Journal of Statistical Software

October 2021, Volume 99, Issue 14. doi: 10.18637/jss.v099.i14

subtee: An R Package for Subgroup Treatment Effect Estimation in Clinical Trials

Nicolas M. Ballarini
Medical University of Vienna

Marius Thomas
Novartis Pharma AG

Gerd K. Rosenkranz
Medical University of Vienna

Björn Bornkamp
Novartis Pharma AG

Abstract

The investigation of subgroups is an integral part of randomized clinical trials. Exploration of treatment effect heterogeneity is typically performed by covariate-adjusted analyses including treatment-by-covariate interactions. Several statistical techniques, such as model averaging and bagging, were proposed recently to address the problem of selection bias in treatment effect estimates for subgroups. In this paper, we describe the **subtee** R package for subgroup treatment effect estimation. The package can be used for all commonly encountered type of outcomes in clinical trials (continuous, binary, survival, count). We also provide additional functions to build the subgroup variables to be used and to plot the results using forest plots. The functions are demonstrated using data from a clinical trial investigating a treatment for prostate cancer with a survival endpoint.

Keywords: model averaging, bootstrap, selection bias, treatment effect heterogeneity.

1. Introduction

Different patients may respond differently to the same treatment or drug. When developing new interventions, it is therefore crucial to evaluate the consistency of the treatment effects across relevant subgroups. On the other hand, exploratory subgroup analyses may help to identify subpopulations with a differential treatment effect.

The European Medicines Agency (EMA) guideline (European Medicines Agency 2019) suggests to identify and discuss a priori which subgroups of patients are expected to have an improved efficacy or improved risk-benefit. In this case, a small to a moderate number of scientifically interesting subgroups (around 5–20) is specified in a prospective data-independent manner. Still, since multiple subgroups are considered, a multiplicity problem arises increas-

ing the possibility of false-positive findings (Lipkovich, Dmitrienko, Muysers, and Ratitch 2018).

A great variety of approaches for exploratory subgroup analyses have been proposed in the literature, and interest in this problem continues to increase with the advent of personalized medicine. Some examples include Bayesian regression methods (Dixon and Simon 1991), recursive partitioning (Lipkovich, Dmitrienko, Denne, and Enas 2011), regression trees (Seibold, Zeileis, and Hothorn 2016), virtual twins (Foster, Taylor, and Ruberg 2011), standardization (Varadhan and Wang 2016). In particular, model averaging and bootstrap estimation techniques were proposed to mitigate the selection bias when estimating treatment effects in subpopulations (Bornkamp, Ohlssen, Magnusson, and Schmidli 2017; Thomas and Bornkamp 2017; Rosenkranz 2014, 2016, 2020). Both approaches look at the subgroup selection problem as a model selection problem. In model averaging, the predicted treatment effect of the selected subgroup is weighted across all considered models using the models' posterior probabilities. In the bootstrap approach, the estimates and the percentage of times the subgroups are selected in the bootstrap samples are used to adjust for possible selection bias.

In terms of software, a number of packages for subgroup analyses in clinical trials are available for the R statistical software (R Core Team 2021). We performed a search across the Comprehensive R Archive Network (CRAN) for packages that include either "subgroup" or "treatment effect" in their titles or description and detail here other packages for treatment effect estimation in presence of subgroups. The **beanz** package (Wang, Louis, Henderson, Weiss, and Varadhan 2018) provides functions for Bayesian hierarchical models, and the **DSBayes** package (Varadhan and Yao 2014) implements the (Bayesian) Dixon-Simon model for subgroup analysis with binary covariates. The **SIDES** package (Riviere 2021) implements subgroup identification based on differential effect search, and the **FindIt** package adapts the support vector machine classifier for estimation of treatment effects in subgroups (Egami, Ratkovic, and Imai 2019). Techniques for subgroup analyses using trees are implemented in the **model4you** package (Seibold, Zeileis, and Hothorn 2019), the **TSDT** package (Battioui, Denton, and Shen 2018) and the **quint** package (Dusseldorp, Doove, van de Put, Van Mechelen, and Claramunt Gonzalez 2020). The **SubgrpID** package implements four algorithms for developing threshold-based multivariate (prognostic/predictive) biomarker signatures (Huang, Sun, Trow, Chatterjee, Chakravartty, Tian, and Devanarayan 2017; Duong 2021). There are also several packages implementing methods to model regression scores for forming subgroups, such as the **MMMS** (Li, Guennel, Marshall, and Cheung 2014), **personalized** (Chen, Tian, Cai, and Yu 2017), **sparsereg** (Ratkovic and Tingley 2016), **credsubs** (Schnell, Fiecas, and Carlin 2020) and **subgroup** (Schou 2014) packages. The **StratifiedMedicine** package (Jemielita 2021) provides analytic and visualization tools to aid in stratified and personalized medicine. Additionally, there are two packages, **SubgrPlots** (Ballarini and Chiu 2020) and **subscreen** (Kirsch, Jeske, Lippert, Schmelter, Muysers, and Kulmann 2021), that are designed to provide graphical displays of treatment effects in subgroups. The graphics in these two packages are built using naive estimates, and do not take into account the fact that many subgroups might have been investigated. Finally, a curated list of software for subgroup analysis is maintained on the Biopharmaceutical Network website (`http://biopharmnet.com/subgroup-analysis-software/`).

In this manuscript we present the R package **subtee** (Ballarini, Bornkamp, Thomas, and Magnusson 2021) that implements model averaging and bootstrapping for obtaining treatment effect estimates in subgroups. Our aim is to provide a flexible user-friendly set of tools for

subgroup analyses, that can be used for a wide range of clinical trials. The package can be applied to any situation where a generalized linear model is applicable, as well as with models with survival endpoints or count data. Results can be displayed as tables or using forest plots, for which we provide dedicated functions.

This manuscript is organized as follow: In Section 2, we introduce the statistical methodology framework for **subtee**, in Section 3 we briefly present the functions in the package and their usage, and in Section 4 we present an example using a data set from a clinical trial investigating a treatment for prostate cancer. We end with a discussion in Section 5.

2. Subgroup analyses in clinical trials

Consider a clinical trial in which n subjects are investigated and a response y_i for each individual $i = 1, \ldots, n$ is observed after being randomized to either an experimental treatment ($z_i = 1$) or control ($z_i = 0$). Further consider P subgroups that are defined by covariates or factors observed at baseline. We denote a subgroup as $S_p = \{i \in \{1, \ldots, n\} | s_{pi} = 1\}$, $p = 1, \ldots, P$; where s_{pi} is defined as the patient-level membership variable for patient i in subgroup p that takes values 0 or 1. Then, P generalized linear models are fitted such that

$$M_p : \; h\left(\mu_{pi}\right) = \alpha_p + \beta_p z_i + \left(\gamma_p + \delta_p z_i\right) s_{pi} + \sum_{k=1}^{K} \tau_k x_{ik}, \tag{1}$$

where h is the link function, $\mu_{pi} = E_p[Y_i]$ is the expectation of the response under model M_p and x_{ik} are additional covariates we control for. For survival data, a proportional hazards model can be used:

$$M_p : \; \lambda_{pi}(t) = \lambda_{p0}(t) \exp\left\{\beta_p z_i + (\gamma_p + \delta_p z_i) s_{pi} + \sum_{k=1}^{P} \tau_k x_{ik}\right\}. \tag{2}$$

In both models, γ_p represents a prognostic effect (modifying the response independent of treatment) and δ_p a predictive effect (modifying the response to treatment) of a subgroup.

It is of interest to identify a subgroup with a differential treatment effect. In other words, we want to search for a subgroup in which the treatment effect is different than in its complement. Different rules to select such a subgroup may be adopted. For example, one may want to select the subgroup for which the p value for the interaction term δ_p is the minimum. Another rule may be to select the subgroup for which the model M_p gives the smaller Bayes information criterion (BIC) or Akaike information criterion (AIC) value. No matter what selection rule is adopted, it is well known that a data driven selection of a subgroup will lead to overestimating the treatment effect (Ruberg and Shen 2015; Thomas and Bornkamp 2017).

2.1. Unadjusted estimates for treatment effects

Assume we are interested in estimating the treatment effect in the selected subgroup and its complement. Consider S the set of indexes $\{1, \ldots, n\}$ corresponding to the subjects in the selected subgroup. An estimate for $\Delta(S)$, the treatment effect in S, may be obtained by predicting the individual treatment effects and averaging over the patients in the subgroup. We predict the treatment effect for patient i under model M_p as

$$\mu_{pi|z_i=1} - \mu_{pi|z_i=0} = \beta_p + s_{pi}\delta_p.$$

Then the treatment effect for subgroup S is given by

$$\Delta_p(S) = \beta_p + w\delta_p, \tag{3}$$

where $w = |S \cap S_p|/|S|$. Note that the treatment effect simplifies to $\beta_p + \delta_p$ if $S = S^{(p)}$, and β_p if S is the complement of $S^{(p)}$.

For the unadjusted or naive estimates, we simply estimate the treatment effect as in Equation 3 from the model that corresponds to the selected subgroup. The unadjusted treatment effect estimator for an identified subgroup S defined by model M_p is therefore simply

$$\hat{\Delta}_{unadj} := \hat{\Delta}_p(S).$$

Additionally, it may be of interest to look only at the differences between the treatment effect in the subgroup and in the complement. Under this model, this is simply the interaction of the subgroup-defining covariate and treatment, δ_p.

2.2. Model averaging for treatment effect estimation

Model averaging (MA) in the subgroup analysis framework was introduced to address the problem of selection bias (Bornkamp *et al.* 2017; Thomas and Bornkamp 2017). The main idea behind MA is that subgroup selection can be viewed as a model selection procedure since each subgroup defines a statistical model $M_1, \ldots, M_P$. By averaging over all models we can better represent the uncertainty in the selection process.

In a fully Bayesian formulation, we would use prior distributions for all the parameters in each model $M_p, p = 1, \ldots, P$. Prior information might be available for some of these parameters (e.g., we might have prior information of the response under the control treatment), but usually little information exists on the other parameters, so that weakly informative priors would be used for all models. In addition, prior probabilities for the models $M_1, \ldots, M_P$ are required, based on the plausibility of the different subgroups.

The approach implemented in the **subtee** package, however, performs approximate inference using MA based on the BIC. This approach uses maximum likelihood estimation instead of defining prior distributions for the model parameters. While this approach is not fully Bayesian it is computationally very efficient and yields similar results as if one assumes weakly informative priors (see Bornkamp *et al.* 2017 for a comparison of the methods).

The prior model weights $P(M_p)$ still need to be specified in this approach, and often equal prior weights for all models might be plausible so that $P(M_p) = 1/P$ (this is the default in the package implementation). Additionally, one can also add a prior weight for the model without any treatment by subgroup interaction. Posterior model weights are then obtained by Bayes' theorem:

$$P(M_p|\boldsymbol{y}) = \frac{P(\boldsymbol{y}|M_p)P(M_p)}{\sum_{p=1}^{P} P(\boldsymbol{y}|M_p)P(M_p)}.$$

where $\boldsymbol{y} = (y_i)_{i=1,\ldots,n}$.

For the implementation in the package, we use the BIC approximations for the model weights as is proposed in Raftery (1995):

$$P(M_p|\boldsymbol{y}) \approx \frac{\exp\left(-0.5BIC(M_p)\right)P(M_p)}{\sum_{p=1}^{P}\exp\left(-0.5BIC(M_{p'})\right)P(M_p)}.$$

The overall posterior distribution of the treatment effect in subgroup S, $\Delta_{ma}(S)$, is then a mixture of all the posterior distributions under each model:

$$f\left(\Delta_{ma}(S)|\boldsymbol{y}\right) = \sum_{p=1}^{P} P(M_p|\boldsymbol{y}) f\left(\Delta_{ma}(S)|M_p, \boldsymbol{y}\right). \tag{4}$$

The median of the posterior distribution can be used as a point estimate, while quantiles can be used to derive credible intervals. For the models we implement in the package, the posterior distributions under each model $f(\Delta_{ma}(S)|M_p, \boldsymbol{y})$ are approximately normally distributed and, therefore, the overall posterior $f(\Delta_{ma}(S)|\boldsymbol{y})$ is a mixture of normal distributions.

Note that it is also straightforward to obtain an estimate of the difference in treatment effect between a subgroup and its complement, $\delta_{ma}(S) = \Delta_{ma}(S) - \Delta_{ma}(\overline{S})$, using:

$$f\left(\delta_{ma}(S)|\boldsymbol{y}\right) = \sum_{p=1}^{P} P(M_p|\boldsymbol{y}) f\left(\delta_{ma}(S)|M_p, \boldsymbol{y}\right).$$

2.3. Bagged estimates for treatment effect

We also implement the methods in Rosenkranz (2016, 2014) which use bootstrapping to account for model selection uncertainty and estimation bias after selection. The methods were originally proposed to estimate the interaction between subgroups and treatment, but we extended the implementation to the treatment effects in subgroup and complement as well. We give details here on the estimation of the interaction term, δ_p.

The idea behind the bagged estimate is that when using the original data, a subgroup S_p will be selected if the model fit of M_p is better (measured in terms of the BIC or AIC) than the model fit of $M_{p'}$ (where $p' \neq p$) and of a model with zero interaction, otherwise no selection takes place. The same process is then replicated for each bootstrap sample b, where a subgroup S_p will be selected if the model fit of M_{bp} is better than the model fit of $M_{bp'}$ (where $p' \neq p$) and of a model with zero interaction. Then the proportion of times the subgroup S_p is selected in the bootstrap samples is used to adjust for the selection bias.

Consider B bootstrap samples from the original data. For $b = 1, \ldots, B$, let $(Y^*_{b1}, \ldots, Y^*_{bN})$ be a bootstrap sample from the original data. Let $(z^*_{b1}, \ldots, z^*_{bN})$, $(s^*_{bp1}, \ldots, s^*_{bpN})$, and $(x^*_{b1k}, \ldots, x^*_{bNk})$ be the corresponding treatment indicators, group indicators, and covariates in the bootstrap samples, respectively. Note that the bootstrap samples can be stratified on treatment (the default in the package implementation), so that independent samples are drawn from each treatment group.

For each subgroup $p = 1, \ldots, P$ and bootstrap sample $b = 1, \ldots, B$ we fit the model in Equation 1 using the bootstrapped data:

$$M_{bp}: \; h\left(E_p[Y^*_{bi}]\right) = \alpha^*_{bp} + \beta^*_{bp} z^*_{bi} + (\gamma^*_{bp} + \delta^*_{bp} z^*_{bi}) s^*_{bpi} + \sum_{k=1}^{K} \tau^*_{bk} x^*_{bik}. \tag{5}$$

Rosenkranz (2016) provides a bias-reduced estimator with decreased variability as:

$$\tilde{\delta}^*_p = 2\hat{\delta}^*_p - \overline{\delta}^*_p, \tag{6}$$

where $\hat{\delta}_p^*$ is the average of the maximum likelihood estimators $\hat{\delta}_{bp}^*$ of δ_{bp}^* across all bootstrap samples b:

$$\hat{\delta}_p^* = \frac{1}{B} \sum_{b=1}^{B} \hat{\delta}_{bp}^*$$

and $\overline{\delta}_p^*$ is an estimator of δ_p given that subgroup S_p provided the best fit, which is calculated as the average of the $\hat{\delta}_{bp}^*$ estimates in the bootstrap samples b where S_p was selected:

$$\overline{\delta}_p^* = \frac{\sum_{b=1}^{B} u_{bp} \hat{\delta}_{bp}^*}{\sum_{b=1}^{B} u_{bp}}.$$

where u_{bp} is an indicator variable that takes 1 if S_p was selected in the bootstrap sample b and 0 otherwise.

The estimates in Equation 6 are displayed as main results in the package's function. The variance of $\tilde{\delta}_p^*$ is calculated by applying a bias-corrected infinitesimal jackknife estimator (Efron 2014; Wager, Hastie, and Efron 2014) and is also provided in Rosenkranz (2016).

We construct approximate confidence intervals using estimate $\pm z_{\alpha/2}$ standard deviation, where $z_{\alpha/2}$ is the $\alpha/2$ percentile point of a standard normal distribution.

To obtain the adjusted estimates for the treatment effect in subgroup and complement, we simply replace the target δ_p with β_p or $\beta_p + \delta_p$ accordingly.

Although it is technically possible to obtain an estimate for each subgroup S_p, $p = 1, \ldots, P$, the bootstrap corrects for the fact that a subgroup was selected based on the original data and is therefore more appropriate to report the corrected estimate only for the selected subgroup.

3. R implementation

The **subtee** package is available from CRAN at http://CRAN.R-project.org/package=subtee. Each of the methods described in the previous section has its respective fitting function. However, they share most of the function arguments. The main functions in the package are:

```
unadj(resp, trt, subgr, covars = NULL, data,
  fitfunc = c("lm", "glm", "glm.nb", "survreg", "coxph", "rlm"),
  event, exposure, level = 0.1, ...)

modav(resp, trt, subgr, covars = NULL, data,
  fitfunc = c("lm", "glm", "glm.nb", "survreg", "coxph", "rlm"),
  event, exposure, level = 0.1, prior = 1, nullprior = 0, ...)

bagged(resp, trt, subgr, covars = NULL, data,
  fitfunc = c("lm", "glm", "glm.nb", "survreg", "coxph"),
  event, exposure, level = 0.1,  B = 100, mc.cores = 1,
  stratified = TRUE, select.by = c("BIC", "AIC"), ...)
```

The arguments are specified as:

- `resp`: Character giving the name of the response variable. The variable can be either defined in the global environment or in the data set `data` specified below. For interactive use it is also possible to use unquoted names (i.e., `unadj(resp, ...)` instead of `unadj("resp", ...)`), avoid this for non-interactive use of the function.

- `trt`: Character giving the name of the treatment variable. The variable can be either defined in the global environment or in the data set `data` specified below. Note that the treatment variable itself needs to be defined as a numeric variable, with control coded as 0, and treatment coded as 1. For interactive use it is also possible to use unquoted names (as for the `resp` argument, see above).

- `subgr`: Character vector giving the variable names in `data` to use as subgroup identifiers. Note that the subgroup variables in `data` need to be numeric 0–1 variables.

- `covars`: Formula specifying additional covariates to be included in the models (need to be available in `data`).

- `data`: Data frame containing the variables referenced in `resp`, `trt`, `subgr` and `covars` (and possibly `event` and `exposure`).

- `fitfunc`: Model fitting functions. Currently one of `"lm"`, `"glm"`, `"glm.nb"`, `"survreg"`, `"coxph"`, `"rlm"`.

- `event`: Character giving the name of the event variable. Has to be specified when using fit functions `"survreg"` and `"coxph"`. The variable can be either defined in the global environment or in the data-set `data`.

- `exposure`: Character giving the name of the exposure variable, needed for negative binomial regression, when using fit functions `"glm.nb"`. This is typically the time each patient is exposed to the drug. The fitted model uses the call `glm.nb(. ~ . + offset(log(exposure)))`. The variable needs to be defined either in the global environment or in the data-set `data`.

- `level`: Confidence level for confidence intervals for treatment effect estimates.

- `prior` (only in `modav`): Vector of prior model/subgroup probabilities of the same length as the number of columns in `subgr`. Probabilities can be specified up to proportionality. If a vector of length 1 is specified automatically equal prior weights are assumed (equal weights are the default).

- `nullprior` (only in `modav`): Numeric giving the prior model probability of the model without any subgroup effect. This needs to be specified on the same scale as the prior argument. E.g., if there are 2 subgroups, `prior = c(1, 1)` (or `prior = 1`) and `nullprior = 2` the prior probabilities will be 1/4 and 1/4 for the two subgroup models and 1/2 for the null model. By default a prior probability of 0 is attached to this model.

- `B` (only in `bagged`): A numeric input. The number of bootstrap samples to perform.

- `mc.cores` (only in `bagged`): A numeric input. This argument is passed to the `mclapply` function to perform computations in parallel. If `mc.cores = 1`, then `lapply` is used.

- `stratified` (only in `bagged`): Should the bootstrap resampling be done stratifying by treatment group? (default: `TRUE`).

- `select.by` (only in `bagged`): Should the model selection be done using BIC or AIC? (default: `"BIC"`).

- `...`: Other arguments passed to the model fitting function.

Internally, these functions fit the model in Equation 1 using the specified `fitfunc` recursively over the P subgroups `subgr`. The variables added in the `covars` argument are included in the fitting formula as main effects for all subgroup models. Note that a main effect and interaction term with treatment for subgroup p is added to the model M_p, $p = 1, \ldots, P$. If one subgroup specified in `subgr` needs to be added as prognostic in each $M_{p'}$, $p' \neq p$, then this needs to be specified in the `covars` argument as well.

The three functions output a '`subtee`' object that contains a table with the treatment effect estimates in the subgroups, a table with the treatment-subgroup interaction estimates, and complementary information from the model fits. Additionally, the package includes dedicated methods for the generic functions `print`, `summary`, `confint`, and `plot`, which produces a forest plot displaying the treatment effect estimates in the subgroups using the **ggplot2** package (Wickham 2009).

The package also includes another function that may be useful when performing subgroup analysis: `subbuild`. The `subbuild` function takes categorical or continuous baseline covariate vectors and builds a matrix of binary subgroup indicator variables in the columns. This matrix of candidate subgroups can then be used as input for the estimation functions in the package, but might also be of interest in general.

Finally, the package provides simulated data sets with normal (`datnorm`), survival (`datsurv`), count (`datcound`) and binary (`datbin`) endpoints that are used in the documentation examples of the corresponding functions as well as a function `get_prca_data` that wraps the code to download the prostate cancer data set that we use here as example (internet connection required).

4. Example

We use the prostate cancer data set that was used in Rosenkranz (2016) to illustrate the usage of the package. The data set consists of $n = 475$ subjects randomized to a placebo group and three dose levels of diethylstilbestrol. The data is provided with the placebo and the lowest dose level of diethylstilbestrol combined to give the control arm, and the higher doses of diethylstilbestrol combined to give an active treatment arm. The considered endpoint is survival time in months. There are six subgroup-defining variables to consider: existence of bone metastasis (bm), disease stage (3 or 4), performance (pf), history of cardiovascular events (hx), age, and weight. While age and weight are continuous covariates, they are dichotomized (age ≤ 65, > 65 and weight ≤ 100, > 100) for obtaining subgroups as in Rosenkranz (2016). As the considered endpoint is survival time in months, we fit Cox proportional hazards models (Cox 1972).

4.1. Data preparation

We first use the `get_prca_data` function that runs without any argument to download the data set. Then, the `subbuild` function creates the desired candidate subgroups. This function takes the data set as a first argument, and then a series of expressions to define the subgroup indicator variables. Note that we also use the option `dupl.rm = TRUE` to remove duplicate subgroups. The output of the `subbuild` function is a '`data.frame`' that might then be concatenated with the original data set to be used in the other functions.

```
R> library("subtee")
R> prca <- get_prca_data()
R> cand.groups <- subbuild(prca, BM == 1, PF == 1, HX == 1, STAGE == 4,
+    AGE > 65, WT > 100)
R> fitdat <- cbind(prca, cand.groups)
```

4.2. Unadjusted estimates for treatment effects

The unadjusted estimates for treatment effects are obtained via the `unadj` function. We fit the models including the six covariates in the data set as prognostic factors as well, which are added through the `covars` argument as a formula. Since we have a survival endpoint, we use `coxph` from the **survival** package (Therneau and Grambsch 2000; Therneau 2021) as fitting function. The function loops through all the variables specified in the `subgr` argument, fitting the models in Equation 2. In this example, we make use of the `...` argument to pass the option `ties = "breslow"` to `coxph`.

```
R> subgr.names <- names(cand.groups)
R> prog <- paste0("`", subgr.names,"`", collapse = " + ")
R> prog <- as.formula(paste(" ~ ", prog))
R> res_unadj <- unadj(resp = "SURVTIME", trt = "RX", subgr = subgr.names,
+    data = fitdat, covars = prog, event = "CENS", fitfunc = "coxph",
+    ties = "breslow")
R> res_unadj

Trt. Effect Estimates
        Group     Subset      LB    trtEff       UB
1     BM == 1   Subgroup -1.1949 -0.785335 -0.37579
2     BM == 1 Complement -0.2442 -0.040729  0.16273
3     PF == 1   Subgroup -0.3965  0.118040  0.63257
4     PF == 1 Complement -0.4212 -0.224282 -0.02734
5     HX == 1   Subgroup -0.2547  0.006503  0.26774
6     HX == 1 Complement -0.6199 -0.363698 -0.10754
7  STAGE == 4   Subgroup -0.6526 -0.376238 -0.09986
8  STAGE == 4 Complement -0.2748 -0.027431  0.21998
9    AGE > 65   Subgroup -0.2517 -0.052995  0.14568
10   AGE > 65 Complement -1.4451 -0.962738 -0.48037
11   WT > 100   Subgroup -0.5730 -0.274251  0.02451
12   WT > 100 Complement -0.3622 -0.126989  0.10819
```

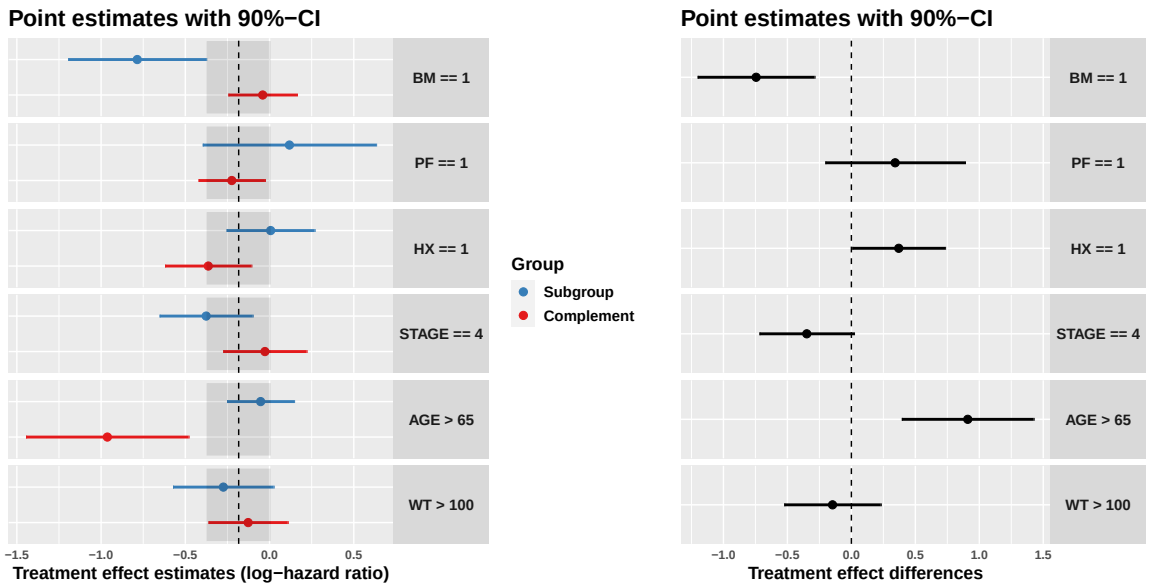

Figure 1: Forest plot of unadjusted treatment effects for subgroups and complements (left) and treatment-subgroup interactions (right). These graphics are the output of the `plot` function applied to a '`subtee`' object.

```
Difference in Trt. Effect vs Complement
       Group        LB trtEffDiff       UB
1    BM == 1 -1.201671    -0.7446 -0.28754
2    PF == 1 -0.203438     0.3423  0.88808
3    HX == 1  0.007603     0.3702  0.73280
4 STAGE == 4 -0.718343    -0.3488  0.02073
5   AGE > 65  0.394367     0.9097  1.42512
6   WT > 100 -0.524689    -0.1473  0.23016

Subgroup Models fitted with "coxph"
Effect estimates in terms of the log-hazard ratios
```

The output shows first the treatment effect estimates (`trtEff`) and the lower and upper bounds of the confidence intervals (`LB` and `UB` respectively). A second table is displayed with the information on the difference in treatment effects in subgroups vs. their complements.

Although the significance level for the confidence intervals needs to already be fixed in the fitting functions, there is also the option to use the generic function `confint` to recalculate them at a different level. For example, using `confint(res_unadj, level = 0.80)` generates a new '`subtee`' object identical to `res_unadj` except for the confidence intervals, which are calculated using a 80% confidence level instead of the default 90%. Moreover, using the `summary` method the user obtains further information such as the p values for the treatment-by-covariate interactions and groups sizes.

The `plot` generic function can be used with '`subtee`' objects to obtain a forest plot of the treatment effects (`type = "trtEff"`) or the interactions (`type = "trtEffDiff"`) and their confidence intervals (Figure 1). To ease the visual check for heterogeneity, the plot with

the treatment effects in subgroups and complements also shows the overall treatment effect under the model with no treatment-subgroup interactions with a dashed line, as well as its confidence intervals in the gray shaded area. For the plot of the interactions, the dashed line is at 0, which indicates no interaction between subgroup and treatment. Further modifications or new elements can be added to the plots using **ggplot2** functions.

For the prostate cancer example, we see that the new treatment leads to better outcomes when compared to control, as the overall treatment effect is negative. However, its confidence interval covers the no-effect value of 0. Using the unadjusted estimates for subgroups leads to the conclusion that the subgroups defined by age and by existence of bone metastasis may have a differential treatment effect.

```
R> plot(res_unadj, show.compl = TRUE)
R> plot(res_unadj, type = "trtEffDiff")
```

4.3. Model averaging for treatment effect estimation

We use the `modav` function to obtain the model averaging estimates. In this case, we use the same options as in the `unadj` function. We use the default settings, so that all models have equal prior weights and there is zero prior weight for the model without treatment by subgroup interaction.

```
R> res_modav = modav(resp = "SURVTIME", trt = "RX", subgr = subgr.names,
+    data = fitdat, covars = prog, event = "CENS", fitfunc = "coxph",
+    ties = "breslow")
R> res_modav

Trt. Effect Estimates
        Group     Subset      LB  trtEff        UB
1     BM == 1   Subgroup -1.0089 -0.3182 -0.082706
2     BM == 1 Complement -0.3633 -0.1577  0.080986
3     PF == 1   Subgroup -0.4800 -0.2247 -0.006514
4     PF == 1 Complement -0.3813 -0.1889  0.004497
5     HX == 1   Subgroup -0.3421 -0.1533  0.044871
6     HX == 1 Complement -0.4253 -0.2228 -0.029070
7  STAGE == 4   Subgroup -0.4663 -0.2493 -0.048809
8  STAGE == 4 Complement -0.3614 -0.1547  0.086863
9    AGE > 65   Subgroup -0.3016 -0.0915  0.121691
10   AGE > 65 Complement -1.3799 -0.7415 -0.086948
11   WT > 100   Subgroup -0.3732 -0.1769  0.021429
12   WT > 100 Complement -0.3909 -0.2040 -0.016736

Difference in Trt. Effect vs Complement
       Group        LB trtEffDiff       UB
1    BM == 1 -0.994109   -0.08220 -0.02315
2    PF == 1 -0.291039    0.01613  0.03506
3    HX == 1  0.017764    0.06258  0.12345
```

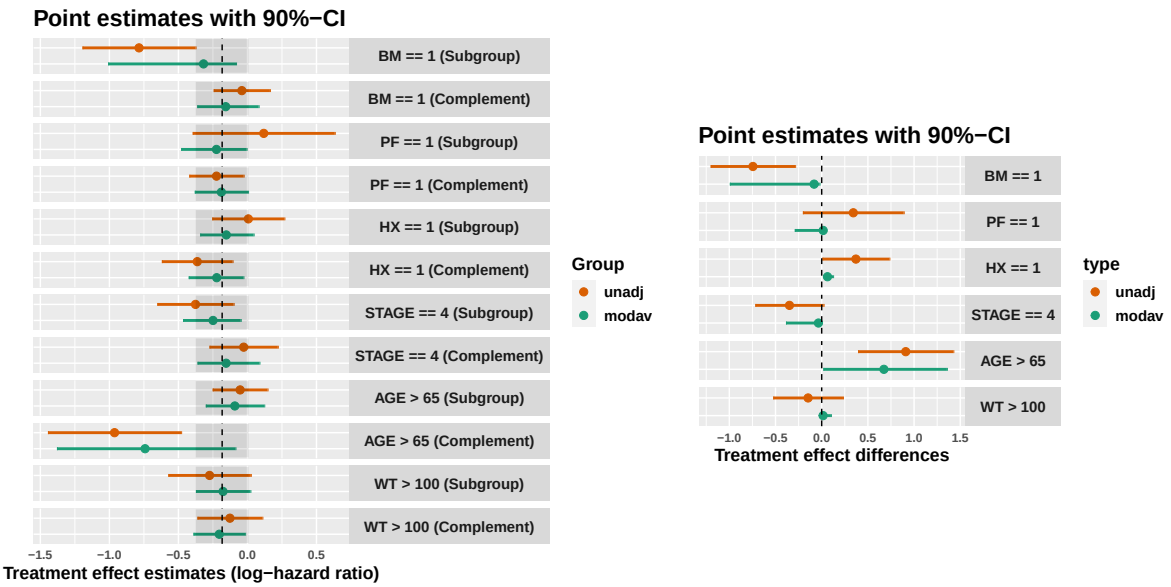

Figure 2: Forest plot of treatment effects for subgroups and complements (left) and treatment-subgroup interactions (right). The plots compare the results using unadjusted estimates and those obtained with model averaging. These graphics are the output of the `plot` function using two '`subtee`' objects resulting from the `unadj` and the `modav` functions.

```
4 STAGE == 4 -0.386656   -0.03536 -0.01087
5   AGE > 65  0.017744    0.67341  1.35540
6   WT > 100 -0.001293    0.01531  0.09971

Subgroup Models fitted with "coxph"
Effect estimates in terms of the log-hazard ratios
```

Using the `plot` function with the result of the `modav` function, we obtain a forest plot with the estimates like the one in Figure 1. However, we can also provide both the results of the `unadj` and `modav` functions and obtain a comparison of the estimates (Figure 2).

```
R> plot(res_unadj, res_modav, show.compl = TRUE)
R> plot(res_unadj, res_modav, type = "trtEffDiff")
```

For objects resulting from the `modav` function, the `summary` method displays the model posterior probabilities rather than the p values for the treatment-by-covariate interactions.

4.4. Bagged estimates

Finally, we obtain the bagged estimates using the `bagged` function. In this function we must also specify how the subgroup is selected (`select.by = "BIC"`) and the number of bootstrap samples to use (`B = 2000`). We also let the default option for the `stratify` function parameter, so that the bootstrapping is stratified over treatment. Note that we use `set.seed` to obtain reproducible results.

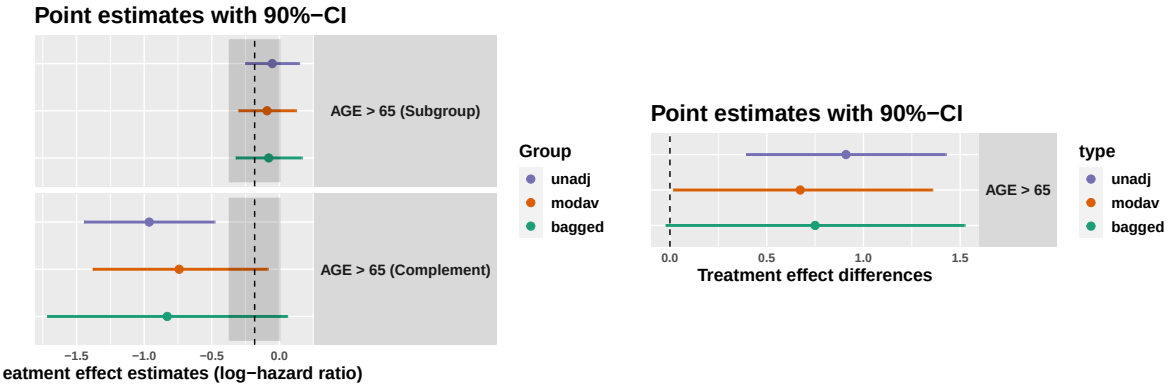

Figure 3: Forest plot of treatment effects for subgroups and complements (left) and treatment-subgroup interactions (right) using unadjusted, model averaging and bagged estimates. These graphics are the output of the `plot` function applied to three '`subtee`' objects.

```
R> set.seed(46312)
R> res_bagged = bagged(resp = "SURVTIME", trt = "RX", subgr = subgr.names,
+    data = fitdat, covars = prog, event = "CENS", fitfunc = "coxph",
+    ties = "breslow", select.by = "BIC", B = 2000)
R> res_bagged

Trt. Effect Estimates
     Group      Subset      LB   trtEff      UB
1 AGE > 65   Subgroup -0.3228 -0.07913 0.16458
2 AGE > 65 Complement -1.7162 -0.82958 0.05703

Difference in Trt. Effect vs Complement
     Group       LB trtEffDiff    UB
1 AGE > 65 -0.02153     0.7504 1.522

 AGE > 65 is the selected subgroup.
 It was selected in 49.75% of 2000 bootstrap samples.

Subgroup Models fitted with "coxph"
Effect estimates in terms of the log-hazard ratios
```

The bootstrap method provides bias-adjusted estimates, which corrects for the bias that is introduced when selecting a subgroup. Therefore, it only makes sense to display the results of the selected subgroup. While the selection percentage for the selected subgroup is displayed in the output of the function, the user may obtain the percent of selection for each subgroup using the `summary` method. This is important so that the user can assess the reliability of the results.

Finally, the plot function might take the results from the three methods to display a comparison of the estimates.

```
R> plot(res_unadj, res_modav, res_bagged, show.compl = TRUE)
R> plot(res_unadj, res_modav, res_bagged, type = "trtEffDiff")
```

5. Discussion

This article described the **subtee** R package for subgroup treatment effects estimation. The functions that are provided are helpful for exploratory subgroup analysis in randomized clinical trials where it is necessary to examine treatment effect heterogeneity. The package works with widely used modeling functions and provides the flexibility to fit any generalized linear model, Cox regression models, and parametric survival models.

Binary subgroup indicators can be supplied by the user or generated from baseline covariates. The treatment effect of the experimental treatment vs. control is estimated for each subgroup, or the treatment-subgroup interactions are investigated. Either case, the analysis may suffer from overfitting/selection bias if naive estimates are used. It is well established that subgroup analyses that lack pre-specification and use a large number of subgroups without adjusting for multiple comparison may lead to finding spurious subgroup effects.

Two estimation techniques are available in the package to allow researchers to implement recently proposed methods to address the issue of selection bias in the estimation: model averaging and bagging. These techniques share the same philosophy that subgroup analysis is a model selection problem, taking into account the uncertainty in subgroup selection. As shown in the examples, in practice this usually results in having wider confidence intervals and a shrinkage of the subgroup-specific treatment effects towards the overall effect, which helps avoiding overoptimistic conclusions.

We focused on the case where subgroups of interest are specified in a prospective data-independent manner, which will usually result in a small number of subgroups to be evaluated. Even in these cases, since multiple subgroups are considered, a multiplicity problem arises and appropriate analysis methods are needed, such as those that we implement in our package. The methods implemented in the **subtee** package were primarily developed for dealing with a relatively small number of pre-specified subgroups or biomarkers. We did not investigate how these approaches would work in retrospective data-driven situations where the definition and selection of subgroups is post-hoc.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Sklodowska-Curie grant agreement No. 633567. Gerd Rosenkranz received funding from the UK Medical Research Council under the Project No. MR/M005755/1. The views expressed are those of the authors and should not be attributed to the funding institutions or the organizations with which the authors are affiliated.

References

Ballarini N, Bornkamp B, Thomas M, Magnusson B (2021). ***subtee**: Subgroup Treatment Effect Estimation in Clinical Trials*. R package version 1.0.0, URL https://CRAN.R-project.org/package=subtee.

Ballarini N, Chiu YD (2020). ***SubgrPlots**: Graphical Displays for Subgroup Analysis in*

Clinical Trials. R package version 0.1.3, URL https://CRAN.R-project.org/package=SubgrPlots.

Battioui C, Denton B, Shen L (2018). ***TSDT**: Treatment-Specific Subgroup Detection Tool*. R package version 1.0.0, URL https://CRAN.R-project.org/package=TSDT.

Bornkamp B, Ohlssen D, Magnusson BP, Schmidli H (2017). "Model Averaging for Treatment Effect Estimation in Subgroups." *Pharmaceutical Statistics*, **16**(2), 133–142. doi:10.1002/pst.1796.

Chen S, Tian L, Cai T, Yu M (2017). "A General Statistical Framework for Subgroup Identification and Comparative Treatment Scoring." *Biometrics*, **73**(4), 1199–1209. doi:10.1111/biom.12676.

Cox DR (1972). "Regression Models and Life-Tables." *Journal of the Royal Statistical Society B*, **34**(2), 187–220. doi:10.1111/j.2517-6161.1972.tb00899.x.

Dixon DO, Simon R (1991). "Bayesian Subset Analysis." *Biometrics*, **47**(3), 871–881. doi:10.2307/2532645.

Duong T (2021). ***prim**: Patient Rule Induction Method (PRIM)*. R package version 1.0.20, URL https://CRAN.R-project.org/package=prim.

Dusseldorp E, Doove L, van de Put J, Van Mechelen I, Claramunt Gonzalez J (2020). ***quint**: Qualitative Interaction Trees*. R package version 2.1.0, URL https://CRAN.R-project.org/package=quint.

Efron B (2014). "Estimation and Accuracy after Model Selection." *Journal of the American Statistical Association*, **109**(507), 991–1007. doi:10.1080/01621459.2013.823775.

Egami N, Ratkovic M, Imai K (2019). ***FindIt**: Finding Heterogeneous Treatment Effects*. R package version 1.2.0, URL https://CRAN.R-project.org/package=FindIt.

European Medicines Agency (2019). "Guideline on the Investigation of Subgroups in Confirmatory Clinical Trials." Available at https://www.ema.europa.eu/en/investigation-subgroups-confirmatory-clinical-trials.

Foster JC, Taylor JMG, Ruberg SJ (2011). "Subgroup Identification from Randomized Clinical Trial Data." *Statistics in Medicine*, **30**(24), 2867–2880. doi:10.1002/sim.4322.

Huang X, Sun Y, Trow P, Chatterjee S, Chakravartty A, Tian L, Devanarayan V (2017). "Patient Subgroup Identification for Clinical Drug Development." *Statistics in Medicine*, **36**(9). doi:10.1002/sim.7236.

Jemielita T (2021). ***StratifiedMedicine**: Stratified Medicine*. R package version 1.0.4, URL https://CRAN.R-project.org/package=StratifiedMedicine.

Kirsch B, Jeske S, Lippert S, Schmelter T, Muysers C, Kulmann H (2021). ***subscreen**: Systematic Screening of Study Data for Subgroup Effects*. R package version 3.0.5, URL https://CRAN.R-project.org/package=subscreen.

Li L, Guennel T, Marshall S, Cheung LWK (2014). ***MMMS**: Multi-Marker Molecular Signature for Treatment-Specific Subgroup Identification*. R package version 0.1, URL https://CRAN.R-project.org/src/contrib/Archive/MMMS/.

Lipkovich I, Dmitrienko A, Denne J, Enas G (2011). "Subgroup Identification Based on Differential Effect Search – A Recursive Partitioning Method for Establishing Response to Treatment in Patient Subpopulations." *Statistics in Medicine*, **30**(21), 2601–2621. doi:10.1002/sim.4289.

Lipkovich I, Dmitrienko A, Muysers C, Ratitch B (2018). "Multiplicity Issues in Exploratory Subgroup Analysis." *Journal of Biopharmaceutical Statistics*, **28**(1), 63–81. doi:10.1080/10543406.2017.1397009.

Raftery AE (1995). "Bayesian Model Selection in Social Research." *Sociological Methodology*, **25**, 111–163. doi:10.2307/271063.

Ratkovic M, Tingley D (2016). ***sparsereg**: Sparse Bayesian Models for Regression, Subgroup Analysis, and Panel Data*. R package version 1.2, URL https://CRAN.R-project.org/package=sparsereg.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Riviere MK (2021). ***SIDES**: Subgroup Identification Based on Differential Effect Search*. R package version 1.16, URL https://CRAN.R-project.org/package=SIDES.

Rosenkranz GK (2014). "Bootstrap Corrections of Treatment Effect Estimates Following Selection." *Computational Statistics & Data Analysis*, **69**, 220–227. doi:10.1016/j.csda.2013.08.010.

Rosenkranz GK (2016). "Exploratory Subgroup Analysis in Clinical Trials by Model Selection." *Biometrical Journal*, **58**(5), 1217–1228. doi:10.1002/bimj.201500147.

Rosenkranz GK (2020). *Exploratory Subgroup Analyses in Clinical Research*. John Wiley & Sons.

Ruberg SJ, Shen L (2015). "Personalized Medicine: Four Perspectives of Tailored Medicine." *Statistics in Biopharmaceutical Research*, **7**(3), 214–229. doi:10.1080/19466315.2015.1059354.

Schnell PM, Fiecas M, Carlin BP (2020). "**credsubs**: Multiplicity-Adjusted Subset Identification." *Journal of Statistical Software*, **94**(7), 1–22. doi:10.18637/jss.v094.i07.

Schou IM (2014). ***subgroup**: Methods for Exploring Treatment Effect Heterogeneity in Subgroup Analysis of Clinical Trials*. R package version 1.1, URL https://CRAN.R-project.org/package=subgroup.

Seibold H, Zeileis A, Hothorn T (2016). "Model-Based Recursive Partitioning for Subgroup Analyses." *The International Journal of Biostatistics*, **12**(1), 45–63. doi:10.1515/ijb-2015-0032.

Seibold H, Zeileis A, Hothorn T (2019). "**model4you**: An R Package for Personalised Treatment Effect Estimation." *Journal of Open Research Software*, **7**(17), 1–6. doi:10.5334/jors.219.

Therneau TM (2021). ***survival**: Survival Analysis*. R package version 3.2-13, URL https://CRAN.R-project.org/package=survival.

Therneau TM, Grambsch PM (2000). *Modeling Survival Data: Extending the Cox Model*. Springer-Verlag, New York. doi:10.1007/978-1-4757-3294-8.

Thomas M, Bornkamp B (2017). "Comparing Approaches to Treatment Effect Estimation for Subgroups in Clinical Trials." *Statistics in Biopharmaceutical Research*, **9**(2), 160–171. doi:10.1080/19466315.2016.1251490.

Varadhan R, Wang SJ (2016). "Treatment Effect Heterogeneity for Univariate Subgroups in Clinical Trials: Shrinkage, Standardization, or Else." *Biometrical Journal*, **58**(1), 133–153. doi:10.1002/bimj.201400102.

Varadhan R, Yao W (2014). ***DSBayes**: Bayesian Subgroup Analysis in Clinical Trials*. R package version 1.1, URL https://CRAN.R-project.org/package=DSBayes.

Wager S, Hastie T, Efron B (2014). "Confidence Intervals for Random Forests: The Jackknife and the Infinitesimal Jackknife." *The Journal of Machine Learning Research*, **15**(1), 1625–1651.

Wang C, Louis TA, Henderson NC, Weiss CO, Varadhan R (2018). "**beanz**: An R Package for Bayesian Analysis of Heterogeneous Treatment Effects with a Graphical User Interface." *Journal of Statistical Software*, **85**(7), 1–31. doi:10.18637/jss.v085.i07.

Wickham H (2009). ***ggplot2**: Elegant Graphics for Data Analysis*. Springer-Verlag. doi:10.1007/978-0-387-98141-3. URL https://ggplot2.tidyverse.org.

Affiliation:

Nicolas M. Ballarini
Section for Medical Statistics
Center for Medical Statistics, Informatics, and Intelligent Systems
Medical University of Vienna
1090 Vienna, Austria
E-mail: nicoballarini@gmail.com
URL: https://cemsiis.meduniwien.ac.at/en/ms/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
October 2021, Volume 99, Issue 14 *Submitted:* 2019-01-11
doi:10.18637/jss.v099.i14 *Accepted:* 2021-02-10