatistics. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine* (1996); **15**: 361–387.
12. Tibshirani, R. & Knight, K. Model Search by Bootstrap “Bumping”. *Journal of Computational & Graphical Statistics* (1999); **8.4**.
13. Dannegger F. Tree stability diagnostics and some remedies for instability. *Statistics in Medicine* (2000); **19**: 475–491.
14. Becker, U. *Bump hunting software*. Computer software. Institute for Statistics, Ludwig-Maximilian University, Munich, 1999. Unix, downloadable at [www.stat.uni-muenchen.de/~ursula/](http://www.stat.uni-muenchen.de/~ursula/).