**Guidance for Stratified Medicine Analysis**

Draft

2023

**Table of Contents**

[**Section 1:** **Background** 3](#_Toc125703732)

[**Section 2:** **When/Why to Perform Stratified Medicine Analysis** 4](#_Toc125703733)

[**Section 3:** **Statistical Methods** 7](#_Toc125703734)

[**3.1: Overview** 7](#_Toc125703735)

[**3.2: Heterogeneous Treatment Effect (HTE) Stratification** 10](#_Toc125703736)

[**3.3: Risk Stratification** 11](#_Toc125703737)

[**3.4: Treatment Effect Estimation** 13](#_Toc125703738)

[**3.5: Interpretability** 13](#_Toc125703739)

[**3.6: Subgroup Pooling** 15](#_Toc125703740)

[**3.7: Confirmation** 16](#_Toc125703741)

[**3.8: Model Selection** 16](#_Toc125703742)

[**Section 4:** **Discussion** 17](#_Toc125703743)

[**Section 5:** **References** 17](#_Toc125703744)

[**Section 6:** **Appendix** 24](#_Toc125703745)

[**6.1: Methods for Heterogeneous Treatment Effect (HTE) Stratification** 24](#_Toc125703746)

[**6.2: Methods for Risk Stratification** 33](#_Toc125703747)

[**6.3: Simulation Studies** 34](#_Toc125703748)

# **Background**

With recent advances in technology, the general topic of personalized/precision/stratified medicine, which uses patient-level characteristics to identify subgroups of patients who most likely benefit (or do not benefit) from a treatment, has attracted much attention in the pharmaceutical industry. These subgroups are usually defined based on values of demographic, clinical, biomarker, and/or other covariates that can be measured at baseline. Despite advances such as enhanced patient-level data collection and machine learning (ML) based models geared towards detecting heterogeneous treatment effects (HTE), there are a variety of factors to consider when assessing whether to perform subgroup analyses for clinical development. For example, there are legitimate concerns about subgroup analyses that should not be disregarded, such as multiplicity, confounding, small sample sizes, and clinical or biological implausibility. However, well-planned, and well-executed subgroup analyses can play a valuable role in addressing the needs of multiple stakeholders.

The high-level goal of subgroup analyses or stratified medicine analyses is to assess prognostic factors and/or treatment effect heterogeneity across subgroups of patients based on patient-level characteristics. Depending on the context, these subgroups may be pre-specified (ex: BMX+ vs BMX-) or these subgroups may be identified through a pre-specified algorithm or model. For the former, subgroup analyses are typically performed by analyzing an endpoint one-factor or one-variable at a time, such as **S =** [BMX+, BMX-]. A common graphical representation of such analyses is the “forest plot” which illustrates subgroup-level treatment effects across the set of pre-specified subgroups. For the latter, subgroups are identified through a data-driven algorithm, for example tree-based. This approach can generally yield subgroups defined by multiple factors such as **S =** [BMX+, Age>65 & BMX-, Age65 & BMX-], which may better represent potential treatment response differences. See Section 3 for methodological details.

For example, suppose subgroup (pre-specified or not) analyses conducted in a Phase II study identify a subset of the patient population that shows no evidence of benefitting from the study treatment and there is clear biological rationale for this finding. Pursuing a label or future study based on the likely benefitting patients can yield a stronger treatment effect estimates and subsequently aid with product differentiation. Similarly, by pursuing a Phase III study that only includes patients that likely benefit, the probability of success will also increase. An obvious disadvantage is that this strategy could reduce the size of the target population or lead to spurious elimination of a subgroup that actually has benefit. However, by including patients that do not benefit from the study treatment, the estimated treatment effect will be smaller, and it may be more difficult to demonstrate our product’s added value and/or differentiate from related competitor products.

Alternatively, principled subgroup analyses can aid in rebutting external claims arising from unstructured subgroup analyses, as well as show comparably favorable benefit/risk across a broad spectrum of pre-specified subgroups. The increased volume of high-quality real-world data (RWD) with detailed patient-level information only increases the risk that external parties (ex: regulators, insurers, researchers) will assess the efficacy of our products in real-world setting and subsequently claim that specific sub-populations show negligible benefit, which would undermine efforts to show the value of the product. This risk is likely to only escalate as advances in technology further improve the ability to capture high-quality patient-level information in real-world settings.

Notably, there are a variety of considerations beyond statistical issues when considering stratified medicine analyses for clinical development, and multiple stakeholders should be involved in decision making. While agreement on key scientific principles is relatively clear (for example the treatment should benefit patients) alignment of different perspectives on the why/what/when/how of stratified medicine analyses needs work and clarity. This guidance aims to broadly provide the following: when/why to perform stratified medicine analyses, along with details on related statistical methodologies.

# **When/Why to Perform Stratified Medicine Analysis**

The application and considerations for stratified medicine analyses vary across the different phases of clinical development, along with the type of design employed to address scientific questions of interest. See **Tables 1-2** for details on *Phase 1-2* and *Phase 3* respectively. Of note, while these analyses can be implemented at various stages of clinical development, conducting these analyses at the completion of a Phase 2 trial may have the greatest impact on the clinical development as any findings from Phase 2 can be directly incorporated into Phase 3 design considerations. In contrast, the impact may be limited for ad-hoc subgroup analyses at the end of Phase 3.

**Table 1** describes considerations for Phase 1-2 studies, including Phase 1b/2a POC and Phase 2 Dose Ranging. For all, the overall goal of these analyses is to inform future studies, identify prognostic and/or predictive variables, and aid in Go / No-Go decisions. The potential action items largely relate to future Phase 3 designs to increase the probability of success, and the action taken largely depends on the level of evidence. For example, if an identified subgroup has an overwhelmingly strong treatment effect estimate and there is clear biologic rationale, the subsequent Phase 3 strategy may be to enroll the identified subgroup population. Alternatively, if the totality of evidence is less convincing, a dual-primary objectives strategy (all comers and identified subgroup) may be more appropriate. Other options include using the identified subgroups as a stratification factor, or pre-specifying the identified subgroups within the Phase 3 subgroup analysis plan.

A common property across these studies is the generally small sample size, which ultimately means there is relatively lower power to detect meaningful subgroups and subgroup analyses could be viewed as exploratory. For Phase 2 dose ranging studies, predictive factors relate to the doses themselves and there may be an opportunity to leverage the dose response relationship to improve the likelihood of identifying meaningful subgroups.

**Table 1: Stratified Medicine Analyses in Phase 1/2 Studies**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stage** | **Properties** | **Purpose** | **Potential Action Items** | **Considerations** |
| **Ph1b/2a POC** | RCT with potentially small sample size, possibly crossover design | **Inform:**   * Identify potentially prognostic and/or predictive variables * Go / No-Go * Future designs | Design Ph3 accordingly   * Enroll BMx+ only * Dual-primary objectives (all-comers and BMx+) * Stratification factor * Adaptive design? * Accelerated approval? * Pre-specify Ph3 subgroup analyses. * Align with other stakeholders  (HTA, commercial, inform late development plan) | * Small sample size; low power * Subgroups with high response rates could be expanded to confirm results * Crossover study requires repeated measures methods. |
| **Ph2 Dose Ranging** | Multiple doses | * Treatment has multiple dose levels * Consider dose-response relationship in the analysis to increase efficiency of model |

**Table 2** describes considerations of subgroup identification models for Phase 3 studies. In contrast to Phase 1 or 2 studies, applications of subgroup analyses are mostly supportive for Phase 3 registration / market access or to inform future programs. For example, forest plots based on a limited set of pre-specified subgroups can provide supportive evidence that the study treatment may benefit specific sub-populations of interest or has generally consistent efficacy in a broad population. Alternatively, pre-specified subgroup analyses that examine subgroups with respect to multiple factors can likewise provide supportive information on potential treatment heterogeneity. A more “aspirational” example is an adaptive design that can identify and confirm “Benefiting” subgroups within a sequential-like study. This, however, has numerous concerns, such as Type I error control that needs addressing through statistical methodology, questions about adequate surrogate endpoints for early decision making, and operational issues concerning making decisions in a timely fashion. Regardless, while the large Phase 3 sample size may increase the chance of detecting meaningful subgroups, there are considerable risks given the registrational intent. The intent and purpose of subgroup analyses should be clearly articulated and formulated within the broader clinical design and strategy.

**Table 2: Stratified Medicine Analyses in Phase 3 Studies**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stage** | **Properties** | **Purpose** | **Potential Action Items** | **Considerations** |
| **Ph3 RCT** | Large N, randomized | * Evaluate treatment effect based on pre-specified subgroups (forest plot) * To optimize cost-benefit/effectiveness ratio of a product to ensure competitive value for the right patient populations, e.g. HTA reimbursement process * **Aspirational:** Adaptive design to identify and confirm subgroups within a single study  (*prespecified algorithm in protocol as well as documentation of operational and operating characteristic implications during program review*) | * Address subgroup related questions from regulators * Inform market access strategy (ex: pricing and reimbursement) * Inform potential next steps  (ex: based on convincing subgroup finding from failed study) | * Uncertainty about regulatory acceptance/approval of subgroup analyses * How to control Type I error? * How credible are the identified subgroups and corresponding treatment effect estimates? |

To summarize, the main goal of subgroup or stratified medicine analyses during Phase 1-2 is to inform future study strategies and/or designs. For Phase 3, stratified medicine analyses can primarily be considered supportive for registration, along with the chance to leverage subgroup identification methods (see Section 3) to inform future studies. Lastly, we note that Phase 2 studies are likely the best time to perform subgroup identification methods that explicitly aim to partition patients into “benefitting” (should receive study treatment) and “non-benefitting” (should possibly not receive study treatment), as any finding in Phase 2 can be directly incorporated into a subsequent Phase 3 design or strategy, for example, subgroup-specific hypotheses in the Phase 3 design. Of note, this may require an additional confirmation before Phase 3, ideally from another independent dataset (external or internal).

# **Statistical Methods**

## **3.1: Overview**

Subgroup analyses can be divided into five distinct categories:

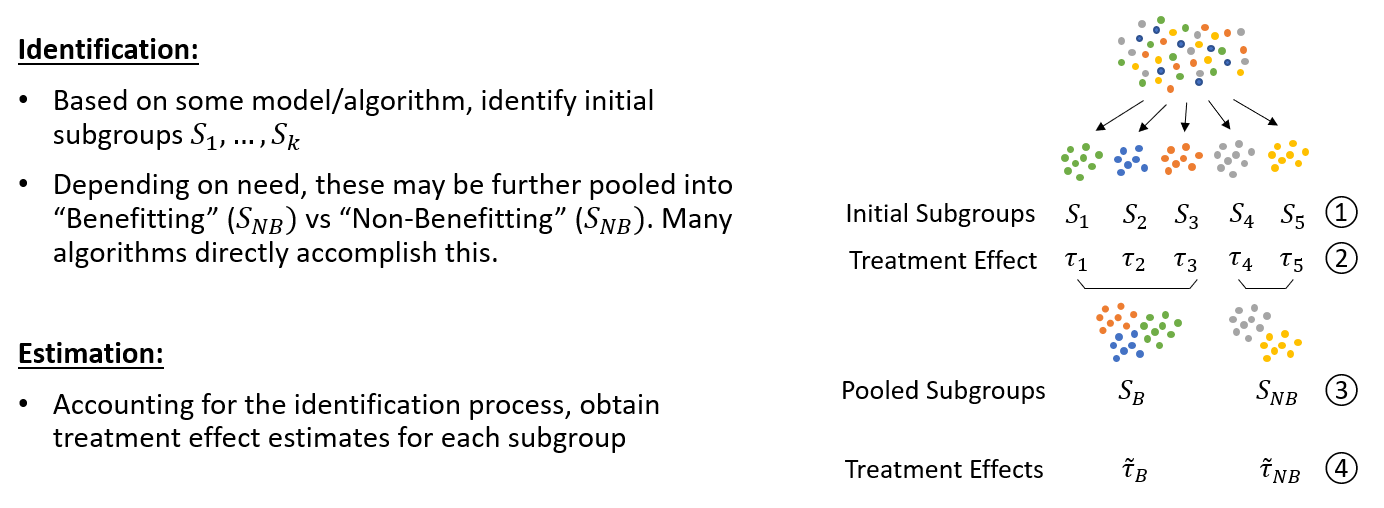
1. Pre-specified subgroup analyses for pre-stated hypotheses (e.g., biomarker+)
2. One-factor-at-a-time subgroup summaries (e.g., represented in a forest plot) for pre-stated factors
3. Analyses of post-hoc subgroups of interest (e.g., to address queries by HTA)
4. Identification of subgroups with prognostic effects (treatment-blinded risk stratification)
5. Identification of subgroups with predictive effects (heterogeneous treatment effects (HTE) stratification)

Note that (1)-(3) correspond to pre-specified subgroups or specific subgroups of interest, for example based on external or regulatory requests. In contrast, (4) and (5) do not pre-specify the subgroups of interest, and instead use some type of model or algorithm to identify subgroups that correspond to prognostic and/or predictive effects, and typically result in subgroups defined by multiple factors. While there are important statistical considerations for (1)-(3) such as multiplicity, this section focuses on (4) and (5) where the subgroups are *identified* through some data-driven model.

There are various challenges in applying subgroup identification methods in drug development. The first challenge is *interpretability*, especially for those so-called “black-box” methods such as random forest-based methods. Due to the sensitivity and regulations of stratified medicine applications, biomarkers and classification rules used for subgroup identification usually need to be clearly interpretable and ideally have underlying biological mechanism supporting such rules. Besides, *feasibility* of the classification rule also needs to be considered in practice, in terms of number of biomarkers required and assay complexity. The number of biomarkers used in such rules generally cannot be too large and the technology used for such assays should be easily accessible for such rules to be widely applied. As such, human intervention and interpretation might be involved during subgroup identification rule development. Another challenge in applying subgroup identification methods is *reproducibility*. The treatment effect in the identified subgroup and ideally the interaction effect between treatment and the subgroups should be confirmed in an independent dataset before being applied in practice. Such independent dataset could be a subsequent confirmatory study, or a validation dataset that was left out from an existing study. *Power and false-positive considerations* are also important aspects in subgroup identification. As it is usually valuable to have such subgroup identified early in drug development, analysis methods that achieve reasonable power, with proper false-positive rate control, with sample sizes commonly seen in Phase 2 studies are highly desirable. Moreover, *unbiased treatment effect estimate* of the identified subgroup is required for adequately design subsequent confirmatory studies or to accurately evaluate the clinical benefit in the identified subgroup. Given the fact that treatment effect is usually estimated in the same dataset that the subgroup identified, resampling approached is usually required to obtain unbiased estimate and more details will be discussed in Section 4.

See ***Figure 1*** for the general description of stratified medicine / subgroup identification analyses and subsequent assessment. Details on specific methods are described later. Ad-hoc illustrative examples can be found in Appendix 6.4.

**Figure 1: General Flow of Stratified Medicine / Subgroup Identification Analyses**



To summarize:

**Identification:** Based on some model (risk stratification or HTE stratification), the initial output is some set of subgroups where refers to some partition of the patient population (ex: BMX+ vs BMX-). Each subgroup could be distinct (ex: Virtual Twins) or the identified subgroups could overlap (ex: SIDES). Many models/algorithms (ex: optimal treatment regime methods) also directly determine which subgroups likely benefit from treatment (Benefitting, ) and which subgroups likely do not benefit (Non-Benefitting, ).

**Subgroup Pooling:** For many HTE stratification methods, the identified subgroups do not directly assign which treatment is recommended. If so, the initially identified subgroups can further be pooled or classified to and . It may also be useful to classify select subgroups as “Uncertain” (not clear whether the subgroup benefits or not); Bayesian credible subgroups takes this approach [Schnell P, et al 2016]. See Section 3.5 for details on subgroup pooling.

A high-level summary of subgroup identification methods can be found at **Table 3,** with more details provided in **Appendix 6.1 and Appendix 6.2.**

**Step 2, Estimation:** The next step is to estimate the subgroup-specific treatment effects to determine whether the identified subgroups have varying treatment effects. For HTE stratification methods, which directly uses the treatment assignment to identify subgroups, some type of resampling is generally required to avoid overly optimistic estimates (See Section 3.4). For risk stratification methods which are blind to treatment assignment, subgroup-specific treatment effect estimates are expected to have low or negligible bias and resampling may not be needed. In general, if the subgroups are identified through some model or algorithm, it is recommended to use resampling-based methods such as bootstrapping or cross-validation for treatment effect estimates.

The follow sub-sections describe the various components of the subgroup identification process. Section 3.2 discusses identification methods for heterogeneous treatment effect (HTE) stratification, which directly searches for subgroups with high and/or low treatment effects. Section 3.3 discusses identification methods for risk stratification which searches for subgroups based on prognostic effects and is blind to treatment assignment. Section 3.4 discusses treatment effect estimation and different resampling approaches. Section 3.5 discusses subgroup pooling, which is relevant for HTE methods that aim to assign subgroups to “Benefitting” and “Non-Benefitting.” Section 3.6 discusses interpretability of the models. Section 3.7 discusses model selection.

**Table 3: Summary of Subgroup Identification Methods**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **Type: Risk or HTE** | **Endpoints**  (C, B, S)\* | **Subgroup Identification Process** | **Number of Subgroups (**) | **Code**  **(R package)** |
| Virtual Twins | HTE | C, B, S | Regress HTEs against tree/linear model to identify subgroups |  | ranger, rpart,  StratifiedMedicine |
| Generalized Random Forest (grf) | HTE | C, B, S | Regress HTEs against tree/linear model to identify subgroups |  | grf |
| ForestSearch | HTE | S\* | Outputs distinct subgroups |  | github |
| Personalized | HTE | C, B, S | Outputs distinct subgroups  (tree or linear) |  | personalized |
| Interaction Trees | HTE | C, B | Outputs distinct subgroups (tree) |  | quint |
| MOB | HTE | C, B, S | Outputs distinct subgroups (tree) |  | partykit, StratifiedMedicine |
| GUIDE | HTE | C, B, S | Outputs distinct subgroups (tree) |  | MrSGUIDE (C only), command line executable |
| One-Step XGBoost | HTE | S\* | Regress OTRs against tree/line model to identify subgroups |  | github |
| SIDES | HTE | C, B, S | Outputs overlapping subgroups |  | SIDES |
| 5-STAR | Risk | S\* | Outputs distinct subgroups (tree) |  | fiveSTAR |

**Note:** C=continuous outcome, B=binary outcome, S=survival outcome. S\*=Developed for survival, but principles should apply to C/B.

## **3.2: Heterogeneous Treatment Effect (HTE) Stratification**

HTE stratification methods directly search for subgroups that are likely predictive. In the seminal paper “Tutorial in Biostatistics: Data-driven subgroup identification and analysis in clinical trials” [Lipkovich I, Dmitrienko A, D’Agostino RB 2017], the authors describe four broad frameworks: (1) Global outcome modeling, (2) Global treatment effect modeling, (3) Optimal treatment regimes, and (4) Local modeling. Admittedly, while these four frameworks have overlapping concepts and certain methods may not neatly fall within this categorization, the overarching principles provide an important summary of general subgroup identification strategies.

Let denote the outcome, binary treatment indicator, and baseline variables respectively. The heterogeneous treatment effect (HTE) or individual treatment effect (ITE) is defined as where without loss of generality we assume that larger values correspond to larger treatment benefit. The optimal treatment is defined as where *c* is some clinical threshold of benefit (ex: 0). Depending on the context of the analysis, the threshold *c* can be either chosen to indicate a clinically meaningful benefit, e.g., *c* = 10, or it could refer to some threshold of “harm” where patients receive more benefit by taking the control treatment, e.g., *c* = -10. For example, ForestSearch takes the latter approach.

*Global outcome modeling* fits a model (or treatment specific models) to estimate the HTE. For example, the Virtual Twins method [Foster J, et al 2011] uses a random forest model with built-in treatment by variable interactions, . The fitted model is then used to predict treatment-specific outcomes, , as well as the estimated HTE . Notably, the random forest model is generally considered a “black-box” method and forming subgroups requires a second step that feeds the estimated ITEs into some type of interpretable model. The most common strategies are either tree-based models, ex: , or linear models, ex: . The tree-based approach forms distinct rule-based subgroups through recursive partitioning (ex: SNP+ vs SNP-), while the linear model approach outputs patient-level “scores” that can be interpreted via the estimated regression coefficients. For the latter, instead of fitting two models, another approach is “personalized” [Huling JD and Yu M 2021]. Here, A-learning or a weighting method is used to directly estimate the HTE given some pre-specified functional form [Chen S, Tian L, Cai T, Yu M 2017]. Overall, the main drawback with global outcome modeling is this relies on the estimated THE which can be noisy. If the HTE estimates are biased and/or do not adequately capture the true treatment-by-variable interactions, this could yield misleading results. Black-box models typically also require a second stage regression for interpretation.

*Global treatment effect modeling* forms subgroups by directly targeting the treatment by variable interactions. These methods are primarily tree-based. For instance, the interaction tree method extends the classic CART method by including a treatment-by-variable interaction split (Su X, et al 2008). Other tree-based methods include GUIDE [Loh WY and Zhou P 2020] and model-based partitioning (MOB) [Zeileis A, Hothorn T, and Hornik K 2008]. Both GUIDE and MOB have the flexibility to search for partitions corresponding to prognostic variables and/or predictive variables. The main advantage with these approaches is that there is no need to estimate the HTE, which heavily relies on modeling assumptions. The main disadvantage of these approaches is that they typically rely on a single tree, which can be unstable and tree-based rules could significantly change depending on the training data.

*Optimal treatment regimes (OTR)* focus on identifying the optimal treatment for each patient. This approach is closely connected to global outcome modeling as OTR methods generally often rely on HTE estimates such that . For example, outcome weighted learning (OWL) regresses with weights to obtain estimates [Zhao Y, et al 2012]. For subgroup identification this strategy can directly apply to the second stage of Virtual twins with as a *tree* function. In general, most OTR approaches rely on HTE estimates to obtain optimal treatment probabilities. A notable exception is the “One-Step XGboost” which directly estimates through gradient-tree boosting [Zhang P, et al 2020]. Interpretable subgroups are then obtained by regressing . As with global outcome modeling, the main disadvantage of OTR is the reliance on HTE estimates and/or the reliance of a second-stage regression for interpretation.

*Local modeling* searches for regions of the covariate space with improved treatment effects. Examples include SIDES, SIDESSCREEN, and PRIM [Lipkovich I, et al 2011, Abu-Hanna A, et al 2010]. This avoids estimating an outcome function across the entire variable space and instead focuses on specific regions. One advantage is that these approaches will give broad insight on potential subgroups. However, these subgroups may be overlapping, which could limit final interpretation.

For details on specific HTE stratification methods, see **Appendix 6.1**.

## **3.3: Risk Stratification**

Risk stratification methods directly search for subgroups that are likely prognostic, aiming to stratify patients into groups with similar levels of risk rather than identifying subgroups with differential treatment effects. While this approach focuses on finding key prognostic variables, it is also likely to identify predictive variables that also have a prognostic component. The key point is that these models do not include or are blind to treatment assignment when forming subgroups.

Risk modeling is typically done in two stages: in the first stage, a multivariate model for predicting risk is generated using either external data or trial data, ignoring treatment assignment. In the second stage, patients are stratified based on the findings from step 1, and the treatment effect within and across risk groups is evaluated. As with treatment-effect subgroup methods, the analysis plan should be fully specified before the data is examined. While it may seem intuitive to examine risk levels within just the control arm, this may introduce or exaggerate treatment-risk interactions due to differential fits between the treatment and control group and can bias treatment effect estimates across the risk levels (Kent et al. 2019). Risk stratification methods therefore rely on data pooled across both the control and treatment arms, blinded to treatment assignment information. By excluding treatment assignment information from the strata formation step, risk modeling approaches allow for covariate selection/strata formation and treatment effect estimation to be done using the same data without introducing bias. This idea is well supported in the literature (e.g., Tukey 1993, Edwards 1999, Pocock et al. 2002).

One major benefit of risk stratification is that it can be used to identify important prognostic covariates to adjust for when estimating overall treatment effect, as well as provide an approximate relationship between these variables and outcome via the formed strata. Appropriate adjustment for key prognostic variables may lead to reduced bias (survival traits) or improved precision (continuous/binary traits) when estimating overall treatment effect in a heterogeneous population, further leading to higher power to detect an overall treatment effect. Further, these approaches provide transparency of how an overall estimate is generated as well as characterization of treatment effect variability over risk levels.

The PATH statement (Kent et al. 2019) lists seven consensus criteria for when a risk-modeling approach may be most useful for an RCT:

1. Overall treatment effect is well established (reduces risk of false positive findings)
2. Benefit/harm of an intervention is well balanced (increases sensitivity)
3. Treatment is associated with nontrivial amount of serious harm/burden (increases importance of identification of correct patient population)
4. Several large RCTs are available for pooling into a meta-analysis (improves precision of within-strata effect estimates)
5. Substantial, identifiable heterogeneity of risk in trial population is anticipated (increases utility of risk-based modeling)
6. There is strong preliminary evidence that a prediction model is clinically useful for treatment selection (increases likelihood of clinical application of the prediction model)
7. Clinical variables in proposed models are routinely available in clinical care (increases likelihood of clinical application of the prediction model)

A thorough explanation of the motivation behind each of these is provided in the PATH Statement Explanation and Elaboration (Kent et al., 2020). The explanation document further suggests some approaches for risk stratification, including, mostly simply, grouping patients using quartiles and reporting treatment effect within each to understand the variability of effect over risk. More sophisticated methods for principled risk-based stratification of patients and overall treatment effect estimation also exist, using similar machine learning approaches as in HTE stratification. More details, for example the 5-STAR method, are provided in **Appendix 6.2**.

## **3.4: Treatment Effect Estimation**

For risk stratification methods, which by design are blind to treatment assignment, subgroup-level treatment effect can be estimated directly within each identified subgroup.

For HTE stratification methods, which directly uses the treatment assignment for identification, it is important to obtain unbiased treatment effect estimate within each identified subgroup. It is well known that naïve estimates which do not account for identification process are usually biased [Foster J, et al 2011]. The general recommendation is to use bootstrapping or cross-validation based procedures for improved treatment estimates and subsequent decision making.

For example, suppose HTE stratification methods are used and subgroups are identified. The “naïve” treatment estimate for some subgroup *k* can be obtained by:

where and are treatment specific sample sizes and is an indicator whether patient falls into subgroup *k*. In general, this estimate will be biased since the subgroup identification model and treatment estimation both use the same data and further inference is generally not valid.

An alternative approach is to use bootstrap resampling. If the subgroups are not pre-defined groups (ex: tree outputs), the approach described by [Loh W, et al 2015] can be used. Here, the subgroup identification process is repeated for each bootstrap resample *r* with outputs bootstrap subgroups, which can differ from the original *K* subgroups, both in terms of rules and the patient groupings. The bootstrap subgroup treatment estimates are then mapped back to the original or observed subgroup assignments, such that:

where is the “naïve” treatment estimate for bootstrap subgroup and is the number of patients that are in both the original subgroup *k* and the bootstrap subgroup . Point-estimates and/or variability metrics can then be obtained directly from the bootstrap distribution .

If the subgroups are pre-defined groups (ex: Benefitting vs Non-Benefitting), the above approach can be used as well as the approach described in [Foster J, et al 2011], which boils down to setting for each resample. This only works because the subgroup model always outputs the same “label” (ex: Benefitting vs Non-Benefitting). Other methods such as bootstrap bias correction can also be used, which was also evaluated by Foster J et al 2011. Other models such as Personalized and ForestSearch also use a bootstrap bias correction for treatment effect estimation.

Other newer methods may also be promising, such as the method described by Guo X and He X 2021, which uses a modified bootstrap resampling scheme (de-biasing bootstrap) that does not require re-fitting the subgroup identification model at each bootstrap resampling. This is computationally advantageous, and this paper likewise derives some asymptotic properties of their method. A related paper by Zhao B, Ivanova A, and Fine J 2023 further explores this method along with other proposals, and while the “de-biasing” bootstrap proposal yields reasonable properties (reduced bias, better coverage), there were still simulation scenarios where no approach maintained the desired 95% coverage of subgroup-specific treatment estimates.

To illustrate the importance of resampling-based treatment estimates, see Appendix 6.3.1 for detailed simulations. In general, compared to “naïve’ estimates, bootstrap resampling reduces bias and yields coverage closer to the nominal 95%. The “naïve” treatment estimates tend to be overly optimistic/biased and have poor coverage properties.

## **3.5: Subgroup Pooling**

Given a set of initial subgroups , there may be a need to classify which subgroups likely benefit from the study treatment () vs those that likely do not benefit (). For example, based on a Phase II subgroup analysis, if there is strong evidence that a subset of patients does not respond to the study treatment as well as a clear biological rationale for the variables defining the identified subgroups, the study team may decide to omit non-benefitting patients from a subsequent study. If this is the case, tree-based methods such as Virtual Twins, MOB and GUIDE (among others) would require an additional pooling step. It is also possible that all subgroups likely benefit (e.g., is empty) or vice versa.

This guidance currently describes two general approaches for subgroup pooling, see Table 4. Let , , and denote the treatment effect estimate, lower CI (one-sided 1- %), and OTR probability for subgroup where *c* refers to the clinically meaningful threshold (ex: 0). Let *p* refer to the OTR probability threshold (ex: 0.50).

The treatment effect point estimate approach is intuitive and straight-forward: “Benefitting” correspond to subgroups with an estimated treatment effect that is clinically meaningful or within a certain bound of benefit, otherwise subgroups are considered “Non-Benefitting.” A variation on this approach is to use the CI for subgroup assignment. This is generally more conservative and needs consideration on the alpha level to choose. Both approaches likely require resampling methods to avoid making decisions based on overly optimistic treatment effect estimates (See Section 3.4).

The OTR probability approach directly uses the subgroup specific OTR probabilities to assign subgroups to “Benefitting” and “Non-Benefitting.” While it is common to set the probability threshold to 0.5, this may not be optimal depending on the data and the analyst may need to investigate other probability thresholds (ex: youden index).

In general, for non-OTR methods (ex: GUIDE), we recommend the treatment effect point estimate approach with resampling-based treatment estimates (See Section 3.4). For OTR methods (ex: One-Step XGBoost), the OTR probability approach is recommended. For detailed simulation results, see Appendix 6.3.1.

**Table 4: Subgroup Pooling Approaches**

|  |  |  |  |
| --- | --- | --- | --- |
| **Approach** | **Benefitting:** | **Non-Benefitting:** | **Considerations** |
| Treatment Effect Estimate |  |  | * Treatment estimates likely require resampling methods |
| OTR Probability |  |  | * May be sensitive to HTE and/or OTR estimates. * How to select probability cutoff *p*? |

## **3.6: Interpretability**

For practical implementation in clinical development, the method or model chosen should ultimately be interpretable. For example, can the discovered subgroups be readily incorporated into a future Phase 3 study (ex: as a stratification factor), or can the subgroups be clearly defined in a label (ex: via simple text). While the definition of an “interpretable model” is admittedly broad, the analyst should consider whether their model can be translated into clinical practice and/or be communicated clearly to clinical or other stakeholders. Notably, “black-box” models such as random forest or neural networks are clearly not interpretable without some type of post-processing. Beyond the model itself, there needs to be careful consideration on the types of variables that could potentially define the identified subgroups. Lastly, beyond the model itself, is there a clear interpretation based on the variables that define the identified subgroups? Is there biological rational and clinical relevance? The variables selected for analysis should be carefully considered and it is recommended that the analyst discuss potential variables with related stakeholders such as clinical.

Overall, there are at least three broad modeling approaches that can be viewed as interpretable: (1) Linear models, (2) Tree models, and (3) Rule-based models. Regardless of the model, it is also recommended that the final interpretable model is trained or fit using the “important” variables, which we denote as . While there are many ways to select the variable set , the main goal is to filter out “noise” variables that are unrelated to the clinical outcome of interest. Potential filter or screening approaches include elastic net, random forest variable importance, permutation importance, Shapley values, and univariate p-values.

**Linear Models**

The end-product of a linear model is some type of “risk score” based on a linear combination of variables. For example, regress the estimated HTE against :

where is the estimated risk score. Given a pre-specified clinically meaningful threshold , patients benefit from the treatment if . Similarly, the analyst could instead regress the estimated optimal treatment probabilities against . While the previous examples comprise of two stages, which could lead to error propagation, methods such as A-learning can directly estimate the risk score in one stage (see: personalized). For all examples, the end-product is an interpretable linear model that describes treatment heterogeneity on some subset of important variables.

**Tree Models**

The end-product of a tree model is some type of partitioned variable space, for . For example, there could be three partitions / subgroups: = [Age>65 and BMX+], = [Age65 and BMX+], and =[BMX-].

As with the linear approach, one option is to regress the individual treatment effect (or optimal treatment probabilities) against V:

An example of this is Virtual Twins. Alternatively, there are numerous tree methods that directly search for partitions relating to treatment heterogeneity. Examples include MOB, GUIDE, and Interaction Trees. This approach effectively fits:

In either case, the end-product is a distinct set of rules that describes treatment heterogeneity on some subset of important variables.

**Rule-Based**

This approach is analogous to tree models, as the end-product is a list of interpretable rules based on simple IF-THEN statements. For example, the “listdtr” R package (Zhang et al 2018) could assign optimal treatment assignments as follows:

If Age>65 and BMX+, then treat with A

Else if Age and BMX+, then treat with A

Else if BMX-, then treat with B

ForestSearch similarly outputs distinct rule-based subgroups. Alternatively, the SIDES method outputs a set of potentially overlapping rule-based subgroups. If the goal is to determine which types of patients should receive treatment A vs B, the analyst will need to consider a reduced set of the SIDES-discovered subgroups.

## **3.7: Model Selection**

It may also be useful to fit several subgroup models and select the “best” fitting model to avoid dependency on a single model. Given a collection of subgroup models, with corresponding subgroup assignments for each *q*, the “best” model minimizes some loss function *L*:

where is the estimated loss based on the subgroups corresponding to model and is the model index that has the lowest loss function. Selecting the appropriate loss function depends on the problem at hand.

**Setting 1 (Risk Stratification):** If the candidate models identify subgroups using risk stratification (e.g., treatment blinded), standard loss functions such as MSE or AUC could be used. For example, the MSE loss is defined as:

where is the estimated mean for subgroup from . This is identical to fitting an OLS model with if is coded as indicator columns for each subgroup.

**Setting 2 (HTE Stratification):** If the candidate models are directly searching for heterogeneous treatment effects, standard MSE loss (for example) are not sufficient as they do not directly relate to the unobserved individual treatment effect, . The appropriate loss function further depends on whether subgroups are assigned to “Benefitting” or “Non-Benefitting” subgroups (See Section 3.5).

**Without Pooling:**

Here, subgroups are identified with potentially heterogeneous treatment effects, but no further classification to “Benefitting” or “Non-Benefitting” is done. Potential loss functions include A-learning loss (Chen et al 2017) and R-learning loss (Nie X and Wager S 2021). Both approaches use as the feature space (with indicators for each subgroup) and an adjustment via propensity score estimates, . R-learning loss functions include an additional adjustment via outcome model estimates, . For either, smaller loss values should ideally correspond to a more representative model of the underlying treatment heterogeneity. For RCTs, ) by definition.

For example, A-learning MSE loss can be estimated by fitting the OLS model with weights of with recoded to 1 (treatment) vs -1 (control).

**With Pooling:**

Here, subgroups are identified with potentially heterogeneous treatment effects and are further classified to “Benefitting” or “Non-Benefitting”. Loss functions from A-learning and R-learning would apply here, but with reduced to two groups, and . An alternative approach is from the optimal treatment regime literature (Zhang B et al 2012, Zhao YQ et al 2019). The standard OTR maximizes the following value function:

defined as the average outcome if everyone was treated according to their “optimal” treatment. At the subgroup level, we can instead use , or the value function defined by the subgroup partitions. For simplicity, we omit the subscripts and use *d* below.

Other variations of this value function have also been proposed, see **Table 5** below for when the study is a RCT with 1:1 randomization and the outcome (Y) is either continuous or binary. Notably, the surrogate value approach assigns higher values if there are larger differences between the group receiving their optimal treatment and those that did not receive their optimal treatment (Zhang P et al 2020). The model-based approaches all require some estimate of . Loss functions for survival have a similar flavor, although there is additional complexity due to the censoring mechanism.

**Table 5: Optimal Treatment Regime Value Functions (Continuous or Binary Outcomes)**

|  |  |  |
| --- | --- | --- |
|  | **Value Function** | **Surrogate Value Function** |
| **Plug-in** |  |  |
| **Model-based** |  |  |
| **Augmented** |  |  |

**Note:** For additional details, see Zhang B et al 2012 and Zhao YQ et al 2019.

Detailed simulation results for optimal treatment value functions can be found in the Appendix (Section 6.3.3). In general, compared to the individual subgroup models, the model selection yielded reasonably performance in terms of treatment classification metrics and selecting the truly predictive variables. The simulation results further suggest using the Plug-in version for either the “vanilla” or surrogate value functions, or the model-based approach if there is relatively high confidence that the outcome model is correctly specified. For the survival simulations, the model-based performed worse than the plug-in, likely due to more model misspecification. At higher sample sizes (ex: N>1000), the augmented should theoretically outperform the plug-in and model-based approaches.

# **Discussion**

The main goal of subgroup analyses or stratified medicine analyses is to assess patient-level heterogeneity, based on either prognostic factors (risk stratification) and/or predictive factors (HTE stratification). The application and subsequent action items of these analyses largely depends on the phase of the trial and more generally, there are a variety of considerations beyond statistical issues (ex: multiplicity, small sample sizes, biological implausibility) and multiple stakeholders should be involved in decision making. Regardless, well-planned and well-executed subgroup analyses can provide valuable context and address the needs of multiple stakeholders.

For Phase 1-2, the goal is primarily to inform future study strategies and/or designs. For Phase 3, subgroup analyses can primarily be considered supportive for registration, or subgroup-specific hypotheses could be explicitly included. Further, Phase 2 is likely the best time to explicitly aim to partition patients into “benefitting” and “non-benefitting” groups, as any finding can directly be incorporated into the Phase 3 design. However, given the relatively small sample size of Phase 2 and the potential for spurious findings, the results should generally be considered descriptive and further confirmation is likely needed, although the totality of data should be assessed. For example, if there is no external information that supports the results and strong biological rationale, the results could be considered hypothesis generating. Alternatively, if there is strong external information and strong biological rationale, this may support further investigation.

From a statistical and modeling viewpoint, it is difficult to provide a definitive model recommendation for risk stratification or HTE stratification, as performance can vary depending on the simulation setup and new methods are constantly being proposed. Depending on the endpoint, various models can be examined, and model selection methods can be applied to determine the final model. In addition, for clinical development, the model should be interpretable, such as linear, tree, or rule-based models. Simulations can also be run to evaluate model performance under data generating conditions like the trial of interest.

Consistent with the broader literature, our simulation findings illustrate that naïve treatment effect estimates for subgroups that are adaptively identified are overly optimistic and biased. Bootstrap resampling methods reduced the bias and resulted in improved coverage. The key point is that estimation of treatment effects needs to account for the subgroup identification process, which is accomplished through resampling. Improved subgroup-specific treatment estimates will be key in the long run and external/internal research will lead to progress.

A broad challenge is type I error control with respect to the subgroup identification. If in fact all patients benefitted from the treatment, then ideally the subgroup identification would only incorrectly identify a “non-benefitting” group 5% of the time (for . Most subgroup identification models do not directly provide say a p-value on whether subgroups are truly different, and it is important to assess subgroup-level treatment effect estimates (with resampling) which can provide context on whether there are heterogeneous treatment effects. Permutation approaches have been proposed (Foster et al 2016), which could also help mitigate type I error concerns and needs more investigation.

Overall, there are many challenges and considerations for subgroup analyses, especially those that are identified through some model or algorithm. While the focus of this guidance was on RCTs with a binary treatment and a single outcome, the principles discussed here apply to more complex settings such as benefit/risk, observational data, and multiple doses.

# **References**

Schnell P, Tang Q, Offen W, and Carlin B. A Bayesian Credible Subgroups Approach to Identifying Patient Subgroups with Positive Treatment Effects. *Biometrics 2016;* **72:**1026-1036

Lipkovich I, Dmitrienko A, D’Agostino RB Sr. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. *Statistics in Medicine* 2017; **36**:136–196.

Foster J, Taylor JMG, Ruberg SJ. Subgroup identification from randomized clinical trial data. *Statistics in Medicine* 2011; **30**

Huling JD and Yu M. Subgroup Identification using the personalized package. *Journal of Statistical Software* 2021; **98**(5): 1-60

Chen S, Tian L, Cai T, Yu M. A General Statistical Framework for Subgroup Identification and Comparative Treatment Scoring. *Biometrics 2017;* 73(4): 1199-1209.

Su X, Zhou T, Yan X, Fan J, and Yang S. Interaction Trees with Censored Survival Data. *International Journal of Biostatistics* 2008; **4(1):** 2

Zhao Y, Zheng D, Rush AJ, and Kosorok MR. Estimating Individualized Treatment Rules Using Outcome Weighted Learning. *Journal of the American Statistical Association* 2012; **107(44):** 1106-1118

Zhang P, Ma J, Chen X, and Shentu Y. A Nonparametric Method for Value Function Guided Subgroup Identification via Gradient Tree Boosting for Censored Survival Data. *Statistics in Medicine 2020.*

Loh WY and Zhou P. The GUIDE Approach to Subgroup Identification. (book, Chapter 6)

Zeileis A, Hothorn T, and Hornik K. Model-based Recursive Partitioning. *Journal of Computational and Graphical Statistics* 2008

Lipkovich I, Dmitrienko A, Denne J, Enas G. Subgroup identification based on differential effect search (SIDES): a recursive partitioning method for establishing response to treatment in patient subpopulations. *Statistics in Medicine* 2011; **30**:2601–2621.

Lipkovich I, Dmitrienko A. Biomarker identification in clinical trials. In *Clinical and Statistical Considerations in Personalized Medicine*, Carini C, Chang M (eds). Chapman and Hall/CRC Press: New York, 2014; 211–264.

Lipkovich I, Dmitrienko A, Patra K, Ratitch B, Pulkstenis E. Subgroup Identification in Clinical Trials by Stochastic SIDEScreen Methods, *Journal of Biopharmaceutical Statistics* 2017; **9**:368–378.

Lipkovich I, Dmitrienko A. Strategies for identifying predictive biomarkers and subgroups with enhanced treatment effect in clinical trials using SIDES. *Journal of Biopharmaceutical Statistics* 2014; **24**:130–153.

Dmitrienko A, Lipkovich I, Hopkins A, Li YP, Wang W. Biomarker Evaluation and subgroup identification in a Pneumonia Development Program using SIDES. In Applied Statistics in Biomedicine and Clinical Trials Design, Chen Z, Liu A, Qu Y, Tang L, Ting N, Tsong Y (eds). Springer: New York, 2015; 427–466.

Hardin DS, Rohwer RD, Curtis BH, Zagar A, Chen L, Boye KS, Jiang HH, Lipkovich IA. Understanding heterogeneity in response to antidiabetes treatment: a post hoc analysis using SIDES, a subgroup identification algorithm. *Journal of Diabetes Science and Technology* 2013; **7**:420–429.

Lipkovich I, Dmitrienko A, D’Agostino RB Sr. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. *Statistics in Medicine* 2017; **36**:136–196.

Nie X, Wager S. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika* 2021; **108**(2): 299-319.

Zhao YQ, Laber E, Ning Y, Saha S, Sands B. Efficient augmentation and relaxation learning for individualized treatment rules using observational data. *J Mach Learn Res*. 2019; 20 (48)

Zhang B, Tsiatis A, Laber E, Davidian M. A Robust Method for Estimating Optimal Treatment Regimes. *Biometrics 2012;* 68(4): 1010-1018

Guo X, He X. Inference on selected subgroups in clinical trials. *J Am Stat Assoc* 2021; 116: 1498-1506

Zhao B, Ivanova A, Fine J. Inference on subgroups identified based on heterogeneous treatment effect in a post hoc analysis of a clinical trial. *Clinical Trials* 2023.

Foster JC, Nan B, Shen L, Kaciroti N, Taylor JMG. Permutation testing for treatment-covariate interactions and subgroup identification. *Stat Biosci* 2016; 8(1): 77-98.

# **Appendix**

## **6.1: Methods for Heterogeneous Treatment Effect (HTE) Stratification**

**Virtual Twins**

Virtual Twins (VT) (Foster J, et. al 2011) is a two-stage procedure for subgroup identification. At the first stage, the individual treatment effect ITE is estimated for each subject. This is usually done by fitting a regression model and then the ITE estimate for each subject is constructed as . To estimate random forests was proposed by Foster et. al., although other methods such as boosting or causal forest can also be applied. At the second stage, the individual ITE estimates are used as the response variable to fit a regression or classification tree to construct subgroups. The goal of this stage is to identify subgroups with differential treatment effects (i.e. difference between the two treatment options) and then identify a subgroup where the average treatment effect is greater than a pre-specified clinically meaningful threshold . At this stage, the ITE estimates can either be used as its original estimate or dichotomized into a binary variable representing treatment assignment. For continuous ITE estimates, a regression tree is fit in the second stage and for dichotomized ITE estimates a classification tree is fit in the second stage. Of note, virtual twins is not limited to continuous or binary outcome and it can also be applied to survival outcomes as long as the ITE can be defined and estimated, for example, through the RMST approach. In addition, although the CART method was proposed to be used in the second stage in the original VT method, the method does not limit the use of other methods in the second stage, such as ctree.

As VT requires estimation of the ITEs in the first stage, its performance heavily relies on the model fit in the first stage and may suffer from error propagation between the two stages. In addition, in order to classify patients into two distinct subgroups with respect to treatment assignment, unbiased estimate of the subgroup level treatment effect is also required, which is further discussed in Section 3.4.

VT can be easily implemented via existing R packages such as ranger, xgboost, rpart, and partykit. As an extended version of VT, various approaches can be applied in both stages and their performance are to be evaluated.

**Personalized**

The “personalized” R package identifies subgroups by directly estimating the ITE through a weighting or A-learning as described by [Chen S, Tian L, Cai T, Yu M 2017]. Both weighting and A-learning methods do not require specification of the full outcome regression model, and instead focus on direct estimation of . Subgroups are then formed by classifying patients with to “Responders” and patients with to “Non-Responders”. Subgroup-specific treatment effects are then estimated using a bootstrap bias correction approach. See [Huling JD and Yu M 2021] for more details.

The weighting method estimates by minimizing the following objective function with respect to some function :

where M is a convex loss function (ex: squared error loss) and is the propensity score function (for RCTs, this is constant, ex: 0.5).

A-learning involves minimizing:

In contrast to weighting, A-learning regresses the outcome against the transformed variable space which directly correspond to the treatment by variable interaction effects.

For both, a modeling choice must be made for and . A simple option is to assume some type of interpretable linear model. Linear models, with or without penalized regression, are currently available in the “personalized” R package. Other more flexible models such as random forest could also be fit using these broad frameworks. Regardless of the model, patients are stratified based on their estimated ITE. For example, responders correspond to , while non-responders correspond to .

The “personalized” R package can been found in the CRAN repository (<https://cran.r-project.org/web/packages/personalized/index.html>) and can accommodate a variety of outcomes, along with multiple treatments.

**Interaction Trees**

The Interaction Tree (IT) method is data-driven tree procedure that explores the heterogeneity structure of the treatment effect across subgroups that are objectively defined in a post hoc manner. Implementation of IT is comprised of three major components (1) growing a large initial tree; (2) a pruning algorithm; and (3) selection of the best tree size. Once a final tree structure is obtained, the subgroups are naturally induced by its terminal nodes.

Growing the initial tree starts with a single split of the data, which divides the study samples into the left child node (tL) and the right child node (tR). For a given node *t*, a split *s* yields the following 2X2 table.

|  |  |  |
| --- | --- | --- |
| Treatment | Child node | |
| tL | tR |
| 1 | , ,, | , ,, |
| 0 | , ,, | , ,, |

{, ,,} are the population mean, the sample mean, the sample variance, and the sample

size for the treatment 1 group in the left child node tL, respectively. Similar notation applies to the other quantities in the table. An intuitive measure for assessing the interaction is then given by:

Where is the pooled estimator of the subgroup variance. The splitting statistics, *G*(*s*), is defined as *G*(*s*) = *t*2(*s*).

Pruning of the initial large tree *T*0 , can be accomplished using an interaction-complexity pruning procedure, which is analogous to the split-complexity pruning algorithm used by LeBlanc and Crowley, resulting in a sequence of optimally pruned subtrees by iteratively truncating the “weakest link” of *T*0 . In the last step, the subtree with the maximum interaction-complexity measure (i.e. the complexity adjusted total interaction that a subtree contains), is selected as the final tree.

In the context of IT, VIM (Variable Importance Measure) for each covariate can be easily obtained as the magnitude of the total-interactions (sum of the G statistics as described above) lost from permuting the values of this covariate in the out-of-bag samples.

Recognizing that the generic IT framework does not allow the users to specify the type of interactions detected by the tree, which may be a significant drawback if qualitative interactions are of the central interest (in the presence of strong quantitative interaction effects, qualitative interactions may remain undetected), the method QUINT (Qualitative INteraction Trees) was proposed recently to specifically induce subgroups involved in ***qualitative*** interactions from RCT data.

QUINT can been found in the CRAN repository (<https://cran.r-project.org/web/packages/quint/>); but for the more generic IT implementation (which includes both qualitative and quantitative interactions), there is currently no R packages available, although the R functions are available upon request from the authors. Currently, both versions of IT do not support survival outcome.

**Model-Based Partitioning**

Model-based partitioning (MOB) is a tree-based method that uses recursive partitioning to form interpretable subgroups. The general idea is to partition the covariate space based on “parameter instabilities” from some regression model [Zeileis A, et al 2008; Seibold H, et al 2016]. For example, when the clinical outcome is continuous, MOB-OLS (ordinary least squares) first fits in the overall population. Next, across a set of candidate partitions (ex: BMX+ vs BMX-), MOB tests whether the parameter estimates exhibit “instability” across partitions. Parameter instability tests are performed using partial score functions and so-called “M-fluctuation” tests. For each candidate variable, MOB selects the partition with the highest parameter instability or lowest “M-fluctuation” test p-value. MOB will then select the variable with the lowest p-value, and if the Bonferonni multiplicity adjusted p-value passes a pre-specified threshold (ex: alpha=0.05), the corresponding partition is made. Within each partitioned group, the process repeats until no more partitions can be made or other stopping criteria are met.

Notably, if there are large changes in , the partition likely corresponds to a prognostic effect. If there are large changes in , the partition likely corresponds to a predictive effect. Alternatively, leveraging the fact that predictive variables are often prognostic, the analyst may instead search for subgroups that are predictive and/or prognostic (ex: look for large changes in and/or ). Lastly, MOB is primarily for subgroup identification and does not provide valid treatment effect estimates and/or variability metrics such as CIs. Further, if the goal is to identify “responders” vs “non-responders”, the tree-based subgroups will likely need additional pooling.

MOB can be implemented in R through the “partykit” package and associated functions “mob” (core function), “lmtree” (MOB-OLS), and “glmtree” (MOB-GLM). Key hyper-parameters include: (1) “alpha” (parameter instability p-value), (2) maxdepth (tree depth), (3) minsize (minimum tree node size), and (4) parm (which parameter instabilities to target). For user-specific regression models, the “mob” function should be used.

**GUIDE**

GUIDE (Generalized, Unbiased, Interaction Detection and Estimation; Loh WY and Zhou P 2020) is a flexible recursive partitioning algorithm for fitting classification and regression trees. The GUIDE algorithm is comprised of three main steps: split variable selection, node fitting, and pruning. For the ‘Gi’ method, which prioritizes predictive effects, splits are selected via a lack-of-fit interaction test within each node, comparing a reduced additive model to a fully interactive model. A separate ‘Gs’ method also exists to prioritize both prognostic and predictive effects through maximization of sum of chi-squared statistics from contingency tables. Splits are defined to be of the form to explicitly incorporate missing values instead of relying on prior imputation or surrogate splitting. This increases clarity in defined subgroups as each patient – even those with missing covariate information – is definitively classified into a stratum. GUIDE handles the large search space for covariate splits by discretizing ordinal covariates into a limited number of categories, defined by relevant sample quantiles. Similarly, for natively categorical variables, linear discriminant analysis is used to define splits on resulting canonical covariates, thus reducing the computational burden. GUIDE splits trees to maximal depth before pruning to a more ideal size using cross validation.

Within each tree node, GUIDE performs treatment effect estimation through a regression model (e.g., for continuous traits this could be a multiple linear, stepwise, or polynomial regression model, optionally adjusted for additional covariates), where missing values are imputed by node-level means. Resampling is built in to provide approximately unbiased treatment effect estimation and bootstrap-calibrated confidence intervals are provided for inference.

The GUIDE approach is very flexible, allowing for continuous, ordinal, nominal, and cyclical/periodic covariates, and continuous, discrete, longitudinal, and right censored responses. For survival outcomes, either Cox proportional hazard models assuming equal baseline hazards across all nodes (thus ensuring comparability between nodes) or RMST response models are employed. Ensemble extensions are also available, including GUIDE forest (which is similar to random forest but using GUIDE for split selection) and bagged GUIDE.

As it is a tree-based methodology, resulting subgroups may be considered reasonably interpretable, though this depends on tree size (prior to pooling down to two subgroups, which is not directly handled in GUIDE) as well as other factors, e.g., whether LDA is used to define subgroups as mentioned above. This can be dealt with in part by choice of covariate input format (e.g., whether covariates are pre-discretized to smaller/clinically interesting groups ahead of time).

GUIDE can be implemented using command-line software available via the main author’s website (<http://pages.stat.wisc.edu/~loh/guide.html>). The software can generate R code for subset selection based on the optimal generated model to use in a downstream workflow. Additionally, a limited R package MrSGUIDE is available from an external group that can be used for continuous response variables.

**One-Step XGBoost**

The one-Step boosting method for subgroup identification is a nonparametric method that connects a value function with the final subgroups identified. The value function is defined using a measure of subgroup identification which reflects the subgroup-treatment interaction. The goal is to identify the treatment performing subgroup and its complementary subgroup, treatment non-performing subgroup, so that the differential treatment effects weighted by the prevalence of subgroups, measured by the value function, will be maximized. Instead of estimating the outcome or treatment contrast or utilizing outcome weighting at the patient-specific level, the proposed method compares difference in RMST in subgroups constructed by subgroup membership scores of patients. The subgroup membership scores are the parameters to be searched so that the treatment-subgroup interaction will be maximized. Gradient tree boosting is proposed to search for the optimal subgroup membership scores. Different from a typical use of gradient tree boosting solving a supervised classification problem with individual loss function, the proposed value function does not involve individual label to evaluate misclassification cost for individual patients. But the value function is differentiable with respect to individual subgroup membership score and the gradient of the value function can be used in gradient tree boosting.

The value function is defined as:

where

are the probabilistic version of the Nelson-Aelen estimate for survival functions of each treatment arm in the treatment-performing and control performing subgroups.

The gradient tree boosting is an ensemble method that constructs a predictive model by additive expansions of decision trees. The final prediction follows an additive expansion of K base tree functions.

To learn the set of base tree functions, the following regularized objective function is minimized at each of the K iterations42.

Where is a differentiable loss function we want to minimize and is a penalty function that penalizes the complexity of the tree functions. First-order approximation can be used to optimize the objective function. We present the proposed value function estimator in (4), here we simply set the loss function , the negative value function.

We further define as a sigmoid function of , the prediction score from trees for patient .

Note that we do not define individual loss function, so our loss function is not a direct summation of individual loss. The loss function is differentiable with respect to , and the first gradient of the loss function can be used as the individual target label in the gradient tree boosting. The goal is to optimize to minimize the loss function and we can classify patients based on their estimated subgroup membership score . The numerical optimization can be implemented with commonly used software such as “xgboost”43.

An implementation of the boosting method is at <https://github.com/liupeng2117/SubgroupBoost/>. The R package may be used for subgroup identification based on continuous, binary and survival endpoints. For continuous and binary endpoints, the value function simply replaces the RMST estimate with weighted sample mean with weights given by the subgroup membership scores and **.** In addition, subgroup identification based on win difference and win ratio defined by a single or multiple efficacy/safety outcomes is also supported.

**SIDES**

The SIDES method utilizes recursive partitioning to perform a direct search for subgroups of patients who experience a treatment benefit. The recursive partitioning algorithm is applied to each individual candidate biomarker and an optimal split is found by maximizing a pre-defined differential effect criterion. The original SIDES procedure [Lipkovich I, et al 2011], referred as the SIDESbase procedure, has three steps – Initialize, Iterate and Finalize. It is worth noting that for the splitting criterion defined in the “Iterate” step, the most commonly used criterion is the differential splitting criterion while other types of splitting criteria can refer to [Lipkovich I, et al 2014; Lipkovich I, et al 2017]. Later, the SIDEScreen procedure (a two-stage subgroup search procedure) [Lipkovich I, Dmitrienko A 2014] was introduced into literature, which improves on the basic SIDES procedure by introducing a biomarker screen and includes the *fixed* SIDEScreen procedure and *adaptive* SIDEScreen procedure [Lipkovich I, et al 2014; Lipkovich I, Dmitrienko A 2014]. Note that, within the SIDEScreen procedure, when performing a multiplicity adjustment to control the probability of incorrect subgroup discovery, it is critical to account for both stages of the algorithm used in this procedure. Furthermore, Lipkovich et al extended the adaptive SIDEScreen and introduced the *stochastic* SIDEScreen method [Lipkovich I, et al 2017], the key idea of which is to introduce randomness in the subgroup generation process, borrowing from bagging methods, to produce a broader collection of subgroups. In general, the most important features of the SIDES method include the following:

* Complexity control (implemented in the “*Iterate*” step), which is applied to reduce the size of the search space and produce easy-to-interpret results, and lessen the multiplicity burden
* Multiplicity control (implemented in the “*Finalize*” step), which is employed to preserve the probability of incorrect subgroup identification
* Biomarker screen (a feature of SIDEScreen), which is introduced to filter out non-informative biomarkers

For these SIDES procedures, SIDESbase effectively deals only with relatively small sets of candidate biomarkers, and its performance tends to deteriorate in “massive” biomarker analysis problems with hundreds of baseline covariates. Performance loss is especially pronounced in settings where most of the candidate biomarkers are non-informative, which is quite common in real life applications. For SIDEScreen, it was specifically developed for the more challenging settings with large sets of pre-specified biomarkers. Particularly, the *stochastic* SIDEScreen method leads to a more reliable biomarker selection process, which is especially important for smaller, early phase studies when biomarker selection is typically carried out. For application of the SIDES method to drug development programs, please refer to [Dmietrienko A, et al 2015; Hardin DS, et al 2013] and the case studies in [Lipkovich I, Dmitrienko A, D’Agostino RB 2017]. Regarding to the program resource, the SIDESbase and SIDEScreen procedures were implemented in the R package RSIDES, and also by the main authors of SIDES on the website <http://biopharmnet.com/subgroup-analysis-software/>

**Forest Search (TO ADD DETAILS)**

**PRISM Framework**

PRISM (Patient Response Identifiers for Stratified Medicine) is a general framework that provides insight on treatment heterogeneity at the patient-level and subgroup-level, with resampling methods to provide more reliable subgroup-specific treatment effect estimates and variability metrics (ex: SE, CIs).

The procedure follows four steps:

* + - 1. Filter variables that are unlikely to be predictive and/or prognostic
      2. Estimate patient-level treatment effects (ex: ITE)
      3. Using tree-methods, form subgroups and estimate subgroup-specific treatment effects
         1. Option to pool initial subgroups into “Benefitting” vs “Non-benefitting”.
      4. Repeat Steps 1-3 using resampling for improved treatment effect estimation, SEs, and CIs.

PRISM can be implemented through the “StratifiedMedicine” R package with various defaults in place (<https://cran.r-project.org/web/packages/StratifiedMedicine/index.html>). For examples, if the outcome is continuous, the default procedure is: (1) Elastic Net Filter, (2) Random Forest based ITE predictions, (3) MOB-OLS, and (4) OLS for treatment effect estimation with bootstrap resampling. If applicable, the default “pooling” option is treatment effect pooling (See Section 3.6 for details).

Each step of PRISM is customizable, and the analyst can input custom models if necessary. For example, one could input their own filter model. Other features include variable importance plots, along with partial dependence plots to examine the functional relationships between variable(s) and the estimated ITE. PRISM is only implementable for treatments with two levels.

## **6.2: Methods for Risk Stratification**

**5-STAR Framework**

5-STAR (5-Step Stratified Testing and Amalgamation Routine) is an algorithm to perform risk-based stratification in a structured, treatment-blinded manner. Its focus differs from the treatment-based subgroup identification, focusing on securing an overall “win” in the trial population first and providing transparency with regards to treatment effect within formed strata second.

The strata formation within 5-STAR is similar to existing tree-based treatment subgroup methods. First, a list of all candidate covariates is generated through discussions within the research team; any covariate deemed plausibly important may be listed, including baseline demographics, prior medical history, and possibly routinely collected lab variables. Continuous covariates may be transformed, and all covariates are passed into the filtering stage (step 2). In this stage, covariates showing no evidence of association with the endpoint are filtered out. This is done blinded to treatment assignment via elastic net regularized regression or random forest variable importance measures. All remaining covariates are passed to step 3, where the conditional inference tree (CTREE) methodology is used to stratify patients into prognostic strata. First, in step 3A, an initial tree is formed using the raw filtered covariates as input; these initial strata are passed to step 3B where a final tree is formed using the ordered strata as input. The purpose of step 3B is to pool together any strata with sufficiently similar risk profiles. The key difference in approach vs. treatment-based subgroup analysis is in forming the strata: in 5-STAR, treatment is ignored when splitting the dataset such that, rather than strata being formed to optimize differential treatment effect, the strata segregate the population into a range of risk levels from lowest to highest.

After strata have been formed, treatment is unblinded and treatment effect is estimated by fitting a relevant model (e.g., Cox PH model or model-averaged AFT) separately in each formed risk stratum. This is unbiased as all strata formation steps were formed in a treatment-blinded manner. 5-STAR also provides an overall treatment effect estimate using weighted averaging over stratum-level estimates.

Some key advantages of 5-STAR include unbiased treatment estimation, overall treatment effect estimation and increase in power to secure a “win”, flexibility to handle multiple trait types, and general interpretability through its tree-based methodology. However, it has the limitations of finding “suboptimal” treatment-based subgroups when predictive covariates are not also prognostic (and in fact, does not guarantee any treatment effect heterogeneity across subgroups at all), and through the pooling step 3B can occasionally lead to rather complicated strata based on sets of “and” and “or” statements. Additionally, it doesn’t explicitly provide rules for patient treatment assignment, leaving room for subjectivity of which subjects should be treated, and can currently only be used for treatments with 2 levels (e.g., test vs control).

The 5-STAR algorithm is implemented in the fiveSTAR R package which is available on Github: <https://github.com/rmarceauwest/fiveSTAR>.

## **6.3: Simulation Studies**

We performed simulation studies to compare the performance of various subgroup identification methods, as described in this section.

## **6.3.1: Treatment Effect Estimation and Subgroup Pooling**

In these simulations, we have simulated binary endpoint (N=600) or survival endpoints (N=1000) of randomized control studies under different scenarios. Under the null hypothesis, a homogeneous treatment effect is assumed across all subjects, and under various alternative hypotheses, heterogeneous treatment effect is defined by 3 predictive variables.

In the simulations, four different existing subgroup methods were evaluated in Step 1 (Initial Subgroup Identification) of ***Figure 1***: MOB-OLS, interaction trees, virtual twins, and personalized. Two different approaches to estimate subgroup treatment effects were applied in Step 2: naive estimate and estimate via the bootstrap method. In the pooling step (Step 3), three methods were evaluated: pooling based on the point estimate of the treatment effect, pooling based on the confidence interval lower bound estimate, and OTR pooling. After pooling, the treatment effect of each pooled group is estimated as the weighted average of the treatment effect estimated in Step 2: the treatment effect estimates of and are represented by and , where

The following metrics were summarized to compare the performance of different methods under three different categories: variable selection, treatment assignment, and treatment effect estimation.

|  |  |
| --- | --- |
| **Category** | **Metric** |
| (Predictive) Variable Selection | Sensitivity; PPV; F1 Score |
| Treatment Assignment | Accuracy |
| Treatment Effect Estimation | Bias; MSE; coverage |

These are defined as:

* Sensitivity: Among the truly predictive variables, what % of variables were correctly identified.
* PPV (Positive Predictive Value): Among the variables identified to be predictive, what % are truly predictive.
* F1 score: 2\*(PPV\*Sensitivity) / (PPV+Sensitvity)
* Accuracy: % of patients that are assigned to the “correct” treatment assignment
* Bias/MSE: Given an identified “Benefitting” or “Non-benefitting” subgroup, what is the bias/MSE of the estimated treatment effect?

The simulation results are provided in ***Figure 2***, ***Figure 3***, and ***Figure 4***. These results suggest that virtual twin and MOB-OLS generally perform the best. For the pooling step, treatment effect estimation combined with bootstrapping is the recommended approach, although this could be computationally burdensome. Bootstrap resampling can reduce the bias and MSE in treatment effect estimate compared to the naïve approach.

**Figure 2: Variable Selection Performance**

**Binary Endpoint**Chart, scatter chart

Description automatically generated

**Survival Endpoint**

**Chart, scatter chart

Description automatically generated**

**Figure 3: Treatment Assignment Performance**

**Binary Endpoint**

Calendar

Description automatically generated

**Survival Endpoint**

Chart, scatter chart

Description automatically generated

**Figure 4: Treatment Effect Estimation Performance by Pooling Method**

**Binary Endpoint**

A picture containing text, cargo container

Description automatically generated**Graphical user interface, text, application, chat or text message

Description automatically generated**

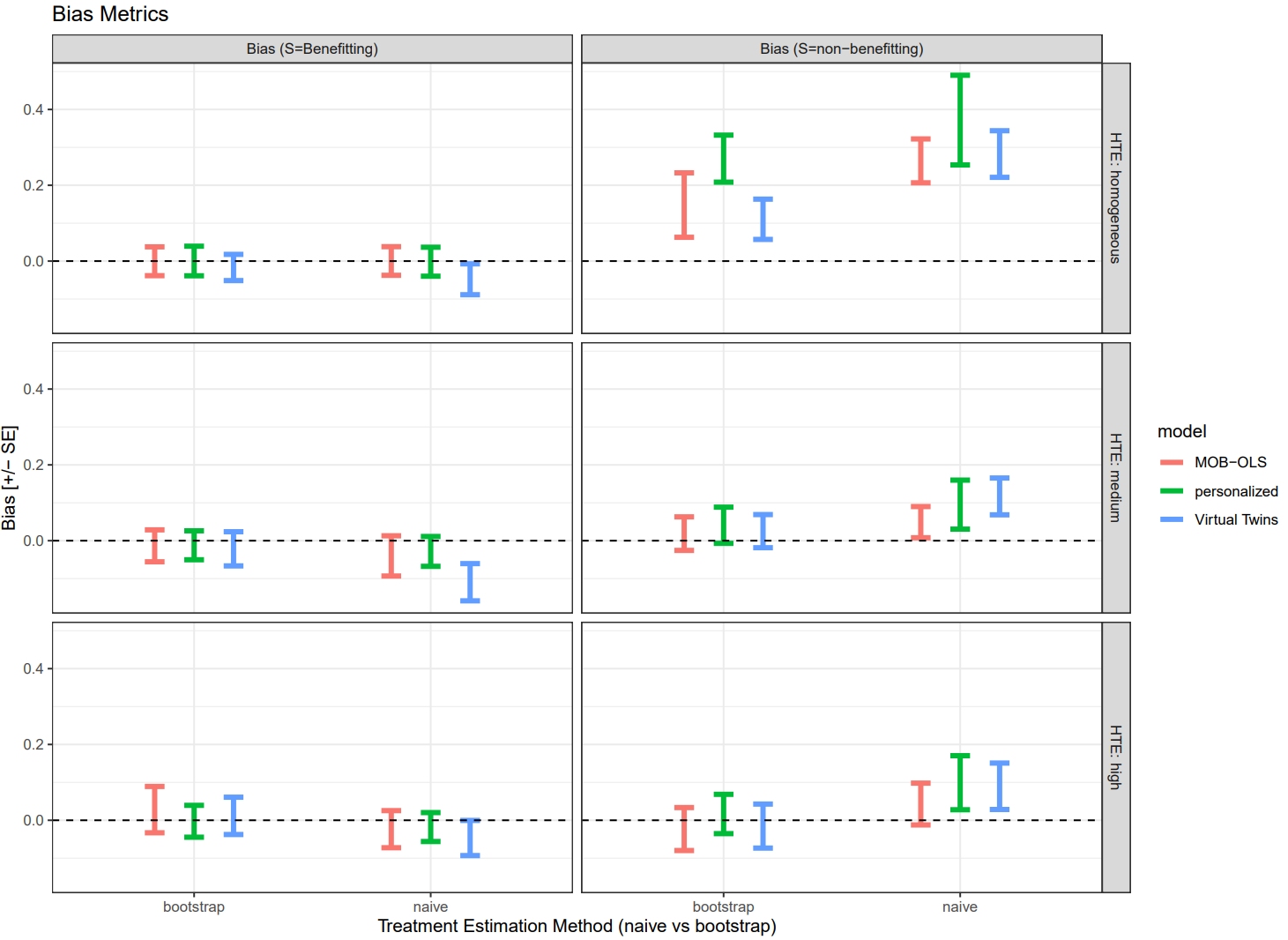
**Survvial Endpoint**

Calendar

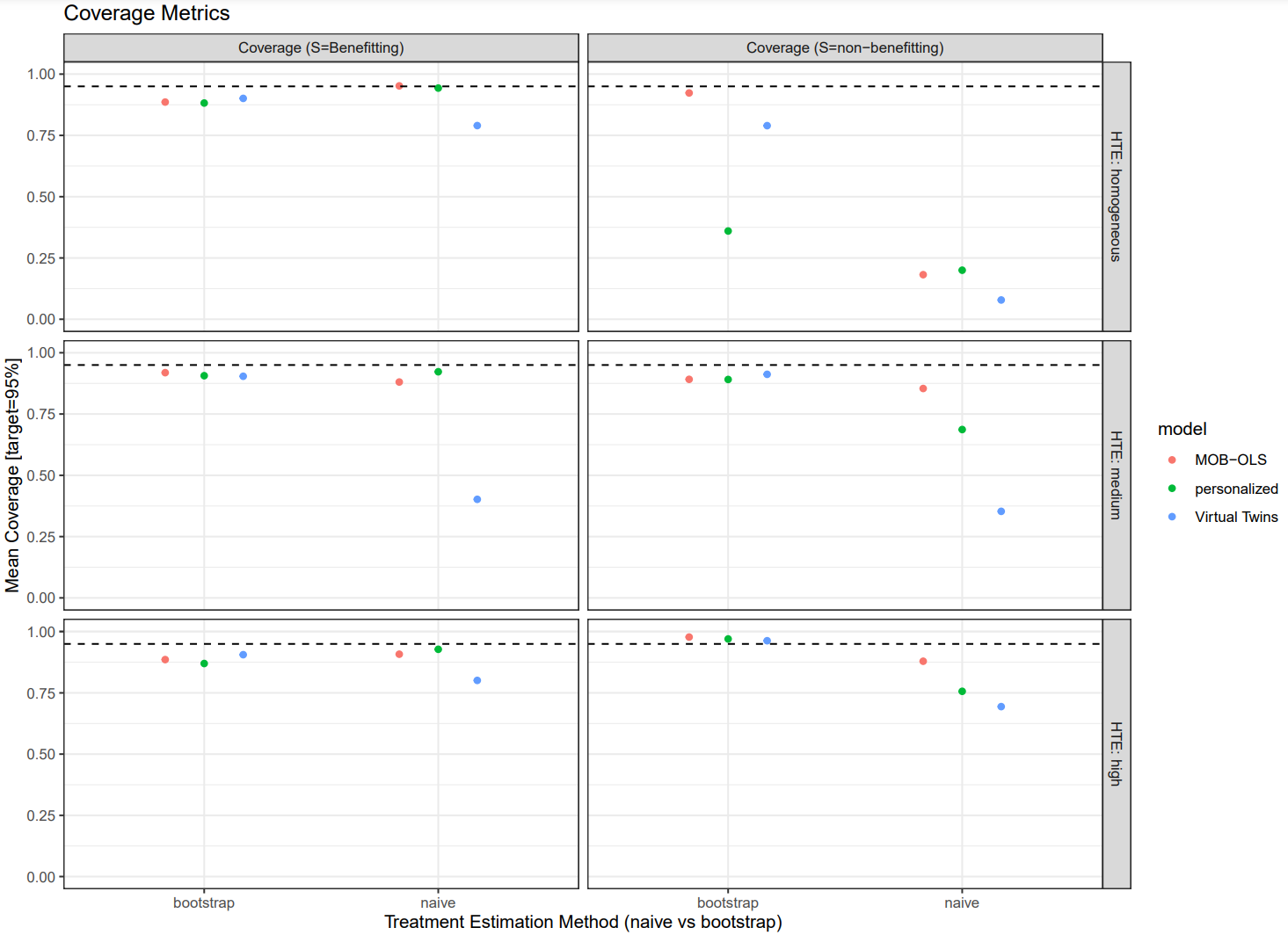
Description automatically generated**Graphical user interface, text, application, chat or text message

Description automatically generated**

While pooling with bootstrap treatment estimates yielded good results, this can be computationally burdensome. Here, we assess the bias/coverage of “Benefitting” / “Non-Benefitting” treatment effect estimates for a binary outcome when “naïve” treatment effect pooling is used, with and without bootstrap resampling for treatment effect estimates. See ***Figure 5*** and ***Figure 6*** for bias and coverage estimates respectively. In general, this further illustrates the importance of resampling methods to estimate treatment effects for subgroups that are adaptively identified by some model/algorithm.

**Figure 5: Treatment Effect Estimation (Bias) by Subgroup, Binary Outcome   
(With Naïve Treatment Effect Pooling)**

**Figure 6: Treatment Effect Estimation (Coverage) by Subgroup, Binary Outcome   
(With Naïve Treatment Effect Pooling)**



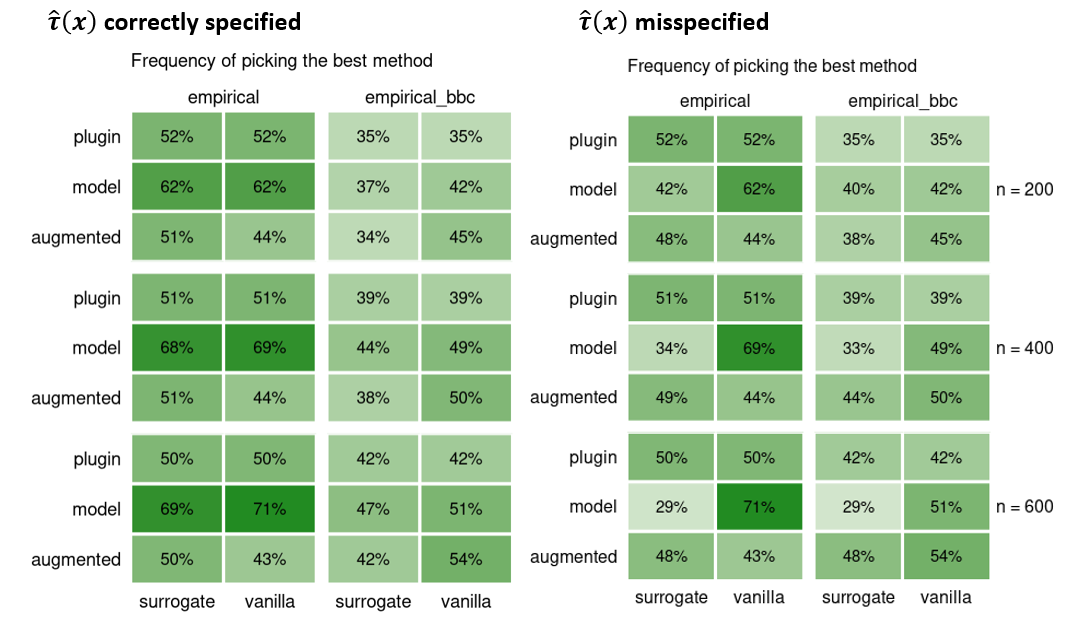
## **6.3.3: Model Selection**

For all results below, candidate models included MOB, Personalized, and Virtual Twins. Here, results are based on moderate model misspecification for model-based or augmented value functions. Some general comments:

* In the presence of no model-misspecification, model-based value functions tended to perform the best in terms of treatment classification metrics and selection of the truly predictive variables. Otherwise, the plug-in approach performed best.
* The augmented value functions, although theoretically should be more efficient, generally performed similarly or slightly worse than the plug-in (especially at the lower sample sizes).
* A bootstrap bias correction approach was tested for each value function. While this bias correction performed worse than the observed value function, whether there is benefit is using resampling for value function estimates needs more research.
* The surrogate value function performed similarly to the “vanilla” value functions. This suggests using the surrogate value as the surrogate approach theoretically uses more information.
* In general, compared to the individual subgroup models, the model selection yielded reasonably performance in terms of treatment classification metrics and selecting the truly predictive variables.

See ***Figure 7*** for the frequency of selecting the best model using various value function approaches, with and without model misspecification of the individual treatment effect estimates [Binary Outcome].

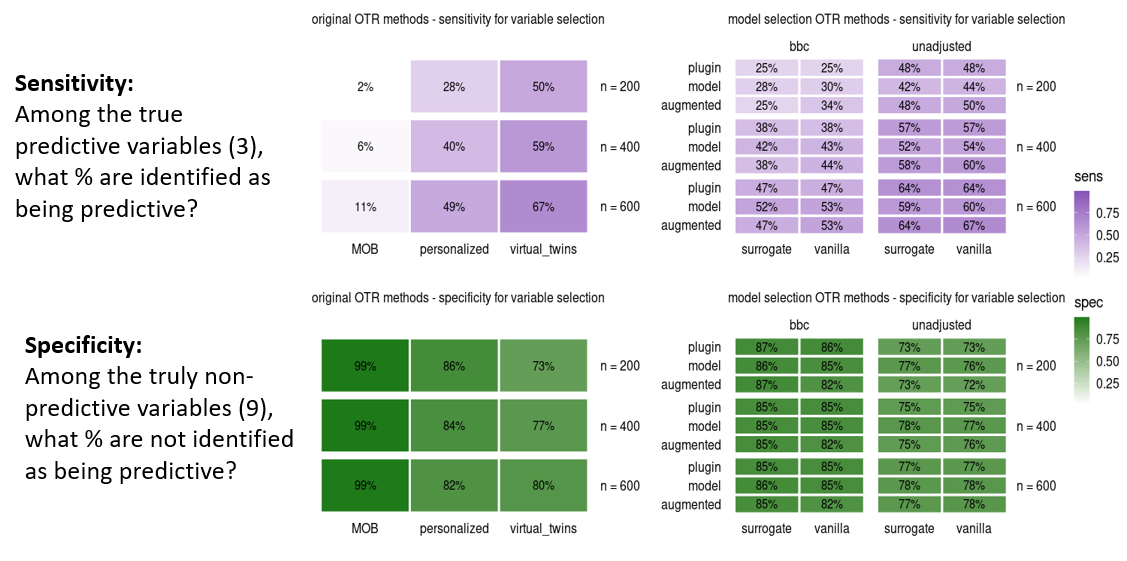
**Figure 7: Frequency of Picking the Best Method, with and without Model Misspecification  
(Binary Outcomes)**

  
**Note:** “Empirical” refers to using the observed data for value function estimates, while “empirical\_bbc” uses a bootstrap bias correction.

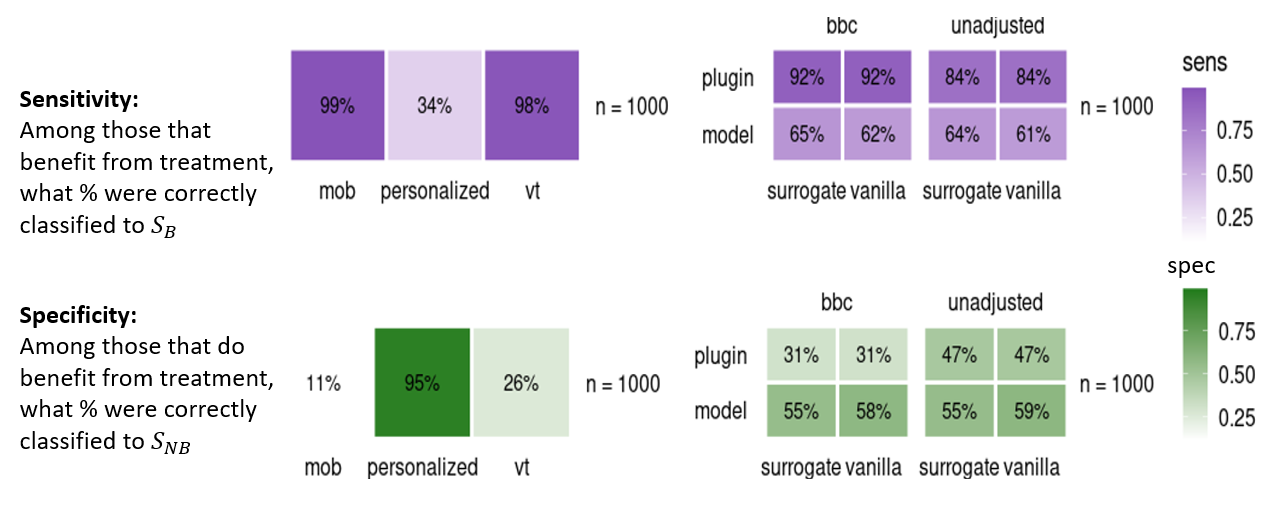
See ***Figure 8*** for subgroup selection simulation results for individual models and using value-based model selection [Binary Outcome]. Among those that benefit from treatment, sensitivity corresponds to the % who are also correctly classified “Benefitting” . Among those that do not benefit from treatment, specificity corresponds to the % who are also correctly classified “Non-Benefitting” .

**Figure 8: Subgroup Selection, Sensitivity, and Specificity of Treatment Assignments   
(Binary Outcome)**  
**Note:** “unadjusted” refers to using the observed data for value function estimates, while “bbc” uses a bootstrap bias correction.

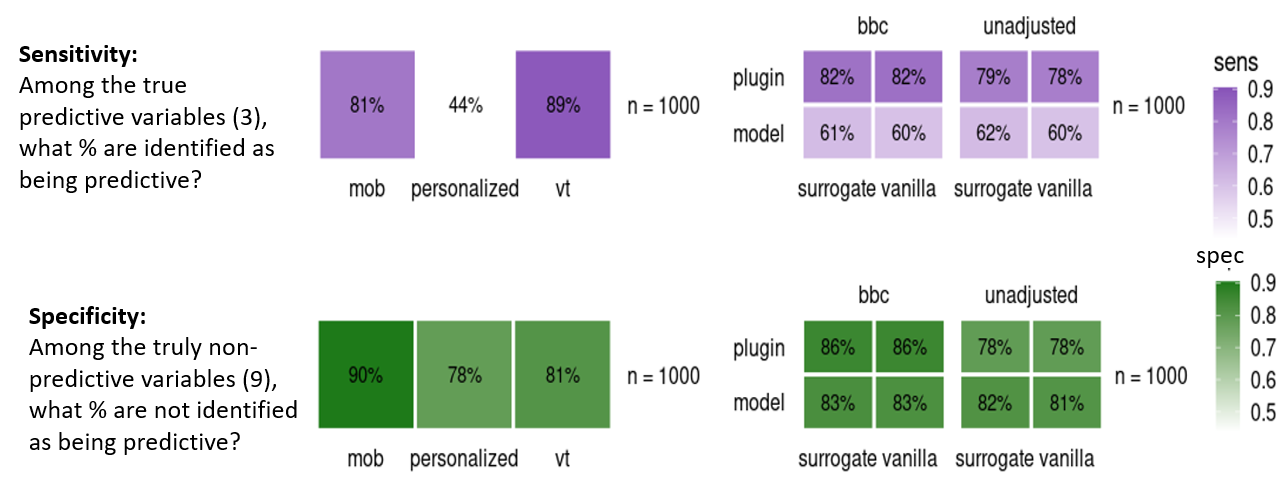
See ***Figure 9*** for variable selection simulation results for individual models and using value-based model selection [Binary Outcome]. This checks whether the truly predictive (or non-predictive) variables were included in the subgroup definitions. Among the truly predictive variables, sensitivity is the % that were correctly identified. Among the truly non-predictive variables, specificity is the % that were correctly not identified.

**Figure 9: Predictive Variable Selection, Sensitivity, and Specificity of Selected Variables (Binary Outcome)**  
**Note:** “unadjusted” refers to using the observed data for value function estimates, while “bbc” uses a bootstrap bias correction.

See ***Figure 10*** for subgroup selection simulation results for individual models and using value-based model selection [Survival Outcome]. Among those that benefit from treatment, sensitivity corresponds to the % who are also correctly classified “Benefitting” . Among those that do not benefit from treatment, specificity corresponds to the % who are also correctly classified “Non-Benefitting” .

**Figure 10: Subgroup Selection, Sensitivity, and Specificity of Selected Variables   
(Survival Outcomes)**  


See ***Figure 11*** for variable selection simulation results for individual models and using value-based model selection [Survival Outcome]. Among the truly predictive variables, sensitivity is the % that were correctly identified. Among the truly non-predictive variables, specificity is the % that were correctly not identified.

**Figure 11: Predictive Variable Selection, Sensitivity and Specificity of Selected Variables (Survival Outcomes)**  


## **6.4: Real Data Examples**

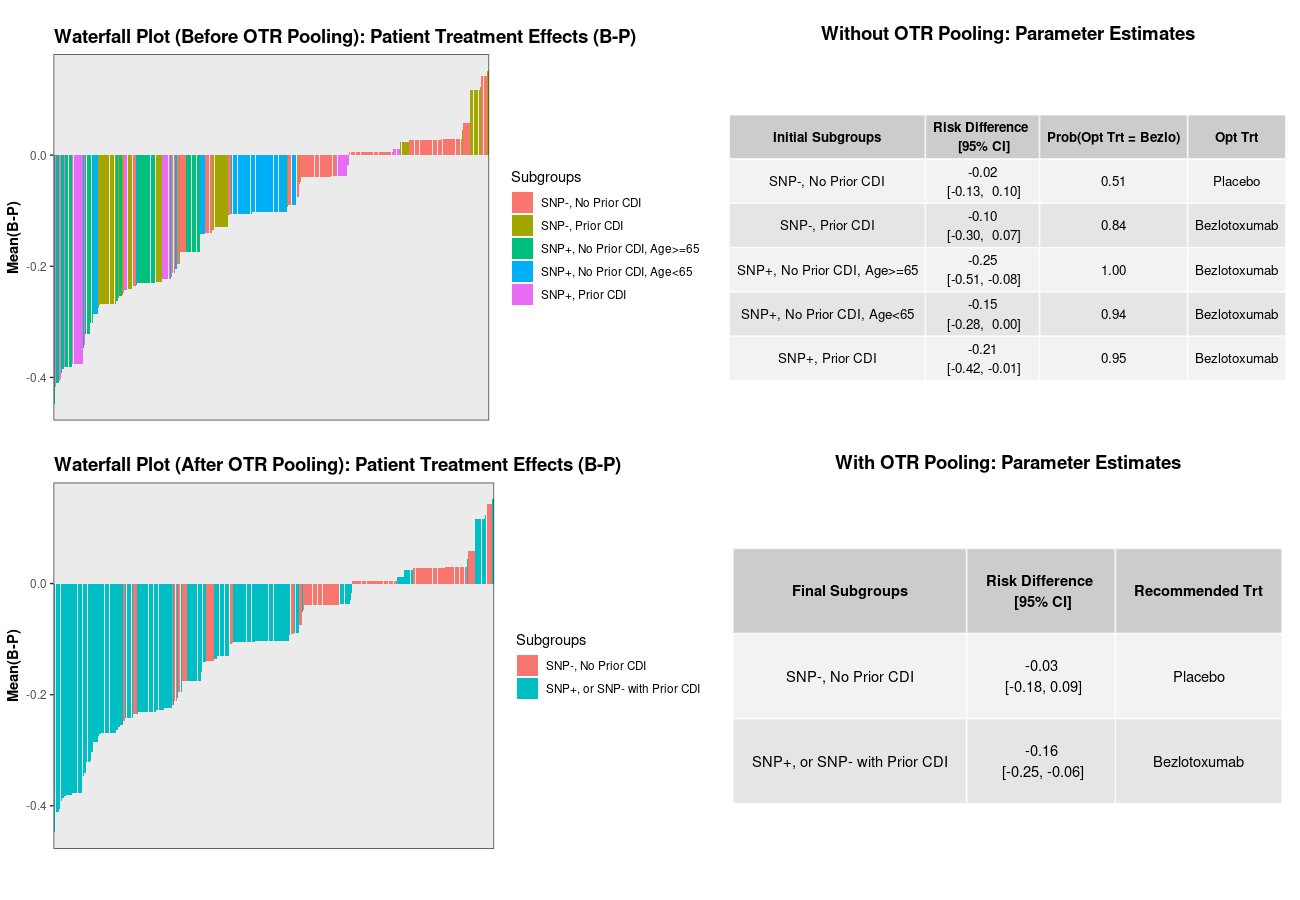
## **6.4.1: The Bezlotoxumab Example**

The data for our motivating example come from a phase III, double-blind, randomized, placebo-controlled clinical trial of an experimental drug (bezlotoxumab) in patients being treated for Clostridium Difficile (CDIFF) infection (CDI) (Wilcox MH et al 2017). There were two bezlotoxumab containing arms; for illustration, both are combined into a single arm and referred to as the bezlotoxumab arm. Of interest here is the CDI recurrence risk difference between the bezlotoxumab and placebo arms with a clinically meaningful difference of -0.10, i.e., a 10 percentage point lower CDI recurrence risk for bezlotoxumab versus placebo. Five baseline variables were pre-specified in the protocol as risk factors for CDI recurrence: prior history of CDI within past 6 months, immune compromised (Yes or No), CDI severity (Yes or No), prior history of CDI (ever, Yes or No), hypervirulent strain (Yes vs No) and age (<65 vs >=65). An additional variable was subsequently identified in a genome-wide association study (GWAS): SNP+ (yes if patient had one or more copies of the minor allele for SNP rs251613, and no otherwise).

For this illustrative example, MOB-OLS was used for the initial subgroup identification with and without OTR pooling (results were the same with treatment effect pooling, but this illustrates how OTR probabilities can generally apply). The patient-level treatment estimates (HTE) were estimated using a Virtual Twins type approach, with bootstrap resampling (Section 3.4) used for subgroup-specific treatment estimates. See ***Figure 12***. Overall, these results indicate that patients with no prior CDI history and SNP- status had limited reduction in CDI recurrence from using bezlotoxumab, while patients with prior CDI history or a SNP+ status had a clinically significant reduction in CDI recurrence. More generally, this example highlights how stratified medicine analyses can highlight treatment effects at the individual and subgroup level.

## **6.4.2: The VICTORIA Clinical Trial Example**

This example comes from a phase III, randomized, double-blind, placebo-controlled clinical trial of an experimental drug (vericiguat) in patients with chronic heart failure and an ejection fracture of less than 45% (Armstrong P et al 2020). The primary outcome was a composite of death from cardiovascular causes or first hospitalization for heart failure. Using the pre-specified list of subgroups shown in Armstrong P et al 2020, PRISM with MOB-OLS (with log-rank transformation of the survival outcome) was fit. See ***Figure 12***. In general, treatment heterogeneity seems largely driven by NT-proBNP, such that patients with patients with NT-proBNP > 5314 showing limited benefit from using vericiguat, while patients with NT-proBNP 5314 had a clinically significant improvement in the primary outcome (composite of death).

**Figure 12: Illustrative Example, CDIFF Clinical Trial**  
****  
**Note:** Top Left: Waterfall plot of patient-level treatment effects by initial subgroups Top Right: Estimated optimal treatment probabilities along with optimal treatment classification (based on Youden Index of 0.67). Bottom Left: Patient-level treatment effects by final subgroups. Bottom Right: Bootstrap based point-estimates / CIs for the final subgroups.

**Figure 13: Illustrative Example, VICTORIA Clinical Trial**

