University of Cologne

Faculty of Business, Economics and Social Science

Master Thesis

Comparison of Machine Learning Methods for Optimal Treatment Assignment and the Winner's Curse

Author
Fabian Meessen
Matr.-No.: 7306010

Supervisor
Prof. Dr. Tom Zimmermann

June 17, 2022

Contents

List of Figures

List of Tables

1 Introduction

Many policy and business decisions require the estimation and understanding of causal effects and the effectiveness of treatments. Historically, the examination of average effects has been the most prominent. Recent advancements in machine learning methods, the availability of large data sets, and causal inference literature lead to the estimation of heterogeneous treatment effects (HTEs) becoming of wide interest. In the center of the discussion lies the question of if and which treatment optimally to use or assign. Therefore, one of the primary uses for the identification of heterogeneous treatment effects is the assignment of an optimal treatment, with individual treatment effectiveness being judged by their predicted treatment effect.
Economic literature on causal inference gained in popularity in recent years. Especially the medical sector with growing interest in patient-centered outcomes (Willke et al., 2012) has been the focus of heterogeneous treatment effect estimation and optimal treatment assignment. But with application fields such as the incentivization of effort (e.g. for employee management) (DellaVigna and Pope, 2018) or the effectiveness of public policies (e.g. education, tax), the potential for economical use cases should not be neglected. From a business perspective, areas such as the differing impact of advertising or marketing offers on consumer purchases, or individualized website presentations are of interest.

Kernel methods, nearest-neighbor matching, or series estimation are classical nonparametric approaches for the estimation of heterogeneous treatment effects (Wager and Athey, 2018). While these methods provide good prediction performance with few covariates, performance rapidly declines with an increasing number of covariates. This makes the argument for machine learning methods, which tend to perform much better with many covariates than the classical approaches, but oftentimes need a larger number of observations for good prediction performance.
With a growing literature on machine learning methods for treatment effect estimation, the question arises about which methods perform well for optimal treatment assignment and how to compare the methods on an empirical dataset. In this thesis, I will present the counterfactual tree-based method called "Virtual Twin Random Forest" (Foster et al., 2011), the directly es-

timating tree-based method "Causal Forest" (Wager and Athey, 2018), and a method based on feed-forward neural networks, "Causal Net" (Farrell et al., 2021). Furthermore, I will compare the methods predicting heterogeneous treatment effects for optimal treatment assignment on an empirical dataset about the incentivization of manual labor in an online experiment using a method based on Hitsch and Misra (2018)[1].

A major challenge to optimal treatment assignment using heterogeneous treatment is the winner's curse. Overestimated treatments are more likely to be identified as the optimal treatment, as the optimal treatment is chosen by the treatment with the highest predicted treatment effect. I will outline the concept of the winner's curse, present shrinkage methods as a possible solution, and evaluate the effectiveness of the shrinkage methods applied to the predictions of the machine learning methods on the empirical data set.

My thesis proceeds as follows: First, the related literature will be portrayed in Section 2. I will present the machine learning methods in Section 3. The data set and the empirical analysis design will be outlined in Section 4, and the respective results in Section 5. Section 6 explains the winner's curse, the shrinkage estimators, and empirical results with applied shrinkage. Section 7 concludes.

2 Related Literature

This study adds to the existing literature on causal inference and optimal treatment assignment. Kleinberg et al. (2015) clarify the distinction between the need for causal inference and prediction in policy applications. Kitagawa and Tetenov (2018) develop a frequentist *Empirical Welfare Maximization* method for optimal treatment assignment. Manski (2004) suggest optimal treatment assignment to maximize social welfare is distinct from the usually used point estimation with hypothesis testing approach. Based on this, Hirano and Porter (2009) develop asymptotic normality theory for statistical treatment rules mapping empirical data into treatment choices.

Closely connected to my thesis is the growing field of methods used for the estimation of heterogeneous treatment effects. There are three methods I will use in this thesis. The virtual twin random forest (VTRF) (Foster et al., 2011) is a method using random forests to create counterfactuals and predict the dif-

[1]For the code for this thesis, see `https://github.com/fmssn/master-thesis`.

ference between observed outcome and counterfactual outcome. The causal forest[2] (Wager and Athey, 2018) alters the classic random forest/causal tree approach to directly estimate the treatment effect. The causal net (Farrell et al., 2021) is a recent contribution to the HTE estimation literature and uses custom layers appended to a feed-forward neural network to jointly estimate the effect of covariates and the treatment status on the outcome.

There are also various methods for estimating heterogeneous treatment effects not further outlined in this thesis. Imai and Ratkovic (2013) present a Support Vector Machine method with LASSO constraint for estimating heterogeneous treatment effects with the goal of optimal treatment assignment. Taddy et al. (2016) examine an online experiment and predict heterogeneous treatment effects using Bayesian nonparametric methods, including Bayesian Trees/ Forests. The authors further make the point for machine learning methods over conventional methods in this application, as their data set contains very poor behaving properties for classic approaches. The effect sizes are tiny with the majority of probability mass at zero, long tails of the distribution and density spikes, which they state are fairly representative for A/B tests on online websites/services.

As used in this thesis, one primary use for estimating heterogeneous treatment effects is optimal treatment assignment. Hitsch and Misra (2018) (whose methods I will use for model comparison) cite Simester et al. (2020) as one of their primary influences for the recent interest in heterogeneous treatment effects in evaluating marketing policies. Conventionally, one would compare two or more proposed targeting policies with randomization by policy at the cost of not being able to use the data for comparison with other policies afterward. Now, interest in randomization by action makes alternative policy evaluation possible and paves the way for treatment effect estimation.

The data I am using for my empirical analysis is from an experiment that was conducted on Amazon MTurk, a platform used primarily for small-scale contract labor with increasing popularity for behavioral experiments. Next to making collecting such a large scale sample possible and decreasing costs for payoffs, the more diverse participant pool compared to traditional college experiments may provide better external generalization (Follmer et al., 2017). Very closely related to this study is the work by Opitz et al. (2022), who examine the performance of targeted assignment of incentive schemes by a machine

[2]More specifically, the used transformed objective causal forest.

learning algorithm. They conducted two large-scale experiments, each with an extensive personality trait survey and a following manual labor task. The first experiment was used for pre-analysis, model selection, and training of the model, and the second one compared the performance of treatment assignment by the virtual twin random forest. The data I will be using in the empirical section is from the first experiment. The authors found that, the performance of the participant could be predicted by previously captured personality traits. Assignment by predictions using the virtual twin random forest yielded significantly higher outcomes than assigning the best overall performing treatment from the first experiment.

3 Causal Machine Learning of Effect Heterogeneity

In this section, I will outline the methodology behind conditional average/ heterogeneous treatment effect estimation and the used machine learning methods. First, the conditional average treatment effect will be defined. I will then explain the fundamentals of tree and forest based estimators. Subsequently, the forest based estimators virtual twin random forest and causal forest will be described. Lastly, I will portray the causal net approach.

3.1 Conditional Average Treatment Effect

Say the data consists of observations $(Y_i, T_i, X_{1i}, ..., X_{Ji})$, $i = 1, ..., n$ with n being the number of observations and J being the number of covariates observed. T_i is the individual treatment status and is equal to 1 if the individual was treated and equal to 0 if not. $X_{1i}, \ldots, X_{Ji}$ denotes the J covariates for individual i.
The conditional average treatment effect (CATE), which represents the heterogeneous treatment effect, is defined as

$$CATE = E\left[Y_i(1) - Y_i(0) \mid X = x\right] = \tau(x). \tag{1}$$

The conditional average treatment effect would be the same for each individual and equal to the average treatment effect (ATE) in case of no heterogeneity.

3.2 Basics of Forest Predictors

Before moving on to outlining the tree-based methods of the virtual twin random forest and the causal forest for predicting HTEs, I will explain the basic mechanics of regression trees and random forests, as both causal methods are heavily based on the baseline methods and mainly differ by slight alterations.

3.2.1 Regression Trees

Regression trees are a type of decision tree that predict a continuous outcome variable Y_i. They can be seen as a piecewise constant approximation. Regression trees recursively perform binary splits along one variable X_j at a time[3], which yields rectangular regions $R_t,\ t = 1, \ldots, T$ with T being the number of total regions. An exemplary regression tree with resulting regions can be seen in Figure 2.

In each recursion step, for each variable $X_j, j = 1, ..., J$, the set S_j splitting into two nodes (regions) is found which minimizes (or maximizes) the objective function (criterion along which to split, e.g. mean squared error). Then the split (X_j^*, S_j*) with the overall minimum (or maximum) is chosen. The recursion is performed until a stopping criterion is met. Common stopping criteria are maximum tree depth (number of consecutive splits) or a minimum number of observations required to perform a split/to be in a leaf (terminal node).[4]

The estimate in region R_t, $\hat{Y}_{R_t}$ obtains as the response of training observations falling into region R_t

$$\hat{Y}_{R_t} = \overline{Y}_{R_t} = \frac{1}{N(R_t)} \sum_{i:X_i \in R_t} Y_i, \tag{2}$$

with $N(R_t)$ being the number of training observations in region R_t. The estimate for x of the overall regression tree, $\hat{Y}_{RegressionTree}(x)$ then also just corre-

[3]The same variable may be chosen multiple times in consecutive splits if it minimizes the objective function.

[4]Another approach other than determining max depth is to grow a very large tree and then prune the tree afterward. The idea behind this is that a "poor" split early on may be followed by a very good split (in terms of reduction of the objective function). Breiman et al. (1984) proposed a form of pruning known as "Minimal Cost-Complexity Pruning", also implemented in the common packages in `Python` and R. This is however not used for this thesis.

sponds to the estimate in the region R_t where x falls[5], $\hat{Y}_{R_t}$,

$$\hat{Y}_{RegressionTree}(x) = \sum_{t=1}^{T} \hat{Y}_{R_t} \mathbb{1}_{R_t}(X = x). \tag{3}$$

How well a regression tree predicts is very dependent on the present data. If there are highly non-linear and complex relationships between covariates, regression trees are expected to outperform classical regression approaches (James et al., 2013). Regression trees provide good visualization and can be interpreted easily. Furthermore, they tend to have a low bias, however, this comes at the cost of overfitting and very high variance, resulting in a comparably poor predictive performance.

3.2.2 Random Forest

The random forest, as introduced by Breiman (2001) with previous work of Ho (1995), is an ensemble learning method based on decision trees and aims to reduce the high variance of decision trees, therefore improving predictive performance. The idea behind the reduction of the variance of random forests is that averaging a set of observations reduces the variance. To illustrate that, the variance of each observation given a set of independent observations $X_i, \ldots, X_n$ is σ^2, while the variance of the mean of the observations $\overline{X}$ is σ^2/n. Therefore optimally, we would like to draw B times from the original distribution and train B times decision trees $\hat{f}^1(X), \hat{f}^2(X), \ldots, \hat{f}^B(X)$ and average over the individual predictions as the final prediction. This is unfeasible to do due to finite data. One can use a resampling method to generate additional data sets, such as the Bootstrap method suggested by Breiman (2001). With bagging (bootstrap aggregation), one draws n times from the training data set of length n with replacement to create B bootstrap training sets of the same length as the original data set. The B bootstrap training sets are replicate data sets drawn from the bootstrap distribution, approximating the underlying distribution. Then B individual trees will be trained on the respective bootstrap training set, yielding the random forest prediction as the average of

[5] $\mathbb{1}_{R_t}(X)$ is an indicator function, with

$$\mathbb{1}_{R_t}(X) := \begin{cases} 1 & \text{if } X \in R_t, \\ 0 & \text{if } X \notin R_t. \end{cases}$$

the predictions of the individual trees by

$$\hat{f}_{\text{bag}}(X) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^b(X), \tag{4}$$

with $\hat{f}^b(X)$ being the prediction of the tree trained on the bagging sample b. One useful side effect of bagging is that it produces so-called *out-of-bag* (OOB) samples. It can be shown that with bagging, individual trees only use about 63% of total training observations (see Appendix A). The left out OOB samples can then be used to calculate accurate estimates for generalization errors (Breiman, 1996). Calculated error estimates are almost identical to those obtained by cross-validation (Friedman, 2017). Therefore, while training the random forest, performance metrics similar in quality to the test error are already available.

One further characteristic of random forests is that to decrease the correlation of the individual trees, at each split creation of the trees, only a randomized subset of covariates `max_features` with size $m \leq J$ is considered[6]. This feature randomization works especially well with a classification random forest and may not yield similarly promising results with a regression forest, as already noted by Breiman (2001). Furthermore, (Geurts et al., 2006, p. 13) find that for regression trees, more often than not the maximum number of features m is optimal at $m = J$[7]. As a random forest without feature randomization is essentially a bagged ensemble learner of ordinary decision trees, it may be referenced by some authors not as a random forest, but as a *Bagged Forest* (James et al., 2013).

3.3 Virtual Twin/Counterfactual Random Forest

The Virtual Twin approach by Foster et al. (2011) is one of the early contributions to the literature on predicting HTEs/CATEs. For the approach, one estimates individual counterfactual outcomes (virtual twins), then constructs the individual treatment effect by the difference between the observed and counterfactual outcome, which can in turn be used to construct the final

[6]The problem may be that if $m = J$ and there is a very strong covariate, in almost all trees the first split will be among this covariate, leading to similar tree structures.

[7]While the package `randomForest` in `R` uses the by Breiman (2001) suggested default value of $m = J/3$, `scikit-learn` in Python uses $m = J$ as default.

model predicting the individual treatment effect.
In the author's approach, the model for estimating the counterfactual is constructed by training a random forest to predict Y_i with input covariates (X_i, T_i). Then the counterfactual $\hat{Y}_i^0$ or $\hat{Y}_i^1$ is obtained by predicting the outcome with the trained model and the same data of input covariates (X_i, T_i), but with treatment status switched. However, when predicting the counterfactual for individual i, the random forest must not have been trained including the individual i. The authors use out-of-bag samples to address this issue. Alternatively, one could use cross-validation and get the counterfactuals from predictions on the test set.
With the counterfactual predicted, the constructed treatment effect obtains as

$$\tilde{\tau}_i = \begin{cases} \hat{Y}_i^1 - Y_i^0 & \text{if } T_i = 0, \\ Y_i^1 - \hat{Y}_i^0 & \text{if } T_i = 1. \end{cases} \tag{5}$$

The authors then train a regression tree on the whole data set and inputs to predict $\tilde{\tau}_i$, which is the final model that can be used for actual prediction of the individual treatment effects.
The basic idea of this method is fairly simple and easily expandable, and there are various alterations and extensions to the original Virtual Twin method. Lu et al. (2018) provide an overview of various random forest approaches for estimating HTEs, mainly based on the Virtual Twin method. For one, as already done by the authors, the input features can be expanded to include interaction terms between treatment indicator and covariates from (X_i, T_i) to $(X_i, T_i, X_i I(T_i = 0), X_i I(T_i = 1))$. The authors state that in their numerical work, this improved predictive performance.
A further modification already proposed, but not performed by Foster et al. (2011) is, instead of training one tree, to split the sample and train two separate random forests for treated and for untreated with which the counterfactuals are then created. That means, for constructing the counterfactual $\hat{Y}_i^0, i = 1, ..., n_0$, train a random forest with input covariates (X_i) only with the observations where $T_i = 1$. Then with that random forest, predict the missing outcome with the input covariates (X_i) where $T_i = 0$. The counterfactual $\hat{Y}_i^1$ is constructed symmetrically. Then the treatment effect is constructed as shown in equation 5 and predicted with a final predictive model. This method is called "Counterfactual Random Forest" by Lu et al. (2018). The counterfactual random forest variation of the virtual twin random forest is what I will be using for

the empirical analysis. Therefore, when referencing VTRF, the counterfactual random forest will be meant.
Notably, while random forests and regression trees are dominant in the presented papers, the method can easily be adjusted to use any predictive model in the step of constructing the counterfactual or in the final prediction step. I will be using a random forest for the prediction of the constructed treatment effect in the final step instead of a regression tree for improved prediction performance. Furthermore, an alteration would be not to train a final model for predicting the constructed treatment effect, one could also use the constructed treatment effect directly as the final prediction.

3.4 Causal Forest

Wager and Athey (2018) extend the random forest framework of Breiman (2001) to directly predict the treatment effect, without the need to construct a counterfactual beforehand. Furthermore, they emphasize discussing asymptotic theory, which before, the authors argue, hasn't been considered extensively in the literature on predicting heterogeneous treatment effects. I will outline the difference between causal trees compared to regular regression trees, focussing on the two methods of transforming the outcome and the transforming the node splitting objective function. Further, I will outline the additional sampling procedure called "honesty" (majorly used by Wager and Athey (2018) for asymptotic theory). Then I will explain the creation of causal forests from causal trees.

3.4.1 Causal Trees

Similar to a random forest, which consists of decision trees, the causal forest is an ensemble of causal trees. Causal trees as presented by Wager and Athey (2018) as well as previously by Athey and Imbens (2016) are an alteration of the regression trees discussed in Section 3.2.1.
Like regression trees, causal trees recursively split into regions R_t. Similar to regression trees, the main idea is to minimize the prediction error $\sum_{i=1}^{N} (\tau_i - \tau(x))^2$. As τ_i is not observed, constructing the tree structure and splitting into nodes is not as straightforward as with regression trees. Broadly speaking, there are two approaches to building causal trees and implicitly

minimizing the prediction error. One can either transform the outcome Y_i and then use regular regression tree methods directly(as outlined in Section 3.2.1, or transform the objective function along which to split.

Honesty

First, I will explain an additional difference to the baseline regression trees/random forests, which is the sample splitting approach called "honesty". While the resulting statistical properties are not directly relevant to my thesis, it is a major alteration from regular regression trees. One large motivation of Athey and Imbens (2016) was to be able to construct confidence intervals and conduct hypothesis tests. They state that this is however not possible with the many existing machine learning methods, as they are "adaptive". This means in the case of decision trees that they use the training data for both model selection (here tree construction) and effect estimation. Consequently, spurious correlations between features and outcomes affect model selection leading to only slowly disappearing biases, which can be difficult to quantify. The authors present an approach that they call "honesty", which leads to consistency and asymptotic normality of predictions without additional assumptions. Honesty will be used for both approaches to building the trees. The idea of honesty is that the training sample is split further into two samples, with one sample S^{tr} being used to split into nodes and build the tree structure and the other sample S^{est} being used to construct the estimate for the treatment effect. S^{te} denotes the test sample. This ensures that the asymptotic properties of the treatment effect estimates within the leaves are just as if the leaves were endogenously given.

Transformed Outcome Trees

Considering the transformed outcome Y_i^* proposed by Athey and Imbens (2016)

$$\begin{aligned} Y_i^* &= Y_i \left(T_i - p\right) / (p(1-p)) \\ &= T_i \frac{Y_i}{p} - (1 - T_i) \frac{Y_i}{1-p}, \end{aligned} \tag{6}$$

with $p = P(T_i = 1)$, one can use regular regression tree methods of splitting to implicitly minimize the error in predicting the CATE, using $E[Y_i^* | X_i = x] = \tau(x)$ (for the full proof, see Appendix B). The estimate of the treatment effect, analogously to regular regression trees (see Section 3.2.1), obtains as

$$\hat{\tau}(x) = \hat{Y}^*_{TOutTree} = \sum_{t=1}^{T} \hat{Y}^*_{R_t} \mathbb{1}_{R_t}(X = x),$$
$$\text{with} \tag{7}$$
$$\hat{Y}^*_{R_t} = \overline{Y^*}_{R_t} = \frac{1}{N(R_t)} \sum_{i:X_i \in R_t} Y^*_i,$$

which corresponds to the average transformed outcome in the region R_t where the observation x falls into. While the transformed outcome method is easy to apply as regular regression tree methods can be used, it is not efficient, as it only uses the treatment indicator T_i and its information for the construction of the outcome and not e.g. for construction of the tree structure. Consequently, the average of Y^*_i within a leaf R_t will only be an unbiased prediction of $\tau(x)$ if the share of treated observations in the leaf is exactly equal to the share p used to construct the transformed outcome. Therefore, the authors primarily use this method as a benchmark model, and it will not be used for my empirical analysis.

Transformed objective trees

The authors focus on transformed objective trees as their primary prediction method. For the later outlined empirical analysis, I will also be using this approach. While constructing the regression tree, alternative criterion/objective functions are used to choose the splits. Then the estimation of the conditional average treatment effect $\hat{\tau}(x)$ is given as the difference in means of Y between treated and untreated observations in the respective region R_t where x falls into, with

$$\begin{aligned} \hat{\tau}_{R_t} = &\frac{1}{N(R_t, T_i = 1)} \sum_{i:T_i=1, X_i \in R_t} Y_i \\ &- \frac{1}{N(R_t, T_i = 0)} \sum_{i:T_i=0, X_i \in R_t} Y_i, \end{aligned} \tag{8}$$

and

$$\hat{\tau}_{TObjTree}(x) = \sum_{t=1}^{T} \hat{\tau}_{R_t} \mathbb{1}_{R_t}(X = x). \tag{9}$$

where $N(R_t, T_i = 1)$ is the number of observations in region R_t which are treated and $N(R_t, T_i = 0)$ the number of observations in region R_t which are untreated. Therefore, with (transformed objective) causal trees, there always

needs to be a minimum number of observations ≥ 1 of both treated and untreated in each region[8].

As already stated, the true treatment effect τ_i can not be observed, therefore a corresponding error function like the mean squared error can not be constructed directly. Because of that, there are various objectives feasible for transformed objective trees. They largely focus on increasing the between leaf heterogeneity and may also reward splits which lower the in-leaf variance.

Athey and Imbens (2016) derive as their primary splitting criterion the negative expected value of a modified mean squared error given a partitioning Π[9] over the (tree constructing) training sample, which is maximized,

$$\begin{aligned} &- \widehat{\mathrm{EMSE}}_\tau \left(\mathcal{S}^{\mathrm{tr}}, N^{\mathrm{est}}, \Pi\right) \equiv \frac{1}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{S}^{\mathrm{tr}}} \hat{\tau}^2 \left(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi\right) \\ &- \left(\frac{1}{N^{\mathrm{tr}}} + \frac{1}{N^{\mathrm{est}}}\right) \cdot \sum_{R_t \in \Pi} \left(\frac{S^2_{\mathcal{S}^{\mathrm{tr}}_{\mathrm{treat}}}(R_t)}{p} + \frac{S^2_{\mathcal{S}^{\mathrm{tr}}_{\mathrm{control}}}(R_t)}{1-p}\right), \end{aligned} \tag{10}$$

where $\hat{\tau}^2 \left(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi\right)$ is the treatment effect prediction for X_i in the tree construction data sample $\mathcal{S}^{\mathrm{tr}}$ for the candidate tree structure Π. N^{tr} and N^{est} correspond to the total number of observations in the tree constructing and treatment effect estimating training samples. $S^2_{\mathcal{S}^{\mathrm{tr}}_{\mathrm{treat}}}(R_t)$ and $S^2_{\mathcal{S}^{\mathrm{tr}}_{\mathrm{control}}}(R_t)$ are the within-leaf variances of leaf R_t of observations in the tree constructing training sample only considering the treated or untreated observations, respectively. p is the share of treated observations in the whole training sample. The first term of the above splitting criterion rewards splits that lead to high treatment effect heterogeneity, while the second term penalizes splits with high within-leaf variance.

The recursive construction of the tree is otherwise identical to those of regular regression trees. It is constructed by iterative binary splits, choosing the split (X_j^*, S_j^*) with the highest increase in $-\widehat{\mathrm{EMSE}}_\tau$ compared to the previous partitioning Π, $\Delta\left(\mathcal{S}^{\mathrm{tr}}, N^{\mathrm{est}}, S_j^*\right)$. This yields the partitioning Π^* with the highest splitting criterion $-\widehat{\mathrm{EMSE}}_\tau \left(\mathcal{S}^{\mathrm{tr}}, N^{\mathrm{est}}, \Pi^*\right)$.

As outlined before, the goal of the splitting criterion should be that mainly between leaf heterogeneity is maximized, with optionally putting a restraint on the in-leaf variance. Therefore, the authors also outline alternative splitting

[8]With honesty, this means that not only $(X_i, T_i, Y_i) \in S^{est}$ are used for constructing the tree structure, also $(X_i, T_i) \in S^{te}$ are used for determining whether as split leads to regions containing enough or too few observations of both treated and untreated.

[9]Π stands for the full tree structure, including all splits.

criteria such as maximizing the t-statistic for testing the null hypothesis that the average treatment effect is the two child nodes of the potential split, as proposed by Su et al. (2009). More splitting criteria are proposed by Athey et al. (2019) in a subsequent paper on generalized random forests.

3.4.2 From Causal Trees to Causal Forests

The step from causal trees to causal forests is very similar to the step from regression trees to random forests outlined in Section 3.2.2: A predetermined number of bootstraps samples are drawn with replacement from the (training) sample with individual causal trees being trained on each. Additionally, at each split decision for each tree, only a randomized subset of covariates smaller or equal to the number of covariates is considered to reduce the correlation of individual trees.
The final prediction of the causal forest obtains as the mean predicted treatment effect of the individual causal trees,

$$\hat{\tau}_{\text{Forest}}(X) = \frac{1}{B} \sum_{b=1}^{B} \hat{\tau}^b(X), \tag{11}$$

with $\hat{\tau}^b(X)$ being the prediction of the causal tree trained on the bagging sample b.

3.5 Causal Net

The causal net is a method developed by Farrell et al. (2021) for estimating heterogeneous treatment effects, which jointly estimates the treatment effect and the effect from the covariates using a feed-forward neural network. Previously, neural networks were very infrequently used in economic academic research in comparison to other machine learning methods. I will explain the basics of feed-forwards neural networks and outline the changes to the network architecture used to construct the causal for heterogeneous treatment effect prediction.

3.5.1 Basics of the Feed-Forward Neural Network

A feed-forward[10] neural network serves as a non-linear approximation of a target function $f(x)$. The input layer consists of J nodes, where J is the number of covariates. The hidden layers each consist of a previously determined number of nodes, with the number of the layer being called *width*. The input of node u in the hidden layer l is given by $z_u^{(l)} \equiv \sum_{b=1}^{(l-1)} \beta_{bu}^{(l-1)} a_b$, where $\beta_{bu}^{(l-1)}$ is the parameter (weight) of node b in layer $(l-1)$ for node u in layer (l) and $a_b^{(l-1)}$ is the output of node b in layer $(l-1)$. For the output $a_u^{(l)}$ of the node, here the *ReLU* activation function[11] is applied to the input, so $a_u^{(l)} = max(z_u^{(l)}, 0)$. There is no activation function applied to the input and output layer, so $a_b^{(l=1)} = x_b$ and $a^{(l=L)} = \hat{y}$ [12].
The weights are then received by a backpropagation algorithm, which iteratively updates the weights. The main idea is that for each prediction, the contribution of each weight to the overall error (the value of the chosen loss function) is calculated, and the weight is then adjusted accordingly (in sign and magnitude).

3.5.2 Causal Net Architecture

The causal net is a feed-forward neural network with *ReLU* activation functions and, most notably, two custom layers as the parameter and outcome layer appended to the hidden layers. For an exemplary structure with two hidden layers, see Figure 1. In the causal net, the second to last layer (parameter layer, L_4 in Figure 1) consists of three nodes, two of which originate from the regular neural network and which receive their values from feed-forwarding from the last hidden layer with no activation function applied. The third node is the treatment indicator T_i and therefore does not depend on previously hidden layers. It can be seen as an additional input layer. The final output layer, aiming to predict the outcome Y_i, is given as the sum of both "regular" nodes in the previous custom layer, while one of those nodes is multiplied with the

[10] Feed forward refers to the connection between the nodes not forming a circle, with a clear structure of first, second, etc. layer.

[11] The authors only use the *ReLU* activation function $x \mapsto \max(x, 0)$, stating that the change from smooth sigmoid-type activation function to the *ReLU* activation functions is (next to improved computational power and stochastic optimization) the main reason for the recent rise in popularity and performance of neural networks, outperforming the previously used smooth sigmoid-type activation functions.

[12] Neural networks are also able to predict multiple outcomes simultaneously, so $a_b^{(l=L)} = \hat{Y}_b$, in the present case there is only one outcome variable

(binary) treatment indicator. The node that is interacted with the treatment indicator, therefore, estimates the heterogeneous treatment effect, $\hat{\tau}(X)$, while the node not interacted with the treatment indicator estimates the outcome of no treatment $\hat{Y}^0$. The network is then trained to minimize the mean squared error[13] in predicting Y, meaning $Y^0(X)$ and $\tau(X)$ (therefore also $Y^1(X)$) are jointly estimated by solving

$$\begin{pmatrix} \hat{Y}^0(X) \\ \hat{\tau}(X) = \hat{Y}^1(X) - \hat{Y}^0(X) \end{pmatrix} := \underset{\tilde{Y}^0, \tilde{\tau}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \tilde{Y}^0 (X_i) - \tilde{\tau} (X_i) T_i \right)^2 . \tag{12}$$

The predictions for the CATE/HTE are retrieved by feed-forwarding the covariates and then using the node value for $\hat{\tau}(X)$ (the node interacted with the treatment indicator) in the parameter layer. The authors state that this joint estimation outperformed separately estimating $Y^0(X)$ and $Y^1(X)$ with neural networks on the respective samples (similar to the Virtual Twin approach) in their findings.

4 Implementation of Estimators

This section describes the conducted empirical analysis. I will outline the data-set, the training and hyperparameter tuning for the models and the method to compare performance of optimal treatment assignment by predicted treatment effects on an empirical data-set.

4.1 Data

The data was taken from the first experiment round performed by Opitz et al. (2022) using Amazon MTurk with a US-only sample in September 2021 over roughly two and a half weeks.

Opitz et al. (2022) focused on whether a machine learning model, assigning individual optimal treatments, can outperform assigning only the best treatment overall in a subsequent MTurk study. I will focus on the first round of the study and the methods to compare the different machine learning models. In the study, participants, before the actual task, had to fill in an extensive survey on their demographics, personality traits, and social preferences. Next

[13] One could use other loss function such as the mean absolute error, however as I will be using the mean squared error as the loss function for my empirical analysis, I will focus on the MSE.

to age, gender, and the education level, questions were asked on the Big 5 personality traits (John et al., 1991), risk preferences, loss aversion, competitiveness, social comparison, altruism, and positive reciprocity (see Appendix of Opitz et al. (2022) for further information). In the subsequent working task, participants could achieve points by pressing the button 'a' then 'b', over ten minutes. The participants could see a timer, their current points, and their current bonus. Each participant was randomly assigned to one of six treatments or the control group, with the assignment determining the incentive (or lack thereof) to press the buttons and achieve points. These treatments were:

1. **Pay for Performance (PfP)** 5 cents for every 100 points
2. **Goal** \$1 if you score at least 2000 points
3. **Gift & Goal** \$1, would appreciate at least 2000 points appreciate if you try to score at least 2,000 points.
4. **Loss** \$1, lose it unless you score at least 2000 points
5. **Real-Time Feedback** \$0.02 times the percentile reached
6. **Social PfP** 3 cents for participant + 2 cents for Doctors without Borders for every 100 points
7. **Control** no extra payment

For detailed descriptions of the treatments and the text shown to the participants, see Appendix C.
The mean points for the treated observations was 1898, for the untreated 1533. For all treatments, the points were distributed significantly higher (see Table 1: Wilcoxon Rank Sum test for $\alpha < 1\%$) than the points in the control group. The mean and median outcomes are depicted in Figure 3. Notably, the distribution of average outcomes of treatments one, two, four, and five are distributed significantly above the points of treatment three and six (according to Wilcoxon Rank Sum tests at the 5% level, see Table 1). The estimated probability density functions of the points for the respective treatments can be seen in Figure 4. There is a density spike around 0, especially in the control treatment and treatment three, as there the payoff was the same no matter how many points the participant scored. This spike also explains the substantial gap between the mean and median.

As some participants did not take part in the experiment orderly, the dataset is cleaned from those participants based on previously defined criteria[14].

4.2 Training and Tuning

Optimal hyperparameters for the models of the previously outlined three methods will be determined by a grid search algorithm with three-fold cross-validation. For details on which parameters were tuned, see Appendix E.
Furthermore, for each method, a model for each of the six treatments will be trained. These models will be trained on a training set subsample which includes only the control observations and the observations which received the respective treatment. The prediction of the model will be the estimated treatment effect for the respective treatment. Consequently, the tuning of hyperparameters will be performed individually for each of the six models of each method. The treatment assignment of the overall model/method then corresponds to the treatment with the highest treatment effect.

4.2.1 Model Selection

The main issue with tuning hyperparameters for models predicting treatment effect is that there is no directly observable error term. Schuler et al. (2018) provide an overview of several metrics for evaluating model performance when predicting heterogeneous treatment effects. From their findings, the authors advocate for using the metric $\widehat{\tau\text{-risk}}_R$[15] as the model selection metric for individual treatment effect predicting models.
They base this on a simulation study, where using this particular loss function (in comparison to other loss functions) for model selection most consistently selects models with a low squared error of the predicted treatment effect $E\left[(\hat{\tau}(X) - \tau(X))^2\right]$[16].

[14]These criteria were: No button presses, spending time under a certain threshold on the pages, or scoring points that indicate cheating.

[15]This metric is derived from the $\mathcal{R}$-Learners by Nie and Wager (2021) using the Robinson (1988) decomposition to re-write the conditional average treatment effect function in terms of the conditional mean outcome. See Appendix D for the basic idea of the metric.

[16]Furthermore, selected models most consistently rightly predicted whether to treat the model at all or not. However, this is not a pressing issue with the present data set, as average treatment effects for all treatments are well above zero, and treatment cost is not regarded in this analysis.

In particular, the loss function is given by

$$\widehat{\tau\text{-risk}}_R = \frac{1}{N_{te}} \sum_{i \in N_{te}}^{N_{te}} \left(\left(Y_i - \hat{m}\left(X_i\right) \right) - \left(T_i - \hat{p}\left(X_i\right) \right) \hat{\tau}\left(X_i\right) \right)^2, \tag{13}$$

where $\hat{m}\left(X_i\right)$ is an estimation of $E[Y_i|X_i]$ and $\hat{p}\left(X_i\right)$ is the estimation of the treatment propensity $E[T_i = 1|X_i]$. I will be using a Lasso-Estimator for $\hat{m}\left(x\right)$ and Logistic Regression for $\hat{p}\left(x\right)$[17], both estimators being cross-validated[18] and trained on the training set to predict the respective $\hat{m}\left(x\right)$ and $\hat{p}\left(x\right)$ in the test set.

The metric $\widehat{\tau\text{-risk}}_R$ will be used for model selection for both the causal forest and the causal net methods.

The model selection for the random forests in the virtual twin random forest method is straightforward, as the regular mean squared error can be used. For the two forests constructing the counterfactual outcomes, the outcomes are estimated, therefore the regular mean squared error will be used for model selection. After the counterfactual-creating forests are tuned, as the difference in predicted outcomes is the variable of interest for the third, final random forest, here again, the regular mean squared error can be used. Theoretically one could also use $\widehat{\tau\text{-risk}}_R$ and then iterate over hyperparameter combinations of the three random forests, although this is computationally infeasible (with the here used grid search algorithm).

4.3 Comparison Method

For comparing the performance of the methods on an empirical dataset with unobservable true treatment effects, I will be using an approach based on the method outlined by (Hitsch and Misra, 2018, 9). The main idea of this approach is to use the observations, where coincidentally the predicted optimal treatment and the actual, randomly assigned, treatment are equal (matched). If the average outcomes of the matched observations are well above the mean treated outcome, this hints toward models properly assigning optimal treatments.

First, the models will be trained on a training sample and predict the treatment effects for all respective treatments of the observations in the test sample.

[17] Similar to the related work of Nie and Wager (2021) on $\mathcal{R}$-Learners.

[18] With regard to the choice of the penalty term.

For each observation and prediction method in the test sample, the optimal treatment will be assigned, according to the highest predicted treatment effect out of the used treatments. For some observations, the assigned optimal treatment will be equal to the randomly assigned treatment. I will refer to those observations as *matched*. One can then analyze the performance of the models by examining the average outcome of the matched observations. For models that correctly assign the optimal treatment, the average outcome of matched observations will be higher than for models which do not[19]. Furthermore, if the mean outcome of matched observations is higher than the mean outcome of all treated observations, this suggests that the assignment via the respective method is better than random assignment. If the mean outcome of matched observations is higher than the average outcome of individuals treated with a specific treatment, this suggests that assignment via the model is better than only assigning that specific treatment.

To ensure the empirical results are not driven by randomness out of the choice of the test and training samples, the described matched observation analysis (which I will refer to as Misra-Matching) will be conducted via one-hundred times repeated three-fold cross-validation. Then over all repetitions and folds, average outcomes of matched treatments will be calculated for each treatment for each method as well as overall.

Next to the average points of the matched observations over all repetitions and folds, I will compare the distributions of the average outcomes of matched observations over the repetitions and the number/share of repetitions in which the average outcome of matched observations is higher than the average outcome of the best performing treatment or higher than the average outcome over all treated individuals.

As outlined in Section 4.1, the average points of treatments three and six are significantly below those of the other treatments. Therefore, it is expected that models predict mostly much lower treatment effects for those treatments, such that they almost never get assigned as optimal. Consequently, there will be very few or no matched observations for treatments three and six. If a method is as good as random assignment for all treatments but treatment three and

[19] A high average outcome of matched observations does not necessarily have to coincide with actually *the best* treatment being assigned, it might be e.g. the second-best (with a high treatment effect). However, for most application examples using treatment assignments, correctly differentiating between effective and ineffective individual treatments is more important than predicting which one out of two very effective treatments is optimal. Therefore, this is only a minor challenge to the analysis.

six, and only does not assign treatments three and six as optimal, the average points of matched observations for the respective method will still be higher than the average points over all treatments. To exclude that models perform well mainly because they do not assign to treatments three and six, I will also compare the assignments only using treatments one, two, four, and five.

5 Main Results

In this section, the results of the previously outlined empirical analysis are reported. I will describe the results when using all six treatments and subsequently the results of the subset of treatments, only using treatments one, two, four and five.

5.1 Full Treatment Set

The average outcomes of matched observations for the full treatment set are depicted in Table 2, with Figure 5 showing the distribution of the overall average outcome of matched observations over the one hundred cross-validation repetitions for the three methods. For the three methods, the average points of matched observations are higher than the average points over all treatments, indicating that all models would perform better for treatment assignment than random assignment. With an average outcome of matched observations of 1982, the causal forest is the best performing model, the virtual twin random forest the second best with 1959, and the causal net the worst one with 1921. Only the causal forest Model has a matched average outcome higher than the average outcome in treatment four (1970). Notably, for all models, the average outcome of matched observations in treatment four is below the average points for treatment four, in the case of the better performing VTRF and causal forest well below with 1815 and 1788 compared to 1970. Furthermore, treatment four for both of these models is assigned the most, which is to be expected with it having the highest overall average outcome. This could be because treatment effects for treatment four are systematically overestimated, or because treatment four is used as a "baseline" treatment of the models. If predicted treatment effects for the other treatments are low because models do not predicted potential there, the prediction for treatment four will still be relatively high so that treatment four gets assigned. The difference in treatment five for those two models is exceptionally large (VTRF 2135, causal forest 2189

compared to 1931 average), which indicates that the models correctly assign more competitive-oriented participants to the Real-Time-Feedback treatment. Measured in the number of repetitions where the average points of matched observations was higher than the average outcome of treatment four, with 81/100 times the causal forest performs much better than the other two models (28/100 VTRF, 5/100 causal net). Only the causal net has repetitions where the overall average outcome of matched observations was worse than the average mean outcome of treated observations, which occurred in 18/100 repetitions.

For the causal forest alone, the results indicate that assigning treatment via the model is better than assigning only the best-performing treatment. Measured by the introduced metrics, the causal net performs much worse than the tree-based models. For understanding why the VTRF performs worse than the causal forest, I additionally matched the observations on the training set they were trained on[20], see Table 6. This can be interpreted as a counterpart to the training error. Here, the VTRF had by far the highest average points of matched estimations, 2339 compared to 2243 of the causal forest. This indicates that the VTRF overfits on the training set.

5.2 Sub-Treatment Set

One previously outlined challenge to the comparison method was, that as the distribution of the outcomes of treatments three and six are significantly below the outcomes of the other treatments, models may perform well mainly by not assigning treatments three and six as treatments, which may distort the comparison. Therefore, additionally, the analysis was performed on the subset of treatments only using treatments one, two, four, and five. The results for the subset of treatments are depicted in Table 3 and the distributions over the repetitions in Figure 6.

Again, with an average outcome of matched observations of 1980 (previously 1982) and 100/100 repetitions over the mean (now 1939, only considering the used four treatments) and a slightly reduced 75/100 repetitions over the average outcome of treatment four, the causal forest performs the best out of the three compared estimation methods. The virtual twin random forest performs relatively better than in the analysis using the full treatment set: The average outcome of matched observations over the hundred repetitions is 1970

[20]For each training split of the 100 times repeated three-fold cross-validation.

(previously 1959), and for 52/100 repetitions the average outcome of matched observations is over the average outcome of treatment four. However, for 3/100 repetitions, the average outcome of matched observations is below the average outcome of treatments one, two, four, and five. The metrics for the causal net are, like in the sample using all six treatments, well below the metrics of the other two estimation methods. The average outcome of matched observations over the hundred repetitions is with 1945 higher than in the full sample (1921), but still below the average outcome of treatment four (1970) and just slightly above the mean outcome of the subset of treatments (1939). Very similar to the previous results, only for 8/100 repetitions of the cross-validation the average outcome of matched observations is over the average outcome of treatment four, for 31/100 repetitions it is below the mean outcome of the considered four treatments.

As the results examining the subset of treatments are overall fairly similar to the ones of the full treatment set, there is no substantial threat to the validation of the findings when using treatments with fairly even average outcome levels compared to using treatments with varying average outcome levels. The most striking difference is the increased performance of the virtual twin random forest, with the results now indicating that using the VTRF for treatment assignment is at least as good as assigning only treatment four. An explanation for this may be that the VTRF wrongly assigns many observations to treatment six because of overestimation. The VTRF has matched 10,005 matched assignments with an average outcome of 1974 points compared to 2280 matched assignments and 2133 points by the much better performing causal forest. Further evidence for that is that for all four treatments, the individual average outcome of matched observations (columns one to four of Table 3) of the VTRF increases in the analysis with the subset of treatments compared to the whole set of treatments. For all four treatments, in assignments by the causal net and the causal forest, they decrease.

6 Winner's Curse

The concept of the winner's curse regarding optimal treatment assignment, as covered by Andrews et al. (2019), describes the problem that a treatment is more likely to be selected as optimal if it is overestimated. Consequently, in

the setting of optimal treatment assignment, the treatment effect of the optimal treatment is systematically overestimated. Therefore, the true underlying optimal treatment may not be chosen because another treatment was overestimated. I will outline the concept of the winner's curse with a stylized example of average treatment effects, present shrinkage methods as a possible solution, and show the empirical results with applied shrinkers.

6.1 Stylized Example

Andrews et al. (2019) illustrate the winner's curse with a stylized example of estimations of the average potential outcomes. Suppose there are n individuals randomly assigned to a binary treatment, i.e. $T_i = 1$ (treatment) or $T_i = 0$ (control) with $\frac{n}{2}$ individuals in each group. For each individual, their outcome Y_i (e.g. number of scored points) is observed. Computing the treatment $(Z_n^*(1))$ and control $(Z_n^*(0))$ averages yields

$$(Z_n^*(1), Z_n^*(0)) = \left(\frac{2}{n} \sum_{i=1}^{n} T_i Y_i, \frac{2}{n} \sum_{i=1}^{n} (1 - T_i) Y_i \right). \tag{14}$$

If the participants are a random sample from a population, so there is unconfoundedness, then $(Z_n^*(1), Z_n^*(0))$ are unbiased estimates for the average potential outcomes $(\mu^*(1), \mu^*(0)) = (E[Y_i^1], E[Y_i^0])$ of the population.
As the treatment is binary in this example, the set of policies is denoted by $\Theta = \{0, 1\}$ and $\hat{\theta}_n = \text{argmax}_{\theta \in \Theta} Z_n^*(\theta)$ is the policy yielding the highest average outcome[21].
Assuming

$$\begin{pmatrix} Z(0) \\ Z(1) \end{pmatrix} \sim N \left(\begin{pmatrix} \mu(0) \\ \mu(1) \end{pmatrix}, \begin{pmatrix} \Sigma(0) & 0 \\ 0 & \Sigma(1) \end{pmatrix} \right) \tag{15}$$

with known variances $\Sigma(0)$ and $\Sigma(1)$. With binary treatments, from $\hat{\theta}_n = \text{argmax}_{\theta \in \Theta} Z_n^*(\theta)$ follows that $\hat{\theta} = 1$ if $Z(1) > Z(0)$. Conditional on $\hat{\theta} = 1$ and $Z(0) = z(0)$[22], $Z(1)$ follows a normal distribution truncated at $z(0)$, meaning there can not be any values of $Z(1)$ observed smaller than $z(0)$ (as then $\hat{\theta} \neq 1$). This holds for all valid $z(0)$, therefore $Z(1)$ has a positive median bias

[21] As in this example there are only two possible policies, Here $\hat{\theta}_n = \text{argmax}_{\theta \in \Theta} Z_n^*(\theta)$ is 1 if $Z_n^*(1) > Z_n^*(0)$ and 0 if $Z_n^*(1) < Z_n^*(0)$. Ties occur with probability zero.
[22] $z(0)$ is an arbitrary number

conditional on $\hat{\theta} = 1$:

$$\mathrm{P}_{\mu}\{Z(\hat{\theta}) \geq \mu(\hat{\theta}) \mid \hat{\theta} = 1\} > \frac{1}{2} \text{ for all } \mu. \tag{16}$$

Moreover, as this holds symmetrically for $\hat{\theta} = 0$ and $Z(0) > Z(1)$, $\hat{\theta}$ has an unconditional positive median bias:

$$\mathrm{P}_{\mu}\{Z(\hat{\theta}) \geq \mu(\hat{\theta})\} > \frac{1}{2} \text{ for all } \mu. \tag{17}$$

This means while $Z_n^*(\theta)$ is an unbiased estimate for $\mu^*(\theta)$ if policies are fixed, if policies are assigned according to the outlined estimations, $Z_n^*\left(\hat{\theta}_n\right)$ systematically overestimates $\mu^*\left(\hat{\theta}_n\right)$. This stylized example can be expanded for a higher number of policies (e.g. additional treatments). As for all policies there is a conditional bias (equation 16), the unconditional bias (equation 17) also holds.

Furthermore, while this example covers average potential outcome, one can easily see that it also holds for average treatment effects. Assuming two treatments and one control and choosing the treatment with the highest average treatment effect $\hat{\theta}_n = \mathrm{argmax}_{\theta \in \{1,2\}} \left(Z_n^*(\theta) - Z_n^*(0)\right)$, conditional on $\hat{\theta} = 1$ and $Z(1) - Z(0) > Z(2) - Z(0) = Z(1) > Z(2)$ with $Z(2) = z(2)$, $Z(1)$ then follows a normal distribution truncated at $z(2)$ and the positive bias still persists. Applied the conditional average treatment effect outlined in Section 3.1, for a fixed individual i, a truncation of the distribution of estimated heterogeneous treatment effect when selected as the optimal treatment also exists.

From the stylized example, it can be seen that overestimation leads to a treatment more likely to be chosen as the optimal treatment. Irrespective of whether chosen by the average outcome, average treatment effect, or conditional average treatment effect, the optimal treatment is more likely to be overestimated than underestimated.

6.2 Shrinkage Estimators

A possible solution to the winner's curse in the context of optimal individual treatment assignment may be to shrink the estimations towards a common mean. As predictions are shrunken towards a mean and overestimated treatment leading to the respective treatment being chosen as the optimal treatment (the treatment with the highest predicted treatment effect) likely being

over the previously chosen mean, the respective predictions are lowered. If the previously assigned treatment was overestimated and another truly better treatment is assigned, this may increase the performance of the estimator in the previously conducted empirical analysis.
I will outline two shrinkage estimators altered to fit the present treatment effect prediction problem. They mainly differ in whether treatment estimations with a large variation should be shrunken more or less.

6.2.1 James Stein Shrinker

The first shrinkage method I will outline is based on the James Stein Shrinkage estimator (Efron and Morris, 1977). The Stein Paradox states that (disregarding any covariates) using the past averages as estimates for future averages (or the true underlying averages) may not always be the best estimator. More precisely, the authors find that by shrinking past averages towards a common overall average, estimators have a lower squared error and are closer to the actual future average for 16 of their 18 examined observations.
The concept of this estimator can also be applied to heterogeneous treatment effects. Suppose we have estimates for the treatment effects of $i = 1, \ldots, n$ individuals for $k = 1, \ldots, K$ treatments, $\hat{\tau}_i{}^k$ (given e.g. by aforementioned estimators in chapter 3.2.1). Then the shrinkage estimator $\hat{\phi}^k_{i,JS}$ for the heterogeneous treatment effect is given by

$$\hat{\phi}^k_{i,JS} = \overline{\hat{\tau}^k} + c^k_{JS}(\hat{\tau}^k_i - \overline{\hat{\tau}^k}), \quad 1, \ldots, n \tag{18}$$

where

$$\overline{\hat{\tau}^k} = \frac{1}{n}\sum_i^n \hat{\tau}^k_i \tag{19}$$

and the shrinking factor c^k for the respective treatment k is

$$c^k_{JS} = 1 - \frac{(n-3)\sigma^2_{ATE}}{\sum_{i=1}^n (\hat{\tau}^k_i - \overline{\hat{\tau}^k})^2}, \tag{20}$$

with σ^2_{ATE} being the variance of the average treatment effect[23] of being treated by any of the six treatments. Therefore, if predictions of the treatment have a high variance in comparison with the variance of the overall treatment effect, c

[23]Measured by the squared standard error of the coefficient β_1 in a dummy OLS regression using the whole data set $y_i = \beta_0 + \beta_1 \times \tilde{T}_i + \epsilon_i$, where $\tilde{T}_i$ indicates whether the individual was treated by any considered treatment. Here, the training set is used.

gets closer to one and $\hat{\phi}^k_{i,JS}$ closer to $\hat{\tau}^k_i$, meaning the predictions of the model for the respective treatment are shrunken less. I will refer to this method as "James-Stein Shrinker" (JS Shrinker).

6.2.2 Variance Shrinker

The second shrinker I will use is based on the work of Chen and Zimmermann (2020). They use a shrinkage estimator to adjust for an upward bias in published stock returns of academic journals, which appears to originate from journals preferring stock return predictors that produce large t-statistics and subsequently prefer large sample mean returns.
The shrinker is given by

$$\hat{\phi}^k_{i,VS} = (1 - c^k_{VS})\hat{\tau}^k_i + c^k_{VS}\overline{\hat{\tau}^k} \tag{21}$$

with

$$c^k_{VS} = \frac{\sigma_k}{\sigma_{ATE} + \sigma_k}, \tag{22}$$

where σ_{ATE} again is the variance of the overall average treatment effect and σ_k is the variance of the average treatment effect of treatment k[24]. For increasing σ_k and σ_{ATE} being fixed, c^k_{VS} increases and approaches one, s.t. $\hat{\phi}^k_{i,VS}$ is shrunken more and approaches the mean of predictions in treatment k, $\overline{\hat{\tau}^k}$. I will refer to this shrinkage method as "Variance Shrinker".

6.2.3 Shrinkage Method Variation

As described, both shrinkage methods shrink towards the mean of the predicted treatment effects for the respective treatment,

$$\overline{\hat{\tau}^k} = \frac{1}{n}\sum_i^n \hat{\tau}^k_i.$$

If suspecting that getting treated at all by one of the presented six (or four) treatments is the main effect and that the treatments do not differ that much, it may be sensible to shrink towards a common overall mean and not towards a treatment-specific mean. Therefore, for both shrinkage methods, I will also

[24]Measured by the squared standard error of the coefficient β_1 in a dummy OLS regression $y_i = \beta_0 + \beta_1 \times T^k_i + \epsilon_i$, where T^k_i indicates whether the individual was treated by treatment k, using only observations from the control group and those that were treated by treatment k. This is done with the training set.

introduce an altered version, shrinking towards the average treatment effect of all considered treatments, where the altered James-Stein Shrinker shrinking towards the overall mean is given by

$$\hat{\phi}^k_{i,JS'} = \overline{\hat{\tau}} + c^k_{JS'}(\hat{\tau}^k_i - \overline{\hat{\tau}}), \quad 1, \ldots, n \tag{23}$$

where $\overline{\hat{\tau}}$ is the average treatment effect (evaluated in the training set) of all considered treatments,

$$\overline{\hat{\tau}} = \frac{1}{N(T_i = 1)} \sum_{i=1}^{n} T_i Y_i - \frac{1}{N(T_i = 0)} \sum_{i=1}^{n} (1 - T_i) Y_i, \tag{24}$$

where T_i denotes whether the participant was treated with any (considered) treatment. The shrinking factor c^k for the respective treatment k is

$$c^k_{JS'} = 1 - \frac{(n-3)\sigma^2_{ATE}}{\sum_{i=1}^{n}(\hat{\tau}^k_i - \overline{\hat{\tau}^k})^2}. \tag{25}$$

The altered Variance Shrinker is given by

$$\hat{\phi}^k_{i,VS'} = (1 - c^k_{VS'})\overline{\hat{\tau}} + c^k_{VS'}\hat{\tau}^k_i \tag{26}$$

with

$$c^k_{VS'} = \frac{\sigma_{ATE}}{\sigma_{ATE} + \sigma_k}. \tag{27}$$

6.3 Empirical Results

Like in section 5, I will first report the results when using all six treatments and then the results of the subset of treatments.

6.3.1 Results Using All Six Treatments

Using all six treatments, Table 4 shows the average outcome of matched observations for all three estimation methods with the applied shrinkage methods. Figures 7, 8 and 9 depict the distributions of the three respective methods in combination with the four shrinkage methods.

For the James-Stein Shrinker shrinking towards the average treatment effect prediction of the respective treatment, the average outcome of matched observations is slightly higher than of the baseline (not shrunken) methods for both forest-based estimators with 1963 compared to 1959 for the virtual twin ran-

dom forest and 1988 compared to 1982 for the causal forest. Furthermore, for the virtual twin random forest, the number of repetitions in which the average points of matched observations is above the average outcome of treatment four increases from 28 to 35 (out of 100) and for the causal forest from 81 to 91 (out of 100), which marks the best overall performance out of all specifications and methods in the empirical analysis. The performance measured by both metrics slightly decreases for the causal net.

The James-Stein Shrinker shrinking towards the overall average treatment effect of all six treatments decreases the performance for all three methods drastically. In no repetition for the causal forest, the average points of matched observations are above the average outcome of treatment four and only in 1/100 repetitions for the VTRF. The average points of matched observations of the causal forest are above the overall average outcome of all six treatments only for 68/100 repetitions, for the VTRF in 97/100. Out of all combinations of methods and shrinkers, this shrinker applied to the predictions of the causal net performs the worst. Only 16/100 repetitions are over the overall average outcome, and none above the average outcome of treatment four. The average points of matched observations is 1864, which is below every average outcome of the treatments but of treatment three.

The Variance Shrinker shrinking towards the average predicted treatment effect of the respective treatment increases the average points of matched observations of the VTRF from 1959 to 1962 and the number of repetitions the average matched observation is above the average outcome of treatment four from 28/100 to 37/100. Both metrics remain almost unchanged for the causal net. The shrinker decreases the average points of matched observations of causal forest from 1982 to 1975 with now 66/100 instead of 81/100 repetitions over the average outcome of treatment four.

The Variance Shrinker shrinking towards the overall average treatment effect decreases the average points of matched observations and the number of repetitions the average points of matched observations are above the overall average outcome and above the average outcome of treatment four for all estimators well below the initial values of the baseline methods.

The shrinkage methods perform very differently between the machine learning methods. While for the causal net the already poor performance could not be increased meaningfully and even got worse, the tree-based methods saw performance increases with applied shrinkers. This could be the case because for the tree-based methods, the winner's curse was (or still is) a problem hindering

the performance of optimal treatment assignment. With the causal net, however, because of the overall poor performance, the winner's curse likely is not the most pressing issue restricting the performance. The shrinkers shrinking towards the overall mean for all three methods and both shrinkers decreased the metrics. Presumably the differences between the treatments were so large, such that estimations got shrunken too much when shrunken towards an overall common mean.

6.3.2 Results of Subset of Treatments

The results using the subset of treatments (treatments one, two, four, and five) are depicted in Table 5 with distributions of repetition average outcomes being depicted in figures 10, 11 and 12.
As stated in Section 6.2.3, one would expect the shrinkers shrinking towards the overall average treatment effect instead of the treatment specific average predicted treatment effect to perform better if the true underlying treatment effects do not differ that much between the treatments. In the case of using all six treatments, as outlined in Section 4.1, this seems far-fetched, which is likely a reason why the respective shrinkers perform worse than the predictions of the baseline methods and the predictions being shrunken toward the individual average outcomes of the treatments. In line with this reasoning, shrinkers shrinking towards the overall average treatment effect performed much better in the analysis using only treatments 1, 2, 4, and 5.
For the VTRF, both the JS-Shrinker shrinking towards the overall and shrinking towards the treatment-specific mean slightly improve the average points of matched observations from 1970 (baseline method) to 1973. The repetitions in which the average points of matched observations of the VTRF were over the average outcome of treatment four improved by using the JS-Shrinker from 52/100 to 61/100 (shrinking toward the treatment-specific mean) and to 58/100 (shrinking toward the overall mean). For the Variance-Shrinker, both metrics stay almost the same when shrinking towards the overall mean and decrease to 1963 average points of matched observations and 89/100 repetitions over the average points of all four used treatments (baseline method: 97/100) and 39/100 over the average points of treatment four.
Very similar to the full treatment set, no shrinker can provide meaningful performance increases to the predictions of the causal net. For all shrinker variations, the average points of matched observations decreased. While the

JS-Shrinker shrinking towards the overall mean of the four used treatments improved the number of repetitions where the average points of matched observations were above the average outcome of treatment four to 18/100 instead of previously 8/100, it also increased the number where it was below the average points of all four treatments in 45/100 instead of 31/100 repetitions. This is because using this shrinker, the average points of matched observations vary much more between repetitions than applying the other shrinkers to the causal net predictions (see Figure 11).
Regarding the causal forest, similar to the results of the full treatment set, the JS-Shrinker could provide noteworthy performance increases. While the JS-Shrinker shrinking towards the average treatment effect over all used treatments performed poorly in the full treatment set, here it improved the average outcome of matched observations to 1987 (baseline method: 1980) and the number of repetitions where it was over the average points of treatment four to 89/100. When shrinking towards the average predicted treatment effect, improvements were marginally lower with 1986 and 88/100, respectively. Coinciding with the findings using the full treatment set, the Variance Shrinker could not improve the predictions of the causal forest substantially, as the metrics for the shrinker shrinking towards the overall mean of the used treatments stayed almost the same, and the metrics when shrinking towards the treatment-specific mean decreased to 1974 average points of matched observations and 65/100 repetitions where the average points were above the average points of treatment four.
The shrinkers shrinking towards the overall ATE of the used treatments indeed performed much better when applied to treatments with similar outcome levels. In the subset of treatments, the James-Stein-Shrinker performed better for both tree-based methods than the Variance-Shrinker, with relative performance between shrinkers being much more ambiguous in analysis using the full treatment set.

7 Conclusion

With causal inference gaining in popularity in recent years, assignment of optimal treatment using predicted heterogeneous treatment effects of machine learning methods is a natural field of interest with vast areas of application. Comparing the VTRF, the causal net, and the causal forest with the intro-

duced Misra-Matching on an empirical data set, the causal forest assigns the optimal treatment the best. The results indicate that optimal treatment by the highest predicted treatment effect of the causal forest yields a higher outcome level than assigning only the best overall performing treatment. The VTRF and the causal net perform only better than random treatment assignment. Only in the case of the reduced subset of treatments, the results indicate that optimal treatment assignment via the VTRF is at least as good as only assigning treatment four. Usage of the VTRF however should not necessarily be discarded. The VTRF method is easier to understand and implement than the causal forest for users not familiar with causal inference, especially considering that no proxy error terms like τ-risk $_R$ are needed for hyperparameter tuning. Opitz et al. (2022) found that using the very same dataset to train VTRFs to assign optimal treatments along treatments four, five, and six for a second experiment round, the model yielded significantly higher average points than only assigning treatment four. One reason for the overall worse performance may be that the VTRF overfits, as the VTRF yielded the highest average outcome when observations were matched on the training set. Further research implementing the τ-risk $_R$ metric for tuning the VTRF, e.g. with a more sophisticated algorithm than grid search, could give insight into whether overfit can be reduced when using τ-risk $_R$.

The causal net performs considerably worse than the other two methods. There may be various reasons for this. The number of observations with around 1160 for each network[25] may be too low, considering that neural networks also need to split into a validation sample. Furthermore, tuning neural networks is difficult, as the interdependence of hyperparameters is higher than for the introduced tree-based methods. The used hyperparameter optimization and the proxy for the error ($\tau - risk_R$) may not have been suiting for the causal net[26]. Moreover, the chosen hyperparameter grids may have not been optimal for the data set and could be expanded, which was however computationally unfeasible for this study.

The winner's curse, which means that optimal treatments are systematically overestimated, is a major challenge to optimal treatment assignment via predicted HTEs. To address this issue, I introduced the James-Stein and the Vari-

[25] On average 864 for each treatment and 879 for the control group. This makes around 1740 for each network, and $1740 * \frac{2}{3} = 1160$ for each training set.

[26] This may be the case as Schuler et al. (2018) did use Elastic Net and Gradient Boosting Trees for their comparison, with the latter one sharing large similarities with the here used tree methods. The authors did examine neural networks in their paper.

ance shrinker. The usage of shrinkage methods could improve the performance of the predictions in some cases. The shrinkers performed differently across the models and subset of treatments, with the shrinkers shrinking towards the overall average outcome performing very poorly in the analysis using all six treatments and much better in the subset using the four similar treatments. Overall, the James-Stein Shrinker yielded higher performance increases than the Variance Shrinker. The two tree-based machine learning methods (VTRF and causal forest) could benefit considerably from applied shrinkers, while the performance of the causal net did not see notable improvements.

Differences between the methods and shrinkage methods are small, and effects depend a lot on the chosen split in the cross-validation. Therefore, it has to be cautioned against fully conclusive implications from the empirical results. It could be that the findings are well within the respective confidence bands. There may be various reasons for the effects being small. While MTurk provides, as already outlined in Section 2, benefits over classical experiment setups, it is to be expected that the dataset is noisier and surveys may have been less sincerely filled out than from students who are paid better and may have more interest in the research (comparing with a university laboratory setup). Furthermore, the data set may just be too small to sufficiently capture the effects or allow the machine learning model to be properly trained (and then compared). Further experiments with an increased number of observations, ideally with higher ensured quality of filled out surveys, would be interesting to examine for more compelling insights. Another reason may be that the chosen treatments and the experiment setup are not suited to allow for sizable variation in heterogeneous treatment effects, but findings from Opitz et al. (2022) indicate otherwise.

Future research should focus on understanding the mechanisms behind the improvement of the predictions by using shrinkage methods. While shrinkers do improve the optimal treatment assignment of the models, the mechanisms are not clear. Especially as shrinkers perform differently between respective methods and between the used set of used treatments, understanding of the mechanisms is necessary. Furthermore, used shrinkers only borrow from previous shrinkage concepts and are not based on proven good asymptotic behavior in context of estimating treatment effects for treatment assignment. If working mechanisms were understood, optimal shrinkers for addressing the winner's curse in optimal treatment assignment via machine learning methods could be developed.

References

Andrews, Isaiah, Toru Kitagawa, and Adam McCloskey, "Inference on winners," Technical Report, National Bureau of Economic Research 2019.

Athey, Susan and Guido Imbens, "Recursive partitioning for heterogeneous causal effects," *Proceedings of the National Academy of Sciences*, 2016, *113* (27), 7353–7360.

_ **, Julie Tibshirani, and Stefan Wager**, "Generalized random forests," *The Annals of Statistics*, 2019, *47* (2), 1148–1178.

Breiman, Leo, "Out-of-bag estimation," 1996.

_ , "Random forests," *Machine learning*, 2001, *45* (1), 5–32.

_ **, Jerome H Friedman, Richard A Olshen, and Charles J Stone**, *Classification and regression trees*, Routledge, 1984.

Chen, Andrew Y and Tom Zimmermann, "Publication bias and the cross-section of stock returns," *The Review of Asset Pricing Studies*, 2020, *10* (2), 249–289.

DellaVigna, Stefano and Devin Pope, "What motivates effort? Evidence and expert forecasts," *The Review of Economic Studies*, 2018, *85* (2), 1029–1069.

Efron, Bradley and Carl Morris, "Stein's paradox in statistics," *Scientific American*, 1977, *236* (5), 119–127.

Farrell, Max H, Tengyuan Liang, and Sanjog Misra, "Deep neural networks for estimation and inference," *Econometrica*, 2021, *89* (1), 181–213.

Follmer, D Jake, Rayne A Sperling, and Hoi K Suen, "The role of MTurk in education research: Advantages, issues, and future directions," *Educational Researcher*, 2017, *46* (6), 329–334.

Foster, Jared C, Jeremy MG Taylor, and Stephen J Ruberg, "Subgroup identification from randomized clinical trial data," *Statistics in medicine*, 2011, *30* (24), 2867–2880.

Friedman, Jerome H, *The elements of statistical learning: Data mining, inference, and prediction*, springer open, 2017.

Geurts, Pierre, Damien Ernst, and Louis Wehenkel, "Extremely randomized trees," *Machine learning*, 2006, *63* (1), 3–42.

Hirano, Keisuke and Jack R Porter, "Asymptotics for statistical treatment rules," *Econometrica*, 2009, *77* (5), 1683–1701.

Hitsch, Günter J and Sanjog Misra, "Heterogeneous treatment effects and optimal targeting policy evaluation," *Available at SSRN 3111957*, 2018.

Ho, Tin Kam, "Random decision forests," in "Proceedings of 3rd international conference on document analysis and recognition," Vol. 1 IEEE 1995, pp. 278–282.

Imai, Kosuke and Marc Ratkovic, "Estimating treatment effect heterogeneity in randomized program evaluation," *The Annals of Applied Statistics*, 2013, *7* (1), 443–470.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani, *An introduction to statistical learning*, Vol. 112, Springer, 2013.

John, Oliver P, Eileen M Donahue, and Robert L Kentle, "The big five inventory—versions 4a and 54," 1991.

Kitagawa, Toru and Aleksey Tetenov, "Who should be treated? empirical welfare maximization methods for treatment choice," *Econometrica*, 2018, *86* (2), 591–616.

Kleinberg, Jon, Jens Ludwig, Sendhil Mullainathan, and Ziad Obermeyer, "Prediction policy problems," *American Economic Review*, 2015, *105* (5), 491–95.

Lu, Min, Saad Sadiq, Daniel J Feaster, and Hemant Ishwaran, "Estimating individual treatment effect in observational data using random forest methods," *Journal of Computational and Graphical Statistics*, 2018, *27* (1), 209–219.

Manski, Charles F, "Statistical treatment rules for heterogeneous populations," *Econometrica*, 2004, *72* (4), 1221–1246.

Nie, Xinkun and Stefan Wager, "Quasi-oracle estimation of heterogeneous treatment effects," *Biometrika*, 2021, *108* (2), 299–319.

Opitz, Saskia, Dirk Sliwka, Timo Vogelsang, and Tom Zimmermann, "The targeted assignment of incentive schemes," *Preprint*, 2022. Available at SSRN: `http://dx.doi.org/10.2139/ssrn.4077778`, version April 8, 2022.

Robinson, Peter M, "Root-N-consistent semiparametric regression," *Econometrica: Journal of the Econometric Society*, 1988, pp. 931–954.

Schuler, Alejandro, Michael Baiocchi, Robert Tibshirani, and Nigam Shah, "A comparison of methods for model selection when estimating individual treatment effects," *arXiv preprint arXiv:1804.05146*, 2018.

Simester, Duncan, Artem Timoshenko, and Spyros I Zoumpoulis, "Efficiently evaluating targeting policies: Improving on champion vs. challenger experiments," *Management Science*, 2020, *66* (8), 3412–3424.

Su, Xiaogang, Chih-Ling Tsai, Hansheng Wang, David M Nickerson, and Bogong Li, "Subgroup analysis via recursive partitioning.," *Journal of Machine Learning Research*, 2009, *10* (2).

Taddy, Matt, Matt Gardner, Liyun Chen, and David Draper, "A nonparametric bayesian analysis of heterogenous treatment effects in digital experimentation," *Journal of Business & Economic Statistics*, 2016, *34* (4), 661–672.

Wager, Stefan and Susan Athey, "Estimation and inference of heterogeneous treatment effects using random forests," *Journal of the American Statistical Association*, 2018, *113* (523), 1228–1242.

Willke, Richard J, Zhiyuan Zheng, Prasun Subedi, Rikard Althin, and C Daniel Mullins, "From concepts, theory, and evidence of heterogeneity of treatment effects to methodological approaches: a primer," *BMC medical research methodology*, 2012, *12* (1), 1–12.

Appendices

A Out-of-bag Percentage

The probability of drawing an observation (x_i, y_i) from a set of observations $S = \{(x_i, y_i)\}, i = 1, \ldots, n$ is $\frac{1}{n}$, so the probability of not drawing it is $1 - \frac{1}{n}$. Drawing with replacement, the probability of not drawing the observation when drawing n times is $(1 - \frac{1}{n})^n$. Therefore for large enough n,

$$\lim_{n\to\infty} \left(1 - \frac{1}{n}\right)^n = e^{-1} \approx 0.368.$$

As the chance for each individual observation of not being drawn is 37%, only 63% of observations are used on average with bagging.

B Transformed Outcome

Let

$$\begin{aligned} Y_i^* &= Y_i \left(T_i - p\right) / (p(1-p)) \\ &= \frac{Y_i \left(T_i - p\right)}{p(1-p)} \\ &= \frac{Y_i T_i}{p(1-p)} - \frac{Y_i}{1-p} \\ &= \frac{Y_i T_i (1 - p + p)}{p(1-p)} - \frac{Y_i}{1-p} \\ &= \frac{Y_i T_i}{p} + \frac{Y_i T_i}{1-p} - \frac{Y_i}{1-p} \\ &= T_i \frac{Y_i}{p} - (1 - T_i) \frac{Y_i}{1-p} \end{aligned}$$

with $p = P(T_i = 1)$.

Then one can show that $E[Y_i^* | Z_i = x] = E[Y_i | x] = E[Y_i(1) - Y_i(0) | x] = \tau(x)$. Defining $\pi(x) = P(T_i = 1 | x)$ and with $E[P(T_i = 1)|x] = P(T_i = 1|x) = \pi(x)$,

$$
\begin{aligned}
E[Y_i^*|x] =& P\left(T_i = 1 \mid x\right) E\left[Y_i^* \mid x, T_i = 1\right] + P\left(T_i = 0 \mid x\right) E\left[Y_i^* \mid x, T_i = 0\right] \\
=& \pi(x) \cdot E\left(T_i \frac{Y_i}{p} - (1 - T_i) \frac{Y_i}{1-p} \middle| x, T_i = 1\right) \\
&+ (1 - \pi(x)) \cdot E\left(T_i \frac{Y_i}{p} - (1 - T_i) \frac{Y_i}{1-p} \middle| x, T_i = 0\right) \\
=& \pi(x) \cdot \frac{E\left[Y_i \mid x, T_i = 1\right]}{E[P(T_i = 1)|x]} + (1 - \pi(x)) \cdot (-1) \cdot \frac{E\left[Y_i \mid x, T_i = 0\right]}{E[1 - P(T_i = 1)|x]} \\
=& \pi(x) \cdot \frac{E\left[Y_i \mid x, T_i = 1\right]}{\pi(x)} + (1 - \pi(x)) \cdot \frac{-E\left[Y_i \mid x, T_i = 0\right]}{1 - \pi(x)}.
\end{aligned}
$$

As

$$
Y_i = T_i Y_i(1) + (1 - T_i) Y_i(0),
$$

it follows that

$$
\begin{aligned}
E\left[Y_i^* \mid x\right] &= \pi(x) \cdot \frac{E\left[Y_i(1) \mid x\right]}{\pi(x)} + (1 - \pi(x)) \cdot \frac{-E\left[Y_i(0) \mid x\right]}{1 - \pi(x)} \\
&= E\left[Y_i(1) \mid x\right] - E\left[Y_i(0) \mid x\right] \\
&= E\left[Y_i(1) - Y_i(0) \mid x\right] = \tau(x).
\end{aligned}
$$

C Treatment Details

Outlined below are the titles of the treatment and the texts the participants were shown with the description of the treatment.

Pay for Performance (PfP) As a bonus, you will be paid an extra 5 cents for every 100 points that you score.

Goal As a bonus, you will be paid an extra $1 if you score at least 2000 points.

Gift & Goal Thank you for your participation in this study! In appreciation to you performing this task, you will be paid a bonus of $1. In return, we would appreciate if you try to score at least 2,000 points.

Loss As a bonus, you will be paid an extra $1. However, you will lose this bonus (it will not be placed in your account) unless you score at least 2,000 points.

Real-Time Feedback You will receive a bonus that is based on how well you perform relative to others. On your work screen you will see how your current performance compares to that of others who previously performed the task. To that end you will see the percentage of participants who previously performed the task and whom you will outperform at your current speed. You will receive a bonus of \$0.02 times the percentage of participants who performed worse than you at the end of the task. That is, you will for instance receive an additional bonus of \$1.00 (=\$0.02*50) if you perform better than 50% of the participants. The ranking shown on the screen is computed assuming you keep the speed with which you pressed 'a' and 'b' for the past 10 seconds. Your current percentile as well as your currently expected bonus is updated every 10 seconds.

Social PfP As a bonus, you will be paid an extra 3 cents for every 100 points that you score. On top of that, 2 cents will go to Doctors Without Borders for every 100 points.

Control Your score will not affect your payment in any way.

D Model Selection Criterion τ-risk$_R$

Assuming the expectation of the outcome given X_i is $m(X_i) = E[Y_i|X_i]$, the propensity score is $p(X_i) = E[T_i|X_i]$ and the (heterogeneous) treatment effect is $\tau(X_i) = E[Y_i|X_i, T_i = 1] - E[Y_i|X_i, T_i = 0] = E[Y_i^1|X_i] - E[Y_i^0|X_i]$. Then with $E\left[\varepsilon_i\left(W_i\right) \mid X_i, T_i\right] = 0$,

$$\begin{aligned} Y_i &= Y_i^0 + T_i\tau(X_i) + \varepsilon_i \\ \varepsilon_i &= Y_i - Y_i^0 - T_i\tau(X_i) \\ \varepsilon_i - p(X_i)\tau(X_i) &= Y_i - [Y_i^0 + p(X_i)\tau(X_i)] - T_i\tau(X_i) \end{aligned} \tag{28}$$

as $m(X_i) = E[Y_i|X_i] = Y_i^0 + p(X_i)\tau(X_i)$,

$$\begin{aligned} \varepsilon_i - p(X_i)\tau(X_i) &= Y_i - m(X_i) - T_i\tau(X_i) \\ \varepsilon_i &= Y_i - m(X_i) - (T_i - p(X_i))\tau(X_i) \end{aligned} \tag{29}$$

such that

$$\varepsilon_i^2 = ((Y_i - m(X_i)) - (T_i - p(X_i))\tau(X_i))^2. \tag{30}$$

Therefore, minimization of τ-risk$_R$ leads to minimization of the squared error term in $Y_i = Y_i^0 + T_i\tau(X_i) + \varepsilon_i$.

E Hyperparameter Grids

As hyperparameter tuning is performed for all models with a grid-search algorithm, hyperparameter grids have to be specified beforehand. The algorithm iterates over all possible combinations and selects the combination with the lowest loss function value. I will outline the parameters lists for the hyperparameters that were tuned and the parameters that were set but not tuned for each used machine learning method below.

E.1 Virtual Twin Random Forest

Tuned Hyperparameters

- *max_features*: 0.2, 0.3,. . . , 0.9, 1.0
- *max_samples*: 0.1, 0.2, 0.3, 0.4, 0.5
- *min_samples_leaf*: 2, 5, 10, 20, 50
- *max_depth*: 5, 10, 25, 50, 75, 100, None

Hyperparameters not tuned

- *n_estimators*: 1000
- *random_state*: 42

All other parameters are their default value, which can be viewed in the *scikit-learn* documentation. No scaling was done for this model.

E.2 Causal Forest

Tuned Hyperparameters

- *max_features*: 0.2, 0.3,. . . , 0.9, 1.0
- *max_samples*: 0.1, 0.2, 0.3, 0.4, 0.5
- *min_samples_leaf*: 5, 10, 20, 50

- *min_var_fraction_leaf*: 0.1, 0.2, 0.3, 0.4, None
- *max_depth*: 5, 10, 25, 50, 75, 100, None

Hyperparameters not tuned

- *n_estimators*: 1000
- *random_state*: 42

All other parameters are their default value, which can be viewed in the *econml* documentation. No scaling was done for this model.

E.3 Causal Net

Regarding the hyperparameter grids for the causal net, I draw from the values used in the original paper of Farrell et al. (2021) and their used parameters.

Tuned Hyperparameters

- *hidden_layer_size / dropout_rate*
  - 1 Layer, 60 nodes, 50% dropout rate
  - 1 Layer, 100 nodes, 50% dropout rate
  - 2 Layers, L1: 30 nodes with 50% dropout rate, L2: 20 nodes with no dropout
  - 2 Layers, L1: 30 nodes with 30% dropout rate, L2: 10 nodes with 10% dropout rate,
  - 2 Layers, L1: 30 nodes with no dropout, L2: 30 nodes with no dropout
  - 2 Layers, L1: 30 nodes with 50% dropout rate, L2: 30 nodes with no dropout
  - 3 Layers, L1: 100 nodes with 50% dropout rate, L2: 30 nodes with 50% dropout rate, L3: 20 nodes with no dropout
  - 3 Layers, L1: 80 nodes with 50% dropout rate, L2: 30 nodes with 50% dropout rate, L3: 20 nodes with no dropout
- *learning_rate*: 0.1, 0.05, 0.01, 0.001
- *alpha*: 0.01, 0.1, 1 (Regularization Strength parameter)

- *r_par*: 0, 0.3, 0.6, 1 (Mixing ratio of Ridge and Lasso regularization. At 1 equal to Lasso.)

Hyperparameters not tuned

- *optimizer*: Adam
- *batch_size*: None
- *max_epochs_without_change*: 60
- *max_nepochs*: 10000
- *seed* [27]: 42

Furthermore, the covariates were standardized for better learning performance and because Lasso/Ridge kernel regularization was used.

[27]For all functions/methods where it was possible to set a random state/seed, a value of 42 was set for replication purposes.

F Tables

X \ Y	T1	T2	T3	T4	T5	T6	T7
T1	0.5	0.5544	**0.0**	0.9014	0.5344	**0.0468**	**0.0**
T2	0.4456	0.5	**0.0**	0.8942	0.5024	**0.0268**	**0.0**
T3	1.0	1.0	0.5	1.0	1.0	0.9939	**0.0**
T4	0.0986	0.1058	**0.0**	0.5	0.1353	**0.0009**	**0.0**
T5	0.4656	0.4976	**0.0**	0.8647	0.5	**0.0475**	**0.0**
T6	0.9532	0.9732	**0.0061**	0.9991	0.9525	0.5	**0.0**
T7	1.0	1.0	1.0	1.0	1.0	1.0	0.5

Table 1: This table contains the p-values for the Wilcoxon Rank Sum tests for the hypothesis $H_0 : P(X > Y) = P(Y > X) \quad vs. \quad H_1 : P(X > Y) > P(Y > X)$ with X being the points of the sample where the treatment corresponds to the treatment of the respective row value, and Y being the points of the sample where the treatment corresponds to the treatment of the column value. Treatment seven is the control group. p-values that are less than 0.05 are highlighted in bold.

	T1	T2	T3	T4	T5	T6	Overall
VTRF	1935	1925	1932	1815	2135	1974	1959
	(9848)	(12328)	(919)	(27381)	(25300)	(10005)	(85781)
CausalNet	1938	1963	1791	1957	1977	1796	1921
	(10431)	(5927)	(4008)	(23265)	(25338)	(17148)	(86117)
CausalForest	2136	1895	2063	1788	2194	2133	1982
	(7721)	(5950)	(25)	(37917)	(30109)	(2280)	(84002)
Average	1926	1930	1764	1970	1931	1871	1898
	(87900)	(86500)	(87500)	(84800)	(87400)	(84500)	(518600)

Table 2: *Average Outcome of Matched Observations: Full Treatment Set & Not Shrunken:* This table shows the results of the 100 times repeated three-fold cross-validation of the Misra-Matching. All six treatments were considered. For each of the three machine learning methods, it depicts the average outcome of matched observations over all folds and repetitions and in brackets the number of observations that were matched in total. This is shown for each treatment and all treatments. The last row depicts the average points for the participants in the respective treatments and the overall average.

	T1	T2	T4	T5	Overall
VTRF	1971	1954	1831	2136	1970
	(12102)	(15824)	(30825)	(27436)	(86187)
CausalNet	1910	1932	1948	1962	1945
	(14266)	(9394)	(29754)	(32824)	(86238)
CausalForest	2145	1926	1786	2185	1980
	(8606)	(6443)	(38062)	(30855)	(83966)
Average	1926	1930	1970	1931	1939
	(87900)	(86500)	(84800)	(87400)	(346600)

Table 3: *Average Outcome of Matched Observations: Subset of Treatments & Not Shrunken:* This table shows the results of the 100 times repeated three-fold cross-validation of the Misra-Matching. Only treatments 1, 2, 4, and 5 were considered. For each of the three machine learning methods, it depicts the average outcome of matched observations over all folds and repetitions and in brackets the number of observations that were matched in total. This is shown for each treatment and all treatments. The last row depicts the average points for the participants in the respective treatments and the overall average.

	T1	T2	T3	T4	T5	T6	Overall
VTRF	1935	1925	1932	1815	2135	1974	1959
	(9848)	(12328)	(919)	(27381)	(25300)	(10005)	(85781)
VTRF JS $\overline{\hat{\tau}_k}$	1934	1925	1953	1817	2142	1976	1963
	(9837)	(11836)	(600)	(28004)	(26111)	(9375)	(85763)
VTRF JS $\overline{\hat{\tau}}$	1931	1928	1851	1782	2154	1921	1928
	(60252)	(11564)	(135)	(8093)	(4340)	(2322)	(86706)
VTRF Var $\overline{\hat{\tau}_k}$	1948	1956	-	1850	2133	2000	1962
	(9499)	(12803)	(0)	(37013)	(24775)	(1909)	(85999)
VTRF Var $\overline{\hat{\tau}}$	1865	1877	1870	1762	2120	1942	1938
	(12811)	(9744)	(5097)	(14678)	(24115)	(18509)	(84954)
CausalNet	1938	1963	1791	1957	1977	1796	1921
	(10431)	(5927)	(4008)	(23265)	(25338)	(17148)	(86117)
CausalN. JS $\overline{\hat{\tau}_k}$	1924	1921	1758	1942	1969	1852	1918
	(9588)	(6013)	(4420)	(22359)	(25267)	(18282)	(85929)
CausalN. JS $\overline{\hat{\tau}}$	1924	1940	1773	1962	1930	1871	1864
	(8275)	(10438)	(23454)	(5084)	(2037)	(36913)	(86201)
CausalN. Var $\overline{\hat{\tau}_k}$	1924	1940	1767	1966	1949	1843	1922
	(10586)	(6350)	(4049)	(23674)	(25913)	(15441)	(86013)
CausalN. Var $\overline{\hat{\tau}}$	1935	1951	1729	1967	2002	1797	1889
	(9257)	(4485)	(12597)	(12101)	(22722)	(23962)	(85124)
CausalForest	2136	1895	2063	1788	2194	2133	1982
	(7721)	(5950)	(25)	(37917)	(30109)	(2280)	(84002)
CausalF. JS $\overline{\hat{\tau}_k}$	2161	1959	2089	1796	2197	2126	1988
	(6806)	(5019)	(4)	(40159)	(31452)	(696)	(84136)
CausalF. JS $\overline{\hat{\tau}}$	1933	1931	1798	1895	-	1870	1910
	(21160)	(34824)	(803)	(3486)	(0)	(25997)	(86270)
CausalF. Var $\overline{\hat{\tau}_k}$	2141	2004	-	1846	2222	2073	1975
	(7697)	(4517)	(0)	(50868)	(20661)	(149)	(83892)
CausalF. Var $\overline{\hat{\tau}}$	1894	1802	1905	1709	2158	1919	1939
	(14465)	(8908)	(2340)	(16361)	(27492)	(13984)	(83550)
Average	1926	1930	1764	1970	1931	1871	1898
	(87900)	(86500)	(87500)	(84800)	(87400)	(84500)	(518600)

Table 4: *Average Outcome of Matched Observations: Full Treatment Set & Shrunken:* This table shows the results of the 100 times repeated three-fold cross-validation of the Misra-Matching. All six treatments were considered. For each of the three machine learning methods and in combination with the four introduced shrinkage methods, it depicts the average outcome of matched observations over all folds and repetitions and in brackets the number of observations that were matched in total. This is shown for each treatment and all treatments. The last row depicts the average points for the participants in the respective treatments and the overall average. JS stands for the James Stein Shrinker, Var. for the Variance Shrinker. $\overline{\hat{\tau}_k}$ indicates the shrinkers shrink towards the average treatment prediction of the respective treatment, $\overline{\hat{\tau}}$ indicates they shrink towards the overall (over all used treatments) average treatment effect.

	T1	T2	T4	T5	overall
VTRF	1971	1954	1831	2136	1970
	(12102)	(15824)	(30825)	(27436)	(86187)
VTRF JS $\overline{\hat{\tau}_k}$	1969	1952	1833	2142	1973
	(11876)	(15026)	(31176)	(28071)	(86149)
VTRF JS $\overline{\hat{\tau}}$	1968	1953	1831	2142	1973
	(12505)	(15191)	(30585)	(27832)	(86113)
VTRF Var $\overline{\hat{\tau}_k}$	1957	1964	1852	2133	1963
	(9885)	(13446)	(37599)	(25077)	(86007)
VTRF Var $\overline{\hat{\tau}}$	1971	1954	1833	2135	1970
	(12184)	(15817)	(31121)	(27121)	(86243)
CausalNet	1910	1932	1948	1962	1945
	(14266)	(9394)	(29754)	(32824)	(86238)
CausalN. JS $\overline{\hat{\tau}_k}$	1916	1923	1943	1962	1944
	(13947)	(9747)	(29892)	(32680)	(86266)
CausalN. JS $\overline{\hat{\tau}}$	1911	1887	1955	2009	1944
	(16296)	(20443)	(28438)	(21192)	(86369)
CausalN. Var $\overline{\hat{\tau}_k}$	1918	1926	1963	1939	1942
	(13840)	(9684)	(29071)	(33624)	(86219)
CausalN. Var $\overline{\hat{\tau}}$	1904	1930	1951	1957	1943
	(14451)	(9410)	(29879)	(32641)	(86381)
CausalForest	2145	1926	1786	2185	1980
	(8606)	(6443)	(38062)	(30855)	(83966)
CausalF. JS $\overline{\hat{\tau}_k}$	2163	1984	1794	2191	1986
	(7160)	(5252)	(40102)	(31595)	(84109)
CausalF. JS $\overline{\hat{\tau}}$	2160	1965	1776	2182	1987
	(7894)	(8114)	(35811)	(32440)	(84259)
CausalF. Var $\overline{\hat{\tau}_k}$	2129	2002	1845	2220	1974
	(7979)	(4265)	(50527)	(21145)	(83916)
CausalF. Var $\overline{\hat{\tau}}$	2151	1933	1788	2187	1981
	(8568)	(6397)	(38472)	(30486)	(83923)
Average	1926	1930	1970	1931	1939
	(87900)	(86500)	(84800)	(87400)	(346600)

Table 5: *Average Outcome of Matched Observations: Subset of Treatments & Shrunken:* This table shows the results of the 100 times repeated three-fold cross-validation of the Misra-Matching. Only treatments 1, 2, 4, and 5 were considered. For each of the three machine learning methods and in combination with the four introduced shrinkage methods, it depicts the average outcome of matched observations over all folds and repetitions and in brackets the number of observations that were matched in total. This is shown for each treatment and all treatments. The last row depicts the average points for the participants in the respective treatments and the overall average. JS stands for the James Stein Shrinker, Var. for the Variance Shrinker. $\overline{\hat{\tau}_k}$ indicates the shrinkers shrink towards the average treatment prediction of the respective treatment, $\overline{\hat{\tau}}$ indicates they shrink towards the overall (over all used treatments) average treatment effect.

	T1	T2	T3	T4	T5	T6	overall
VTRF	2315	2325	2526	2241	2302	2505	2339
	(22233)	(30722)	(6296)	(65269)	(52553)	(49024)	(226097)
VTRF JS $\overline{\hat{\tau}_k}$	2298	2321	2534	2240	2301	2507	2336
	(21828)	(29951)	(5648)	(65754)	(53033)	(48727)	(224941)
VTRF JS $\overline{\hat{\tau}}$	2301	2322	2526	2240	2301	2504	2337
	(21383)	(29688)	(6291)	(65604)	(53134)	(49416)	(225516)
VTRF Var $\overline{\hat{\tau}_k}$	2310	2318	2756	2211	2301	2649	2303
	(20482)	(30367)	(10)	(80422)	(51408)	(19640)	(202329)
VTRF Var $\overline{\hat{\tau}}$	2313	2328	2526	2241	2304	2511	2339
	(22087)	(30865)	(5892)	(66867)	(51883)	(47470)	(225064)
CausalN	1925	1929	1770	1969	1931	1871	1904
	(24609)	(59397)	(31939)	(39592)	(5230)	(10829)	(171596)
CausalN. JS $\overline{\hat{\tau}_k}$	1924	1929	1770	1969	1931	1871	1905
	(24624)	(59953)	(31357)	(39592)	(5230)	(10829)	(171585)
CausalN. JS $\overline{\hat{\tau}}$	1923	1929	1769	1969	1929	1871	1901
	(25826)	(56473)	(34269)	(38414)	(5820)	(10829)	(171631)
CausalN. Var $\overline{\hat{\tau}_k}$	1925	1929	1770	1969	1931	1871	1904
	(24609)	(59397)	(31939)	(39592)	(5230)	(10829)	(171596)
CausalN. Var $\overline{\hat{\tau}}$	1925	1929	1770	1969	1931	1871	1904
	(24609)	(59397)	(31939)	(39592)	(5230)	(10829)	(171596)
CausalForest	2543	2398	2576	1966	2394	2608	2243
	(20206)	(16041)	(101)	(73792)	(61361)	(7023)	(178524)
CausalF. $\overline{\hat{\tau}_k}$	2532	2386	2641	1956	2388	2624	2222
	(18080)	(13636)	(35)	(75905)	(62716)	(4550)	(174922)
CausalF. $\overline{\hat{\tau}}$	2541	2389	2594	1953	2373	2601	2223
	(16595)	(12361)	(109)	(73468)	(66663)	(6355)	(175551)
CausalF. $\overline{\hat{\tau}_k}$	2503	2420	0	1965	2441	2715	2174
	(18156)	(11274)	(0)	(99352)	(43650)	(490)	(172922)
CausalF. $\overline{\hat{\tau}}$	2546	2402	2564	1965	2398	2607	2240
	(19688)	(15978)	(93)	(75149)	(60381)	(6672)	(177961)
Average	1926	1930	1764	1970	1931	1871	1898
	(175800)	(173000)	(175000)	(169600)	(174800)	(169000)	(1037200)

Table 6: *Average Outcome of Matched Observations - Training Set: Full Treatment Set & Shrunken:* This table shows the results of the 100 times repeated three-fold cross-validation of the Misra-Matching. In the cross-validation, the models were trained on and predicted on the training set. The test set was left entirely unused. All six treatments were considered. For each of the three machine learning methods and in combination with the four introduced shrinkage methods, it depicts the average outcome of matched observations over all folds and repetitions and in brackets the number of observations that were matched in total. This is shown for each treatment and all treatments. The last row depicts the average points for the participants in the respective treatments and the overall average. JS stands for the James Stein Shrinker, Var. for the Variance Shrinker. $\overline{\hat{\tau}_k}$ indicates the shrinkers shrinking towards the average treatment prediction of the respective treatment, $\overline{\hat{\tau}}$ indicates they shrink towards the overall (over all used treatments) average treatment effect.

G Figures

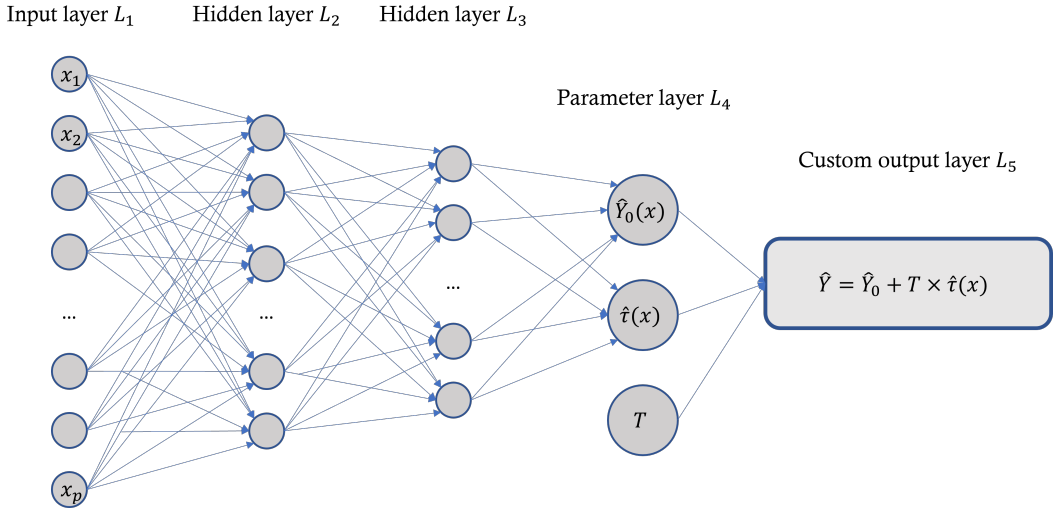

Figure 1: This figure shows an exemplary causal net neural network architecture with two hidden layers.

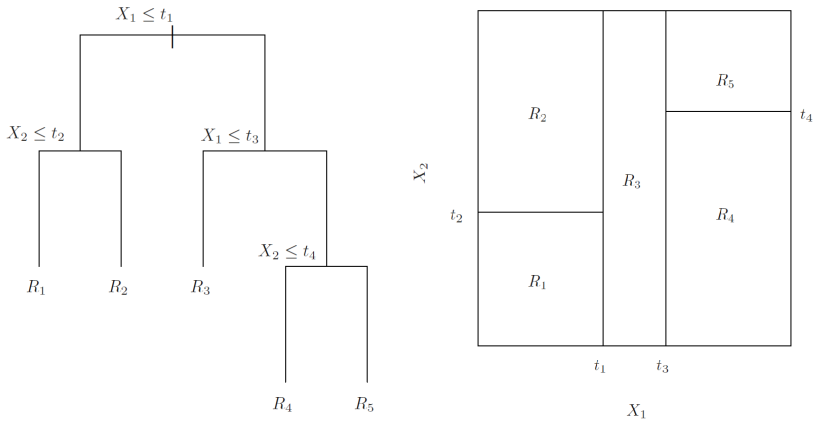

Figure 2: *Visualization Regression Tree:* This figure depicts an exemplary regression tree, splitting along two covariates. Depicted on the left is the tree structure with the four respective binary splits and split thresholds $t_1, \ldots, t_4$, splitting into Regions $R_1, \ldots, R_5$. The resulting partition of the two-dimensional covariate space can be seen on the right-hand side.
Figure source: (James et al., 2013, 332)

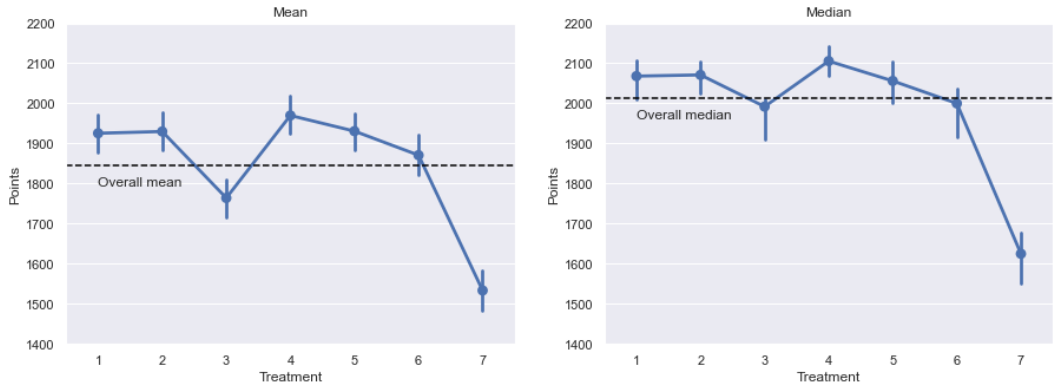

Figure 3: These graphs depict the mean and median outcomes for the respective treatments. The enumeration of treatments is the same as the order listed in Appendix C, with treatment seven being the control group. The overall mean of points is 1845, and the overall median is 2012. The bars represent the bootstrap confidence intervals at the 95% level.

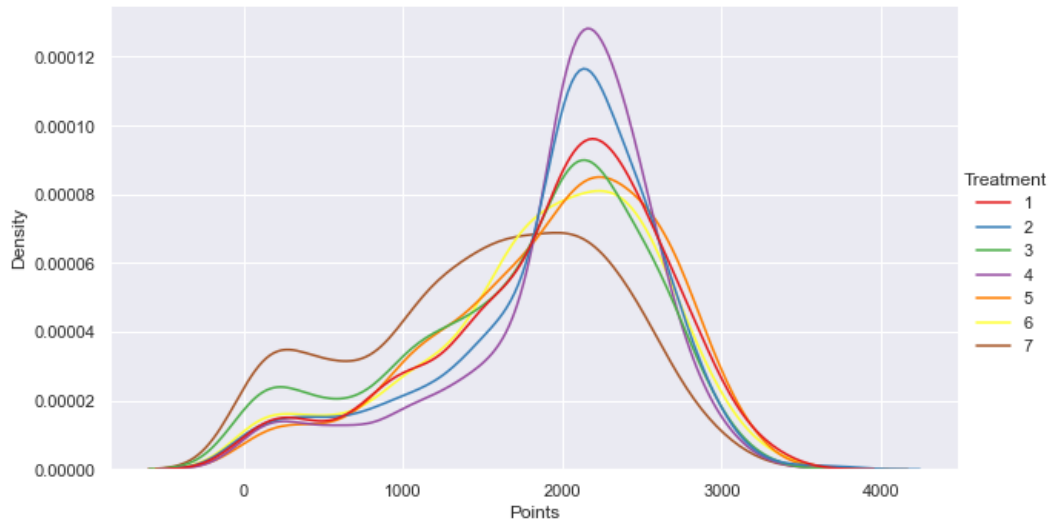

Figure 4: This figure depicts the estimated kernel density functions for the points for each treatment. The functions are colored according to the seven different treatments, to be taken from the adjacent legend. The enumeration of treatments is the same as the order listed in Appendix C, with treatment seven being the control group.

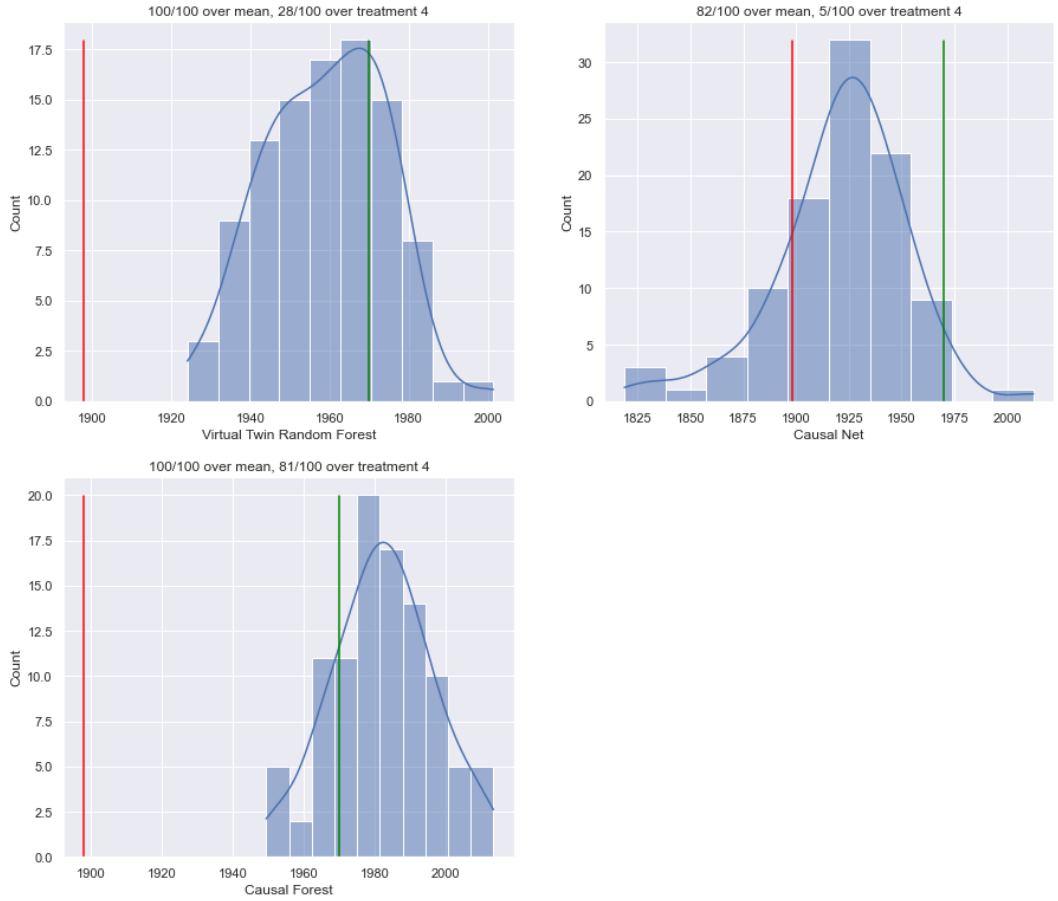

Figure 5: *Misra Matching Baseline Comparison - Full Treatment Set:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation. All six treatments were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the treatments, 1898 points.

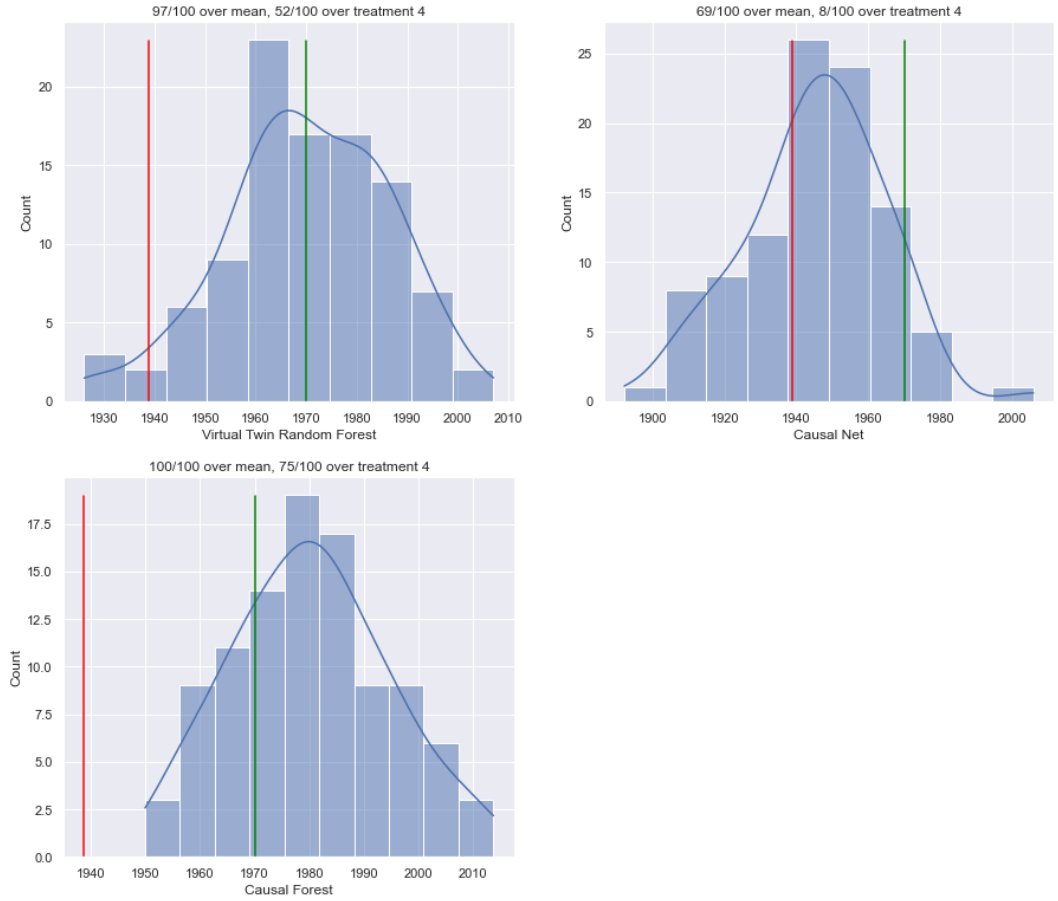

Figure 6: *Misra Matching Baseline Comparison - Subset of Treatments:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation. Only treatments 1, 2, 4, and 5 were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the four considered treatments, 1939 points.

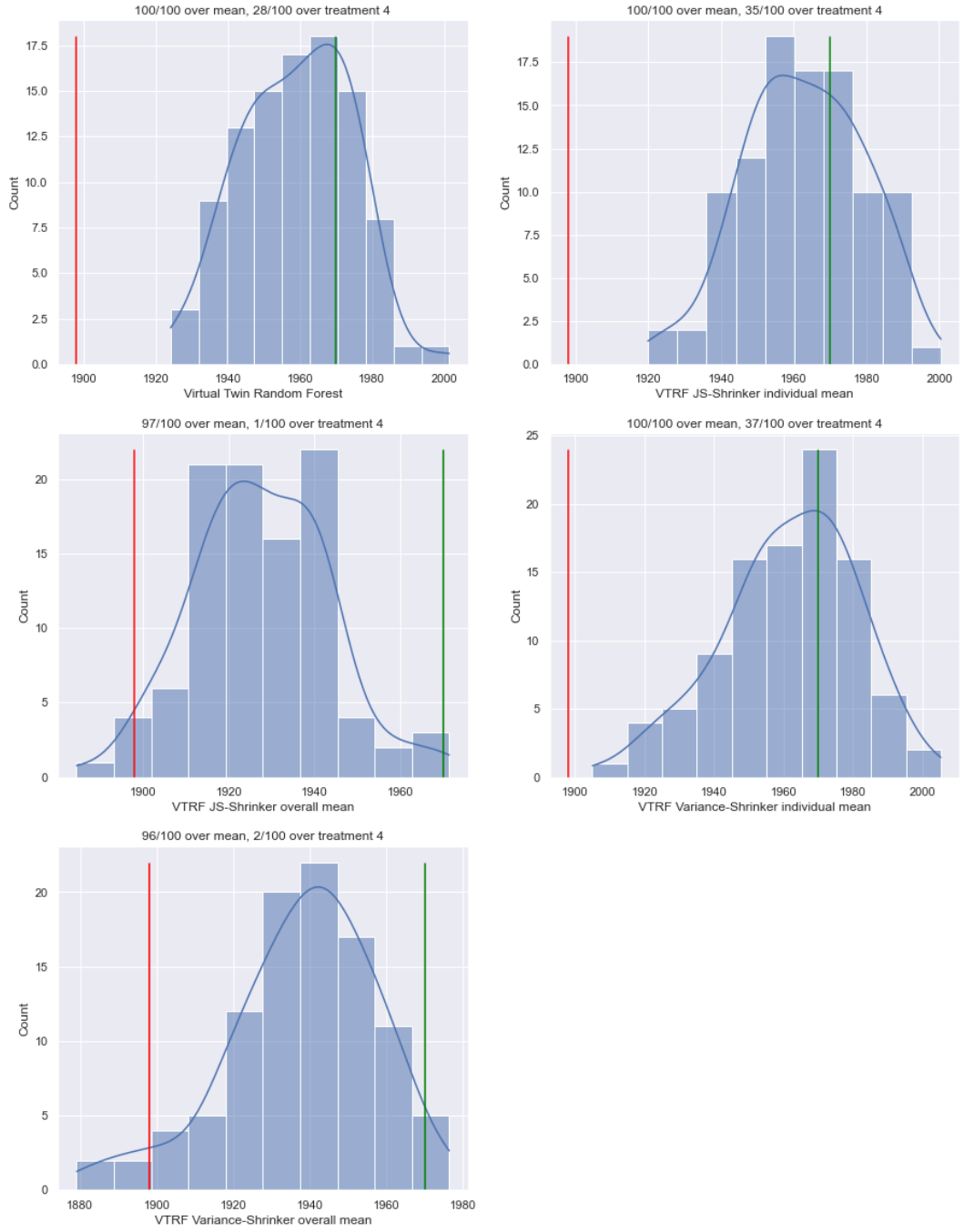

Figure 7: *Misra Matching VTRF: Full Treatment Set & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the virtual twin random forest method and using for all four introduced shrinkage methods. All six treatments were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the treatments, 1898 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).

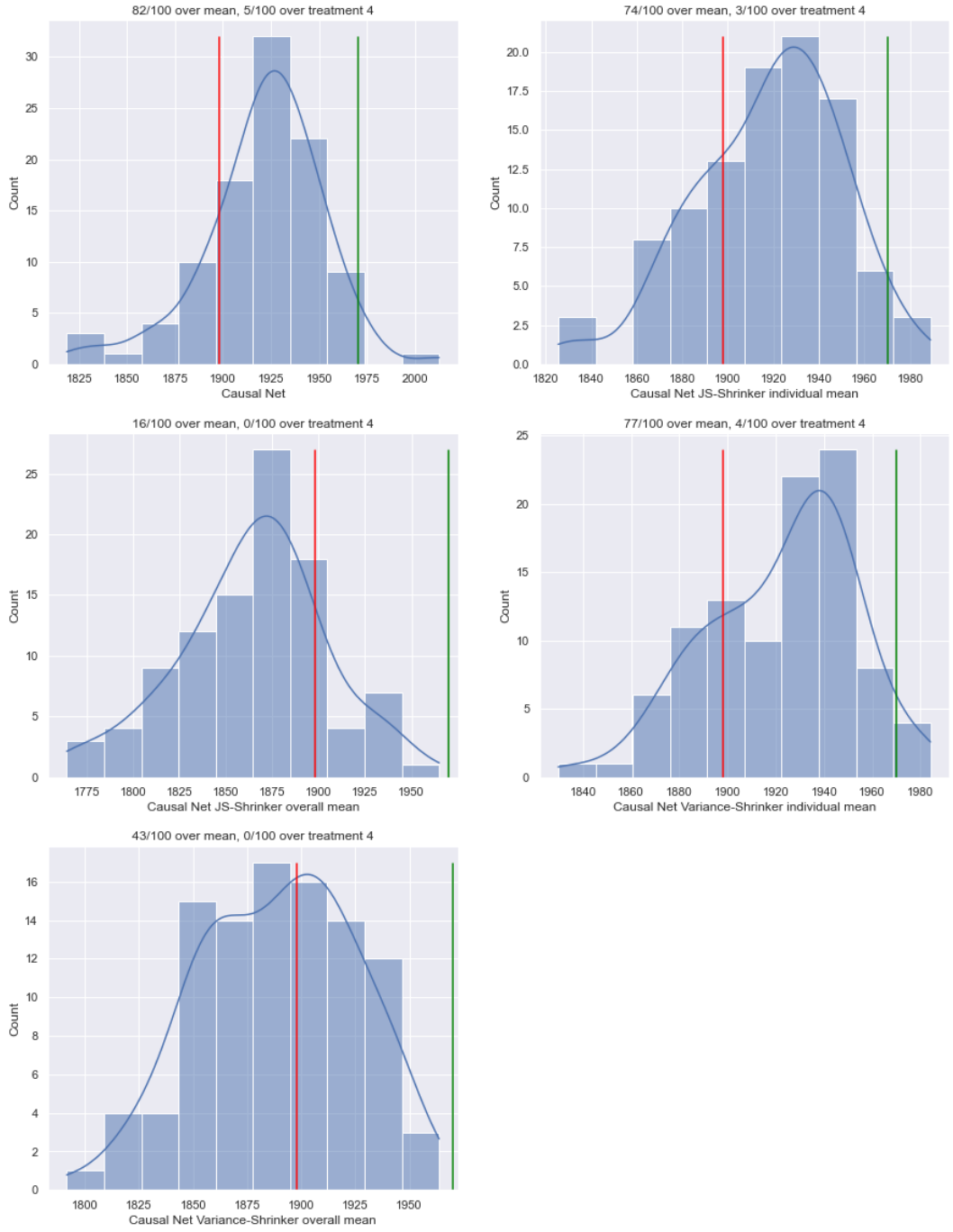

Figure 8: *Misra Matching Causal Net: Full Treatment Set & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the causal net method and using all four introduced shrinkage methods. All six treatments were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the treatments, 1898 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).

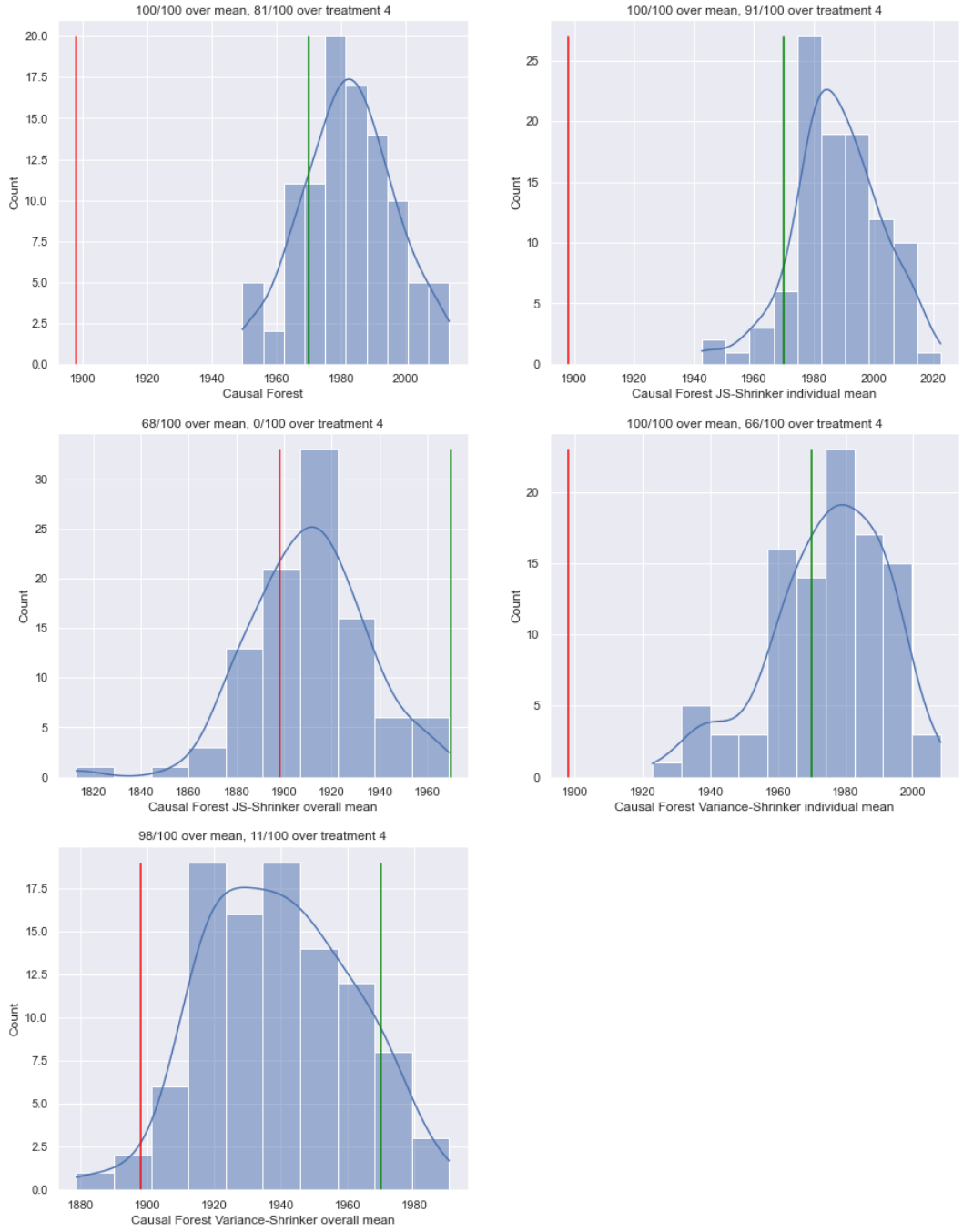

Figure 9: *Misra Matching Causal Forest: Full Treatment Set & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the causal forest method and using all four introduced shrinkage methods. All six treatments were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the treatments, 1898 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).

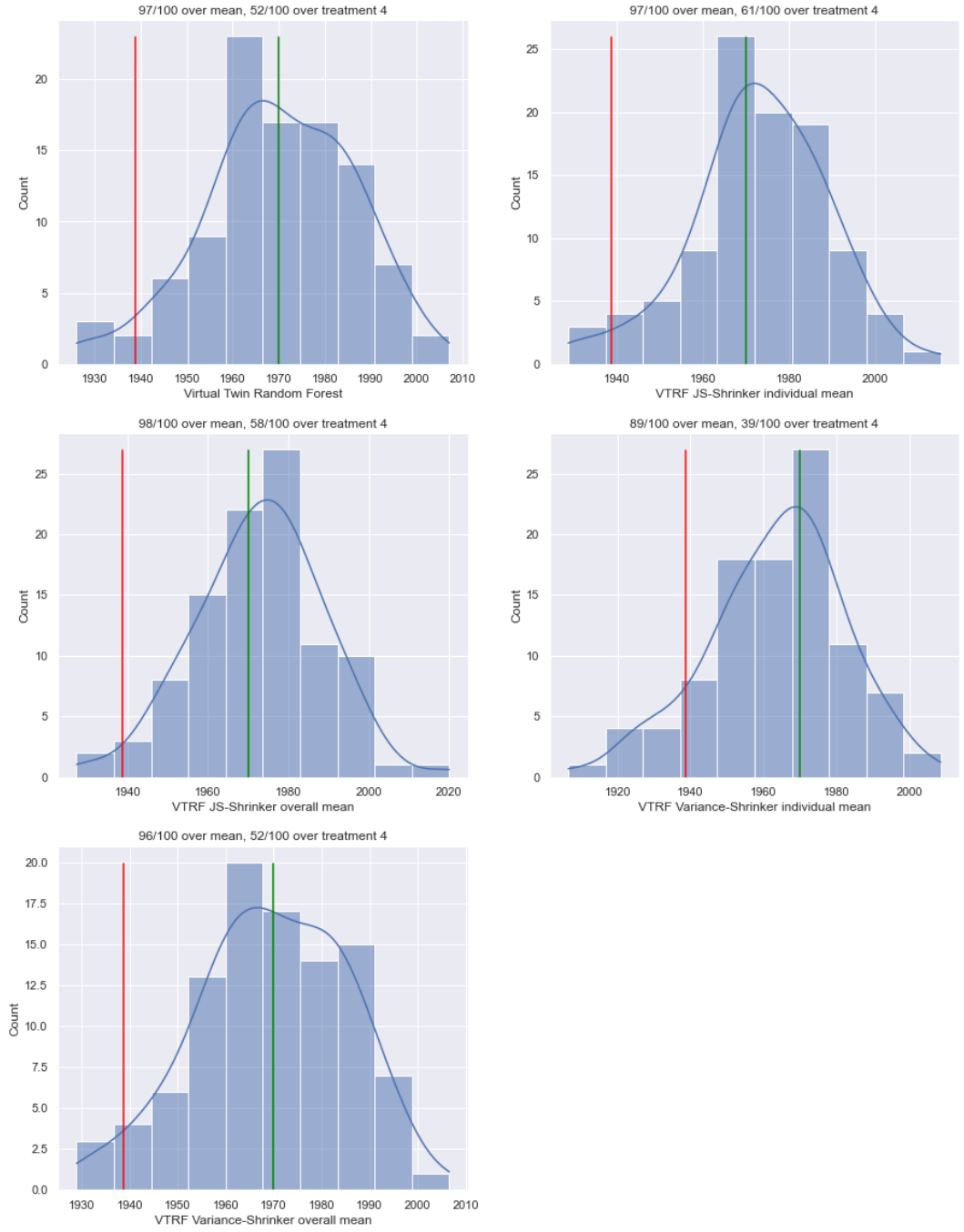

Figure 10: *Misra Matching VTRF: Subset of Treatments & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the virtual twin random forest method and using for all four introduced shrinkage methods. Only treatments 1, 2, 4, and 5 were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the four considered treatments, 1939 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).

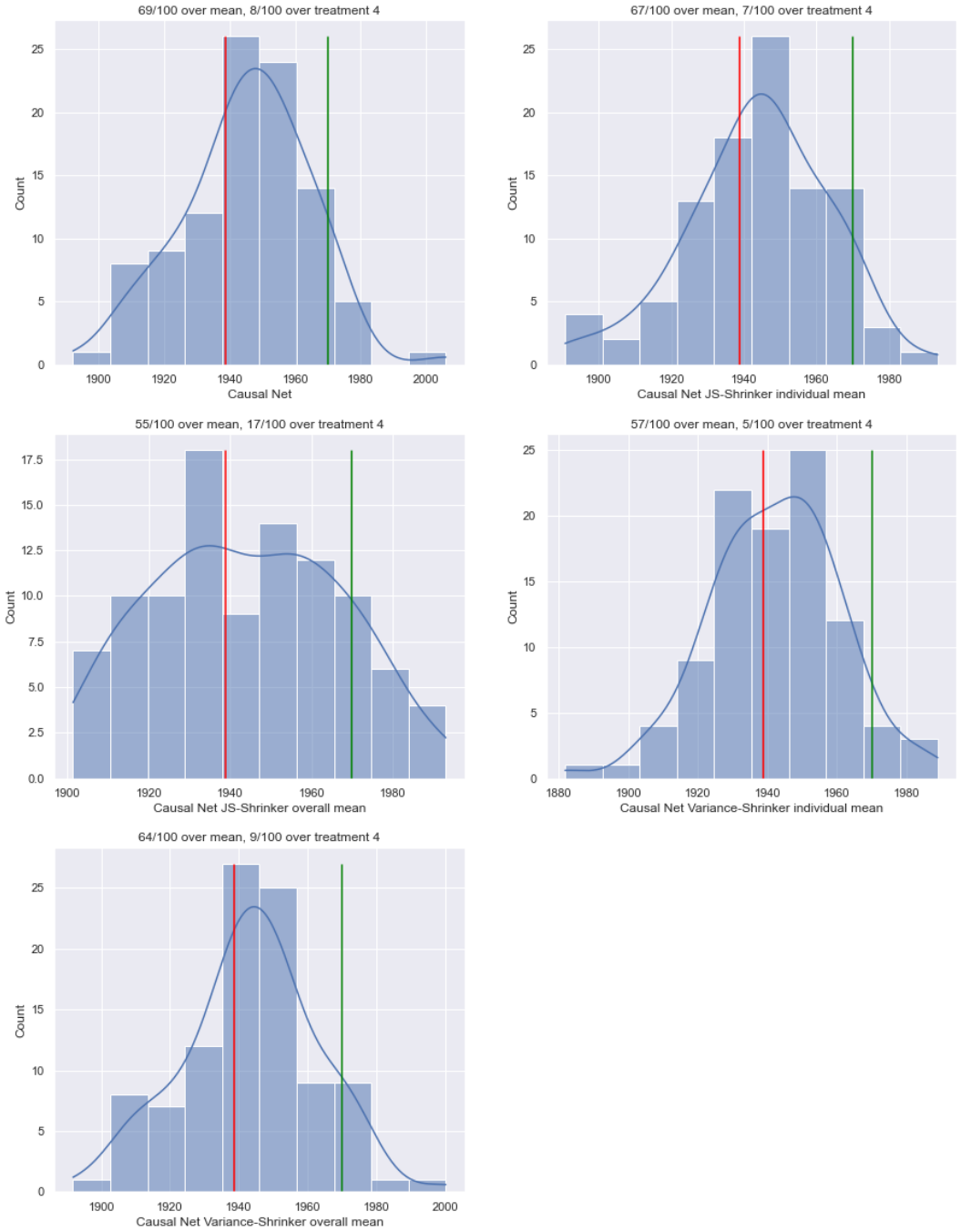

Figure 11: *Misra Matching Causal Net: Subset of Treatments & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the causal net method and using all four introduced shrinkage methods. Only treatments 1, 2, 4, and 5 were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the four considered treatments, 1939 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).

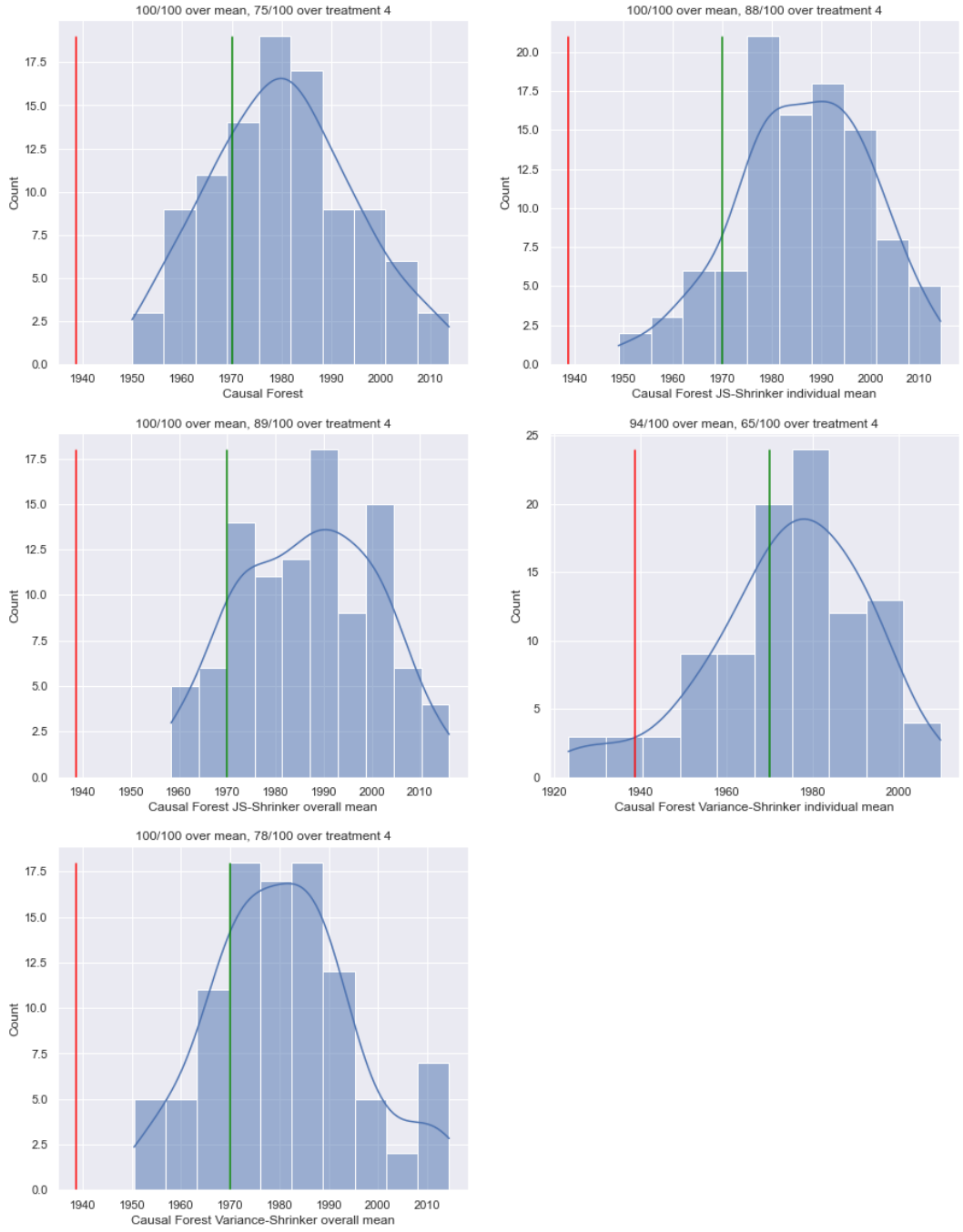

Figure 12: *Misra Matching Causal Forest: Subset of Treatments & Shrunken:* This figure depicts the distribution of the average outcome of matched observations of the individual 100 repetitions of the three-fold cross-validation only for the causal forest method and using all four introduced shrinkage methods. Only treatments 1, 2, 4, and 5 were considered. The green line depicts the average outcome of participants treated with treatment four (loss treatment), 1970 points, and the red line depicts the average outcome of participants treated with any of the four considered treatments, 1898 points. The addition "individual mean" refers to the shrinker shrinking towards the mean predicted treatment effect of the respective treatment. The addition "overall mean" refers to the shrinker shrinking towards the average treatment effect of being treated by any used treatment (in the training set).