Discussion Papers in Economics

No. 16/2017

Machine Learning to Improve Experimental Design

Tobias Aufenanger
University of Erlangen-Nürnberg

ISSN 1867-6707

Friedrich-Alexander-Universität Erlangen-Nürnberg
Institute for Economics
https://www.iwf.rw.fau.de/research/iwf-discussion-paper-series/

Machine Learning to Improve Experimental Design

Tobias Aufenanger *

Friedrich-Alexander University Erlangen-Nürnberg (FAU)

This version: August 2017

Abstract

This paper proposes a way of using observational pretest data for the design of experiments. In particular, this paper suggests to train a random forest on the pretest data and to stratify the allocation of treatments to experimental units on the predicted dependent variables. This approach reduces much of the arbitrariness involved in defining strata directly on the basis of covariates. A simulation on 300 random samples drawn from six data sets shows that this algorithm is extremely effective in increasing power compared to random allocation and to traditional ways of stratification. In more than 80% of all samples the estimated variance of the treatment estimator is lower and the estimated power is higher than for standard designs such as complete randomization, conventional stratification or Mahalanobis matching.

JEL Classification:
Keywords: experiment design, treatment allocation

*Friedrich-Alexander University Erlangen-Nürnberg (FAU), School of Business and Economics, PO Box 3931, 90020 Nürnberg, Germany, e-mail: tobias.aufenanger@fau.de

1 Introduction

In recent years the precision of data-driven prediction increased tremendously due to the rise of modern-day machine-learning algorithms. This paper proposes to use predictions derived from observational pretest data, to allocate experimental units into treatment and control group. I assume a situation in which the researcher has access to an observational data set including the untreated dependent variable as well as several covariates.[1] In addition, the researcher seeks to run an experiment to estimate the causal relationship between the treatment and the dependent variable. For this situation, I suggest to train a machine learning algorithm, in particular a random forest, on the observational data to build a predictive model. After sampling the experimental units, I suggest to predict the untreated dependent variable of all experimental units from their covariates and to stratify based on these predictions.

Stratified randomization or blocking is one of the most frequently applied methods for systematic treatment allocation in economics as well as in other fields of research (Kernan et al., 1999; Bruhn and McKenzie, 2009; Berry, 2011). A huge argument in favor of stratification is that it ensures balanced groups while still including a large degree of randomness. This randomness allows for randomization-based inference, which requires hardly any assumptions apart from a random allocation of treatments (Imbens and Rubin, 2015). Athey and Imbens (2017) show that the variance of the treatment estimator under stratification is always lower than or equal to the variance under complete randomization. Apart from some sampling assumptions, this theorem does not require any assumptions on the relation between the covariates and the dependent variable.

Although the idea of stratification and the randomization inference thereof has a long history (Fisher, 1926; Wilk, 1955; Kempthorne, 1955), there still exists little advice on how to select strata in practice. Researchers agree that strata should be built based on those covariates that most strongly influence the dependent variable (Bruhn and McKenzie, 2009; Moore, 2012). However, classical stratification, i.e., building strata directly out of covariates, quickly results in too many strata. For example, stratifying on income (discretized into 8 categories), gender (2 categories), race (5 categories), and age (8 categories), already results in $8 \cdot 2 \cdot 5 \cdot 8 = 640$ different strata, which will be far too many for a small experiment of 30-100 individuals. High-dimensional stratification is usually a non-bipartite matching approach based on a multivariate measure of distance between the covariates of the experimental units (Greevy et al., 2004; Moore, 2012). Typical distance measures are Mahalanobis or Euclidean distances. High-dimensional stratification allows to simultaneously account for arbitrarily many covariates. Yet, since the distance measures do not assign weights on the covariates, depending on the size of the effect on the dependent variable, this approach nevertheless requires the researcher to select the most important covariates. For example, once one uses this approach on one covariate that highly affects the dependent variable, and ten covariates that do not or only weakly affect the de-

[1] Possible extensions to other types of pretest data are discussed in Section 4.

pendent variable, the resulting treatment allocation will be much worse than if one only stratifies on the one important variable. This paper uses random forests as well as observational pre-test data to transform the covariates into an index that has the highest possible correlation with the dependent variable. In the same example with one highly important variable and ten unimportant variables, the random forest will identify the important variable and the index will be based almost exclusively on this variable.

Using pre-test data in the design of experiments can foster efficient estimation of treatment effects. This insight comes from a broad literature on adaptive experimentation (see Robbins (1952) and the references therein for some initial contributions to this literature and Berry (2011) for some benefits of adaptive experimentation). For a sequence of experiments, these approaches use data from the previous experiments to determine the allocation of treatments in the next experiment.[2] One major focus of adaptive experimentation is to balance two different goals: First, the experimental design should for each experimental unit identify the best out of a set of multiple treatments depending on the covariates of the experimental unit. Second, the design should treat the experimental units as effectively as possible, i.e., for each unit apply the treatment with the highest expected personalized treatment effect. For example, in medical research researchers often aim at applying treatments primarily to those subjects for whom the risk of side effects is low, while still being able to efficiently estimate personalized treatment effects (Sverdlov, 2015; Villar et al., 2015). In online advertisements, companies seek to use the ad that works best as soon as possible, while still being able to efficiently estimate which out of a set of potential ads works best (Bubeck and Cesa-Bianchi, 2012; Tang et al., 2013). A mathematical formulation of this problem goes under the name of the *multi-armed bandit problem* (Thompson, 1933; Mahajan and Teneketzis, 2008) or *contextual multi-armed bandit problem* (Wang et al., 2005; Lu et al., 2010).[3]

While this paper is closely related to adaptive experimentation in the sense that it uses pre-collected data in the design of experiments, it targets a different case. First, whereas adaptive experimentation usually regards (arbitrarily long) sequences of experiments (Bubeck and Cesa-Bianchi, 2012), this paper regards only two stages. The first stage is the collection of observational data; the second stage is the conduction of the experiment using the observational pre-test data. Second, adaptive experimentation usually aims at assigning different propensity scores, i.e., different probabilities of receiving the treatment, to different experimental units (Hahn et al., 2011); the approach of this paper leaves the propensity score constant at 0.5 for each unit. Lastly, in contrast to adaptive experimentation, the approach of this paper does not require experimental pre-test data. Rather, I require an observational data set that contains only the untreated dependent variable (the status quo) as well as the covariates.

[2]Experiments in this case mean a sample of one or more experimental units, in which the experimenter can freely allocate treatments. Before allocating the treatments, the researcher observes the covariates of the experimental units, after applying the treatment the researcher observes the dependent variable of all units.

[3]Depending on whether there is covariate information (context) available or not.

The major advantage of the data-driven approach of this paper is that it reduces much of the arbitrariness governing classical stratification. Using this approach, the researcher can simply take all available covariates into account without repeatedly thinking about which of the covariates are most important. In simulations on 300 random samples drawn from six different data sets, I show that this approach very effectively leads to more precise estimates and to a higher power of the experiment compared to complete randomization, classical stratification, and Euclidean or Mahalanobis matching. On average, this approach cuts the variance of the treatment estimator in half compared to complete randomization. In terms of sample sizes, this means that the approach of this paper requires on average around half the sample size to estimate the treatment effect with the same variance as complete randomization. In addition the approach increases statistical power on average by 25 - 30%. On the SEED data, a data set containing around $1,500$ covariates (see Ashraf et al., 2006), power even increases from around 75% for complete randomization, classical stratification, and matching, to 98%. In this case, the risk of a type II error (i.e., failing to estimate a significant effect, even though the true treatment effect is positive) consequently reduces from one out of four to one out of 50.

This paper is structured as follows: Section 2 presents the stratification approach as well as a theoretical motivation. Section 3 simulates the approach on six data sets. Section 4 discusses possible extensions of the approach. Section 5 concludes.

2 Experimental Design

2.1 Definitions

In this subsection, I define the term *experimental design* as well as the assumptions on the population and the experimental units.

As a model for inference, I use the potential outcomes model (e.g Rubin, 1974; Imbens and Rubin, 2015, Chapter 1). I assume a large population of N_p units. Those units could be people, schools, municipalities, or any other unit of interest. Each unit is characterized by a vector $U_i = (Y_i(0), Y_i(1), X_i)$, with $i = 1, ..., N_p$. $Y_i(0)$ is the potential outcome, if unit i does not receive the treatment, $Y_i(1)$ is the potential outcome, if unit i receives the treatment. $X_i = (X_{i1}, ..., X_{iz})$ is a vector of covariates. Lastly, I denote the sample distribution of U by $\mathbb{P}$.

For unit i, $Y_i(1) - Y_i(0)$ is the causal effect of the treatment. The goal of this paper is to estimate average treatment effects $\tau = \mathbb{E}[Y(1) - Y(0)]$, taking the expectation over $\mathbb{P}$. For estimating the average treatment effect, the researcher draws a sample of size N_s out of the population. For each unit in this sample, the researcher observes a vector $U^{obs} = (Y_i(T_i), X_i, T_i)$, where $T_i \in \{0, 1\}$ is a dummy variables, indicating whether unit i received the treatment ($T_i = 1$), or not ($T_i = 0$). From the definition of U^{obs}, we see what Holland (1986) calls the *fundamental problem of causal inference*: Each unit can only be observed in one state (either treated or untreated) and therefore the causal effect of the treatment cannot be observed for any unit.

In order to estimate the average treatment effect, the researcher thus has to rely on an *experimental design*. The experimental design is the sum of all decisions the researcher has to make while designing an experiment. In practice, many decisions certainly concern the definition and measurement of the dependent variable as well as the design of the treatment. In this paper, I take those things as given and define an experimental design as follows:

Definition 2.1. *(Experimental Design)*
For a given population of units, an experimental design composes of:

1. *A strategy for sampling experimental units from the population.*
2. *A strategy for allocating treatments to experimental units.*
3. *A strategy for analyzing the experimental data.*

2.2 Motivation

This subsection motivates the stratification approach of this paper. I assume, that the researcher knows the potential outcomes of the entire population. Given this knowledge, I analyze how strata should be defined optimally to minimize the variance of the treatment estimator. Of course this is not a realistic scenario. If the researcher knows the potential outcomes of every unit in the population, there is no reason for conducting an experiment anymore. Rather, this subsection should show how ideal strata should look like.

Consider the following experimental design (stratification):

1. Divide the population into m strata of equal size.[4] Sample n experimental units out of each sample, resulting in a sample size of $N_s = n \cdot m$.
2. Allocate treatments randomly under the condition that exactly half of the units in each stratum are allocated to the treatment group and the other half to the control group.
3. Analyze the data via randomization inference on the difference in means of the dependent variable between treatment and control group:

$$\hat{\tau} = \bar{Y}_t - \bar{Y}_c = \frac{2}{N_s} \sum_{i=1}^{N_s} T_i Y_i(1) - \frac{2}{N_s} \sum_{i=1}^{N_s} (1 - T_i) Y_i(0)$$

Under the assumption that N_p is sufficiently large, and thus the experimental units $U_i, i = 1, ..., N_s$ are approximately independent, the treatment estimator for this design is given by:

$$\mathbb{V}[\hat{\tau}]_{st} = \frac{2}{mN_s} \sum_{j=1}^{m} (\sigma_{tj}^2 + \sigma_{cj}^2) = \mathbb{V}[\hat{\tau}]_{cs} - \frac{2}{mN_s} \sum_{j=1}^{m} (\mu_{tj} - \mu_t)^2 + (\mu_{cj} - \mu_c)^2 \quad (1)$$

[4] As a technical assumption, I require that N_p is divisible by m.

Here I used the following notations: $\mathbb{V}[\hat{\tau}]_{st}$ is the variance of the difference in means estimator under the stratification design and $\mathbb{V}[\hat{\tau}]_{cs}$ under complete randomization. As complete randomization, I denote the special case of stratification with only one stratum. Lastly, μ_t and μ_c denotes the mean over $\mathbb{P}$ of the treated and untreated potential outcomes. Similarly, $\mu_{tj}, \mu_{cj}, \sigma_{tj}$ and σ_{cj} are mean and variance inside stratum j.[5]

Equation 1 shows that the variance of the treatment estimator is always lower or equal under stratification than under complete randomization. The variance under the two designs is only equal, if the variables to stratify on do not affect the (treated and untreated) dependent variable at all (i.e., only if $\mu_{tj} = \mu_t$ and $\mu_{cj} = \mu_c$ for all $t = 1, ..., m$). This consideration leads Athey and Imbens (2017) to recommend that one should always stratify. Note that whereas the assumption of random sampling is necessary to show that stratification always results in a lower variance of the estimator than complete randomization, this assumption is not necessary for inference from stratification. Stratified experiments can be analyzed via randomization inference which assumes the experimental sample as given and only exploits the randomness of the treatment allocation for inference (e.g. Imbens and Rubin, 2015). In this motivation, I will however stick to the random sampling assumption.

So how should strata in this situation ideally be selected? Let S_j contain all units of stratum j in the population ($j = 1, ..., N_p$). Then to minimize the variance $\mathbb{V}[\hat{\tau}]_{strat}$, the strata should satisfy the following condition:

$$(S_1, ..., S_m) = \underset{S_1,...,S_m}{\operatorname{argmin}} \frac{N_p}{m} \sum_{j=1}^{m} (\sigma_{tj}^2 + \sigma_{cj}^2), \quad s.t. \, |S_1| = ... = |S_m| = \frac{N_p}{m} \tag{2}$$

Reformulation of the right hand side yields:

$$(S_1, ..., S_m) = \underset{S_1,...,S_m}{\operatorname{argmin}} \sum_{j=1}^{m} \sum_{U_i \in S_j} \|Y_i - \mu_i\|, \quad s.t. \, |S_1| = ... = |S_m| = \frac{N_p}{m}, \tag{3}$$

with $Y_i = (Y_i(1), Y_i(0))'$ and $\mu_j = (\mu_{tj}, \mu_{cj})'$. This is a k-means problem with the condition, that all strata should have the same size.

Next, let us suppose that the researcher only observes the potential untreated outcomes in the population, but not the treated outcomes. In this case, the selection of strata cannot minimize the sum of variances of the treated outcomes ($\sum_{j=1}^{m} \sigma_{tj}^2$). Intuitively, in this case one would simply minimize the sum of variances of the untreated outcomes ($\sum_{j=1}^{m} \sigma_{cj}^2$) and not care about the treated outcomes. I will show that this intuition is correct. Calculation reveals:

[5] Precisely:

$$\sigma_{tj}^2 := \mathbb{V}[Y(1)|U \in S_j], \sigma_{cj}^2 := \mathbb{V}[Y(0)|U \in S_j]$$
$$\mu_{tj} := \mathbb{E}[Y(1)|U \in S_j], \mu_{cj} := \mathbb{E}[Y(0)|U \in S_j],$$

where S_j contains all units of stratum j.

$$\mathbb{V}[\hat{\tau}]_{st} = \frac{2}{mN_s}\sum_{j=1}^{m}\sigma_{tj}^2 + \sigma_{cj}^2 = V[\hat{\tau}]_{cs} - \frac{2}{mN_s}\sum_{j=1}^{m}(\mu_{tj}-\mu_t)^2 + (\mu_{cj}-\mu_c)^2 \quad (4)$$

$$\leq \mathbb{V}[\hat{\tau}]_{rand} - \frac{2}{mN_s}\sum_{j=1}^{m}(\mu_{cj}-\mu_c)^2 = k + \frac{2}{mN_s}\sum_{j=1}^{m}\sigma_{cj}^2 \quad (5)$$

for $k = \frac{2}{mN_s}\sum_{j=1}^{m}(\sigma_{tj}^2 + (\mu_{tj}-\mu_t)^2)$. This is a sharp upper bound in the sense that if strata are built on basis of the untreated outcomes, $\mathbb{V}[\hat{\tau}]_{st}$ will be equal to this bound if treated and untreated outcomes are independent, and lower if there is some kind of dependence (not necessarily linear dependence). Consequently, minimizing $\sum_{j=1}^{m}\sigma_{cj}^2$ always reduces $\mathbb{V}[\hat{\tau}]_{st}$ compared to $\mathbb{V}[\hat{\tau}]_{cs}$ and the reduction is higher, the higher the dependence of the treated and untreated outcomes is. Finally, what does minimizing $\mathbb{V}[\hat{\tau}]_{cs}$ mean? Reformulating the minimization problem into a k-means problem yields:

$$\min_{S_1,\ldots,S_m}\sum_{j=1}^{m}\sum_{U_i\in S_j}\|Y_i(0)-\mu_{cj}\|, \quad s.t.\,|S_1| = \ldots = |S_m| = \frac{N_p}{m} \quad (6)$$

This problem has a simple solution: Rank all units according to their untreated outcome. W.l.o.g., let $Y_1(0) \leq Y_2(0) \leq \ldots \leq Y_{N_p}(0)$. Now put the first $\frac{N_p}{m}$ units into stratum S_1, the second $\frac{N_p}{m}$ units in stratum S_2, and so on.

2.3 Stratification on predicted outcomes

Certainly, the cases in the last section are not very realistic. In this section, I propose an experimental design for a more realistic case. In particular, I assume that the researcher observes the untreated outcomes as well as the covariates only for a randomly drawn sample (observational data set) of the population. The experimental sample is then drawn out of the same population.

This setting involves all experiments that test a treatment on a population that previously did not receive the treatment. Examples are the effect of an education program on future income of people that previously did not receive the program; the effect of a certain incentive scheme on working performance among workers that previously faced another incentive scheme; or the effect of a certain diet on future health of consumers that did not try this diet before. Section 4 discusses possible extensions to other settings.

For this case, I suggest the following experimental design which I call *machine learning stratification* (MLS):

1. Randomly sample $N_s = n \cdot m$ experimental units out of the population, where $n, m \in \mathbb{N}$.

2. - Use the observational data set to derive a predictive model $\hat{Y}(0) = f(X)$ that predicts the untreated potential outcome from the vector of covariates.

- Rank the experimental units with respect to the predicted untreated outcomes, i.e., such that $f(X_1) \leq f(X_2) \leq ... \leq f(X_{N_s})$.
- Define m strata such that the first n units are put in stratum one, the second n units are put in stratum two, and so on.
- Allocate treatments randomly under the condition that exactly half of the units in each stratum are allocated to the treatment group and the other half to the control group.

3. Analyze the data via randomization inference on the difference in means of the dependent variable between treatment and control group.[6]

In principle $f(X)$ could be a linear model $Y(0) = \beta_0 + X'\beta_x + \varepsilon$ that is fitted on the observational data set. However, this turns the initial problem of stratification into a fairly similar problem. Which covariates should be included in the linear model? Are there nonlinear effects or interaction effects? All of this has to be specified by the researcher and will in the end be kind of arbitrary. This is where machine learning algorithms come into play. These algorithms search for good predictive models in a structured way.

In this paper, I use a random forest for building the predictive model. Random forests, as introduced by Breiman (2001), have proven to be very effective on prediction problems. In a comparison of 179 methods on 121 data sets, Fernández-Delgado et al. (2014) find that this machine learning algorithm is best suited for prediction. Féraud et al. (2016) use the random forest for compressing the covariate information in contextual bandit problems with great success. The random forest is a non-parametric machine learning algorithm, meaning that it catches all kinds of nonlinear and interaction effects in the data without requiring the researcher to specify a particular model (Wager and Walther, 2016). Since this algorithm is very robust to noise (Wyner et al., 2017), I suggest to use all available covariates for building the predictive model.

The idea of the random forest is to fit several decision trees on random subsets of the observations and the covariates, and average the predictions of the trees. Decision trees work as follows:[7] Consider a case of a single, ordinal covariate X. Then in a first step, the tree will determine a cutoff value $\bar{X}$ and cuts the sample in two leafs. One contains all units with $X < \bar{X}$, the other contains all units with $X \geq \bar{X}$. The prediction of each leaf is the average dependent variable in the leaf. The selection of $\bar{X}$ minimizes the mean square error in the sample (i.e., the average squared differences between the predicted dependent variables and the true variables). In the next step, each leaf is split again, and so on, until the leafs reach a predefined minimum size. In case of multiple ordinal covariates, each spit of the tree selects the one covariate and the one cutoff level to minimize the mean squared error. Categorical covariates are included through dummy variables. One major issue with those trees is overfitting. Whenever leafs are too small, the out-of-sample predictive power of the tree will be low. Take for example a tree in

[6] For an introduction to randomization inference on stratified experiments see Athey and Imbens (2017).

[7] See Berk (2008) for a textbook discussion of trees and forests from a regression perspective.

which all leafs have size one. In this case the in-sample mean squared error will be equal to zero, but out-of-sample prediction is impossible. For this reason the minimum size of the leafs is typically quite large. By taking the average over many decision trees on random subsets of the sample and a random selection of the covariates, the random forest reduces the overfitting problem and allows for a smaller leaf size. The main principle is that leaf predictions based on noise will occur at random and will thus asymptotically cancel out by the law of large numbers.

In combination with the random forest, the MLS design transfers many decisions from the researcher to the observational pretest data. Typically, an important task in classical stratification is to find the right covariates to stratify on. Even though some researchers might take pretest data or previous literature into account for selecting these covariates, the final decision is still more or less arbitrary. The approach of this paper presents a more structured way of selecting these covariates. It uses the random forest and the available data to create a new covariate that has a high correlation with the (untreated) dependent variable, and stratifies on this covariate. The only decision that the researcher has to take is the size of the strata n. In the simulations of this paper, I follow Athey and Imbens (2017), who recommend a strata size of four.[8]

Another benefit of this approach is that once the researcher decided for a strata size of n, the approach ensures that all (or at least all but one) strata actually are of size n. With classical stratification this is hardly possible. Suppose one seeks to stratify on gender and race. Then random sampling can easily result in three white females and five white males. Stratified sampling requires knowledge of the covariates in the entire population and even then it is possible that some subjects refuse to show up for the experiment.

3 Simulations

In this section, I compare the performance of the MLS design to complete randomization, classical stratification and multivariate matching. I measure performance with respect to the variance of the treatment estimator as well as the power, i.e., the probability of finding a significant treatment effect. For testing significance of an estimator, I use Fisher's exact test (Fisher, 1926, Chapter 20.02). For an application to experiments involving continuous dependent variables, see Athey and Imbens (2017). Exploiting the randomness of the treatment allocation, this tests assesses the null hypothesis that the average treatment effect in the given sample is zero.

[8]The authors suggest a sample size of four, even though strata of size two will typically yield lower variances of the treatment estimator. The reason for this suggestion is that standard errors of the estimator can be more effectively estimated for a strata size of four or more.

Simulation Procedure

For simulation, I use the software R (R Development Core Team, 2008). The MLS design requires the randomForest package developed by Breiman (2001). Multivariate matching makes use of the nbpMatching package implemented by Lu et al. (2011). The latter matches on Mahalanobis distance whenever the variance-covariance matrix of the covariates can be estimated from the data (i.e., whenever the number of covariates is low compared to the sample size), and on Euclidean distance otherwise.

To simulate the performance of the experimental designs on real data, I use a similar approach to Bruhn and McKenzie (2009) (see also Schneider and Schlather (2017) for simulations using the same approach on the same data). This approach requires a data set that contains an (untreated) dependent variable $Y(0)$ as well as several covariates X. Treatments are simulated by adding a constant to the dependent variable $Y(1) = c + Y(0)$.[9]

Data

Bruhn and McKenzie (2009) use this simulation approach to compare the performance of several experimental designs, among others complete randomization, classical stratification and matching, on five different data sets. These data sets are subsets of 430 units from larger data sets, in particular: The Mexican employment survey (ENE), the Indonesian Family Live Survey (IFLS), data on microenterprises in Sri Lanka and child and household data from the Learning and Educational Achievement project (LEAPS) project in Pakistan. From the last data set, the authors extract two dependent variables as well as two different sets of covariates and split it into two data sets. For all datasets, they define one subset of the covariates as observables and one as unobservables. Further, out of the set of observables, they select a set of four variables for stratification. Stratifying on the first two yields 8 strata, the first three yield 24 strata, and all four yield 48 strata. For a more detailed description of the data, see Bruhn and McKenzie (2009) as well as the supplementary material thereof.

In this paper, I use the same five data sets as Bruhn and McKenzie (2009). Classical stratification in this paper uses the same strata as the original contribution. I present results for experimental sample sizes of 32, and thus only include 8 strata. Appendix A regards sample sizes of 96 and uses 8, 24 as well as 48 strata. Matching and MLS take all covariates into account, i.e., those originally labeled as observables as well as those labeled as unobservables. Given the data sets of 430 experimental units I delete observations with missing[10] and split the data into an experimental data set of 32 units and an observational data set containing the remaining units. The observational data set is the training set for the random forest. On the experimental data set, I allocate treatments for $1,000$ times for each experimental design and calculate the variance of the difference in

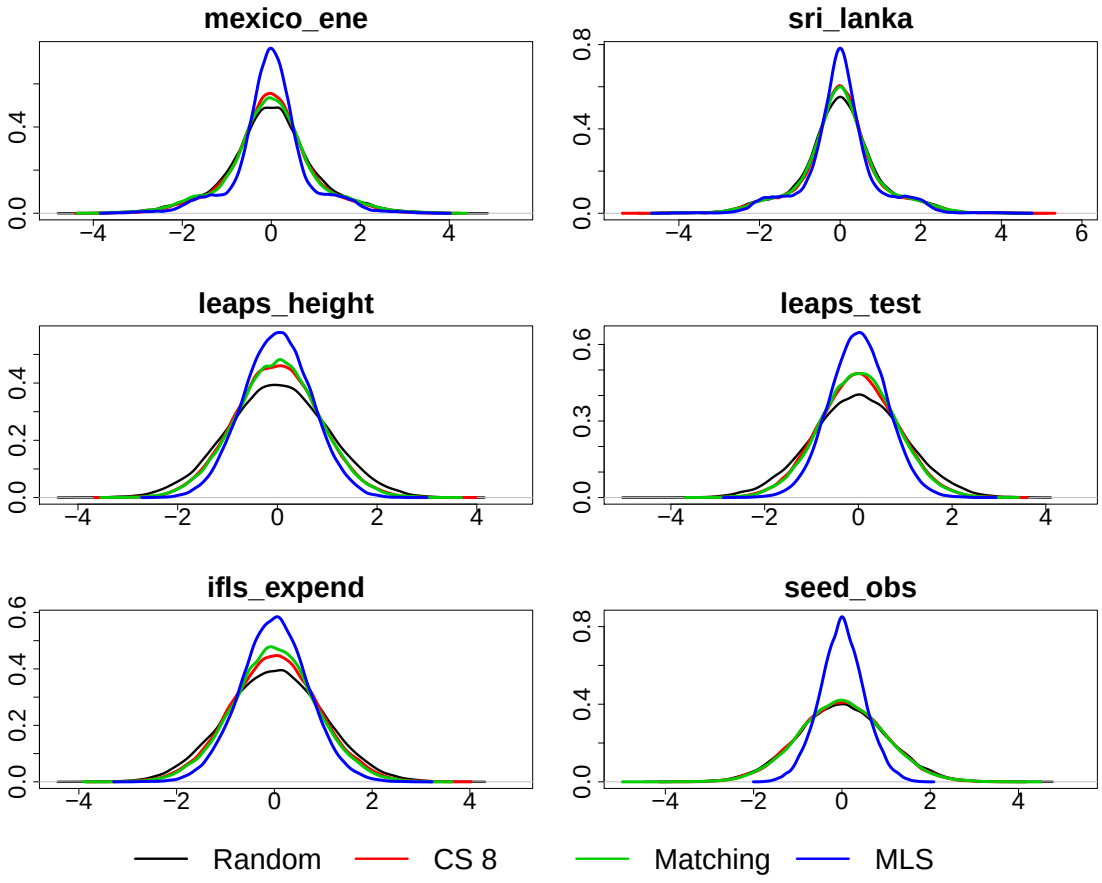
[9] As a robustness check, I include another simulation approach in Appendix B that attempts to take over the treatment effect heterogeneity from an experimental data set.

[10] Out of the five data sets, only the sri_lanka data contained missing values.

means estimator as well as the power of the Fisher exact test. To ensure that the results are not driven by a favorable or unfavorable split, I repeat this procedure (splitting the data and allocating treatments) for 50 times. This makes $50,000$ Monte Carlo steps for each experimental design on each of the five data sets.

In addition to those five data sets, I include an additional one. It contains data on a novel commitment savings product called SEED (Save, Earn, Enjoy Deposits) offered by a Philippine bank (Ashraf et al., 2006). Apart from the treatment (the offering of the commitment savings product) and the dependent variable (the change in the overall balance for a given time frame), this experiment reports around 1,800 covariates.[11] Most of the covariates have only little influence on the dependent variable. For this simulation, I keep only the control group. After removing variables with many missings as well as outliers, this results in around 700 units and $1,500$ covariates. Similar to Bruhn and McKenzie (2009), I select four variables for classical stratification: The saving balance before the experiment, gender and age of the subject and a dummy indicating whether the subject exhibits hyperbolic preferences.

Results



[11]The covariates comprise of responses to surveys conducted by the researchers and administrative bank data.

Figure 1: Distribution of difference in means estimator

Figure 1 shows the distribution of the difference in means estimator for a treatment effect of $c = 0$. Since the true treatment effect is zero, distributions with higher mass around zero have a higher probability of precisely estimating the true effect. In line with Bruhn and McKenzie (2009) and Schneider and Schlather (2017), the plots show that classical stratification (CS) and matching effectively increase the precision of the estimation compared to complete randomization. Only on the SEED data there is little increase in precision. On all data sets, the MLS design outperforms all other designs. Especially on the SEED data MLS very successfully increases the precision of the estimator. This shows the power of this approach compared to conventional designs. The SEED data contains almost 1,500 covariates, all of which have only little influence on the dependent variable. The random forest very effectively extracts the relevant information out of these covariates. By stratifying on the random forest predictions one can effectively take information on all covariates into account.[12] Matching also takes all covariates into account. However, since Euclidean distance does not carry any information of the importance of covariates for the dependent variable, matching performs poorly compared to MLS.

Table 1 calculates the variance of the difference in means estimator as well as the power of the Fisher test. As 'power' I denote the probability that the p-value of the test is lower than 0.05. For this table, I apply a positive treatment effect. I calibrate the effect size in a way that the power for complete randomization is around 0.7. The column 'Variance of Estimator' presents the variance of the difference in means estimator for each design in terms of the variance for complete randomization. On average, the MLS design cuts the variance for complete randomization in half. It is common knowledge that twice the sample size leads to half the variance of the treatment estimator in case of complete randomization. Thus, the MLS design on average requires around half the sample size as complete randomization to estimate the treatment effect with the same precision. In addition, the power of the MLS algorithm is higher than for complete randomization, classical stratification, or matching on every single data set. The benefits of MLS compared to the other designs are largest on the high-dimensional SEED data. With the given treatment effect size, a researcher using classical designs would obtain an insignificant result in one out of four experiments. Using the MLS design this would happen only in one out of 50 times.

Finally Table 2 presents a summary over all splits of the data. As mentioned in the beginning of this section, I split each data set into experimental and observational sample for 50 times. For all data sets this makes a total of 300 samples. On each of the samples, I allocate treatments 1,000 times for each design. Using these 1,000 observations, I calculate the variance of the difference in means estimator as well as the power for each sample. Table 2 reports for each design the number of samples in which the particular design performed best. The MLS design lead to the lowest variance of the estimator and to the highest power in

[12] Note that the training data set of 658 units is quite small for $1,500$ covariates. For a larger training data set, the MLS design will probably perform even better.

Table 1: Performance of experimental design

	Variance of Estimator	Power	Number of Covariates	Training dataset size
mexico_ene:				
Random	1.00	0.73	30	396
CS.8	0.83	0.79	30	396
Matching	0.94	0.75	30	396
MLS	0.65	0.86	30	396
sri_lanka:				
Random	1.00	0.73	34	363
CS.8	0.95	0.76	34	363
Matching	0.95	0.74	34	363
MLS	0.78	0.83	34	363
leaps_height:				
Random	1.00	0.69	7	398
CS.8	0.71	0.81	7	398
Matching	0.70	0.80	7	398
MLS	0.46	0.93	7	398
leaps_test:				
Random	1.00	0.73	7	398
CS.8	0.69	0.84	7	398
Matching	0.67	0.85	7	398
MLS	0.39	0.96	7	398
ifls_expend:				
Random	1.00	0.70	7	398
CS.8	0.80	0.78	7	398
Matching	0.74	0.79	7	398
MLS	0.49	0.91	7	398
seed_obs:				
Random	1.00	0.72	1476	658
CS.8	0.93	0.75	1476	658
Matching	0.93	0.74	1476	658
MLS	0.26	0.98	1476	658

Number of splits: 50, MC steps per split: 1000, Size of experimental dataset: 32

more than 80% of all samples. In addition, complete randomization was almost never the best design. This supports the reccommendation of Athey and Imbens (2017) to always stratify.

Larger samples

For a larger experimental data sets of 96 units, appendix A finds similar results to this simulation. Especially the variance of the estimator in terms of the variance for complete randomization remains equal, or becomes slightly lower. This is surprising, since results from a linear model inference framework propose that the

Table 2: Share of splits in which the algorithm performs best

	random	strata8	matching	strataForest
Variance of Estimator	0.00	0.03	0.13	0.84
Power	0.02	0.05	0.16	0.86

Number of splits: 300, MC steps per split: 1000, Size of experimental dataset: 32

benefits of alternative designs compared to complete randomization should decrease as the sample size increases. In particular, Aufenanger (2017) finds that in a linear model framework, the variance of the treatment estimator for any systematic allocation of treatments in terms of the variance for complete randomization converges to one as the sample size increases and the number of covariates stays constant. Also Bruhn and McKenzie (2009) report: "Our simulations suggest that in samples of 300 or more, the different methods [i.e., designs] perform similarly.". The simulations of this paper suggest that in a randomization inference framework with difference in means estimation, the benefits of alternative designs compared to complete randomization do *not* decrease with the sample size of the experimental data set. As Appendix A shows, theory supports this suggestion.

4 Possible extensions

This paper provides a first approach for using observational pretest data in a structured way for the design of experiments. This section discusses several possible extensions to this approach.

For example, the random forest yields a proximity measure between covariate vectors, indicating in how many of the forests' trees the two covariate vectors would have been in the same leaf. This measure includes both information of the difference in predicted dependent variables (the prediction of a tree is equal for all units in the same leaf) as well as the direct difference in covariates. Take for expample two units that have been in different leafs in every tree of the forest. If the predicted dependent variable for these two units is the same, the MLS approach would regard those two units as identical. On the contrary, for the random forest proximity measure, those two units are entirely different. If one believes that the probability for the two true outcomes to be close is higher if the covariates of the two units are close, it might be a good idea to use multivariate matching on the forest proximity measure. However, if one doesn't believe that this probability is higher if covariates are closer and predictions remain the same, matching on the proximity measure could also diminish the performance compared to MLS, since the two units with identical predicted outcomes and different covariates will not be matched.

Further extensions of the MLS design include applications to alternative types of pretest data. In the case of experimental pretest data, one possibility would be to train the forest only on the control group (or only on the treatment group). However, this would neglect important information. Another possibility would be

to train two forests: One on the control group to predict the untreated outcomes, one on the treatment group to predict the treated outcomes. The motivation of Section 2.2 suggests to select strata according to the following k-means problem:

$$(S_1, ..., S_m) = \underset{S_1,...,S_m}{argmin} \sum_{j=1}^{m} \sum_{U_i \in S_j} \|\hat{Y}_i - \hat{\mu}_j\|, s.t. |S_1| = ... = |S_m| = n, \qquad (7)$$

Where $\hat{Y}_i = (\hat{Y}_i(0), \hat{Y}_i(1))$ is the vector of the predicted untreated and treated outcome and $\hat{\mu}_j = (\hat{\mu}_{cj}, \hat{\mu}_{tj})$ is the average predicted treated and untreated outcome in stratum S_j.

Finally, regard the case of observational pretest data that contains both treated and untreated outcomes. In case of observational data, subjects typically self-select into treatment and control. In this case, the propensity score, i.e., the probability that an experimental unit with a given vector of covariates receives the treatment usually has little overlap. This means there exists one group with specific values of the of covariates that receives the treatment and another group with other covariate values that receives no treatment. One way to treat this data is to train one random forest on all units that received the treatment and another on all units that received no treatment. Further, estimate the propensity score on the observational data set (for example, using the random forest). In the experimental data set, calculate the propensity scores of all units. For units with a propensity score of more than 0.5, the random forest will be better at predicting the treated potential outcome than the untreated outcome, since the training data for similar covariate values was larger. Similarly, for units with propensity scores lower than 0.5, the random forest will be more efficient at predicting the untreated outcome. To account for this, one could split the experimental sample in two strata, one with propensity scores above, the other with propensity scores below 0.5. In the first group, stratify on the predicted treated outcome, in the second on the predicted untreated outcome.

5 Conclusion

This paper shows a great potential for using observational pretest data in the design of experiments. Modern day machine learning algorithms, such as the random forest, make it possible to account for all available covariates in the design. The machine learning stratification approach introduced in this paper provides several benefits over classical experimental designs.

First, it provides a structured approach on how to define strata in practical applications. Up to now the decision on which covariates to stratify or match on was always up to the researcher. The MLS design makes it possible to transfer the task of selecting the most important variables to the machine learning algorithm and the pretest data. Second, in contrast to classical designs, the approach of this paper takes into account the importance of the covariates for the dependent variable. For example, suppose there are two covariates, one of which has a large effect on the dependent variable and the other one has a medium size effect. Then

for classical designs, such as classical stratification or matching, the researcher has to decide whether to take only the first covariate or both into account. In the latter case, the design treats both covariates equally and does not account for the fact that the first covariate is more important than the second. In the same example, the MLS design will account for both covariates *and* for the fact that the first one is more important than the second. Third, as the main goal of treatment allocation is to distribute potential outcomes equally across treatment and control group, the MLS approach directly targets this goal.

In the simulations of this paper, the MLS design outperforms all competing designs by far. Certainly, the performance of this design depends on the signal in the data. If the joint distribution of the dependent variable and the covariates is entirely different in the observational data than in the experimental data, the approach will not provide any benefits at all. Yet, in this cases, also researchers using classical designs might struggle finding the right covariates to stratify on. If there exists a common signal in the observational and the experimental sample, the random forest is quite effective in extracting this signal from the observational data. However, even if there exists a signal in the data, the MLS design does not necessarily have to outperform classical designs in any case. One could easily construct an example with one important covariate and 999 noise variables, in which the best choice would be to stratify exactly on the important variable and neglect the rest. Nevertheless, it is much easier for a computer to detect this one-in-a-thousand variable than for a human.

References

Ashraf, N., D. Karlan, and W. Yin (2006): "Tying Odysseus to the Mast: Evidence From a Commitment Savings Product in the Philippines*," *The Quarterly Journal of Economics*, 121, 635–672.

Athey, S. and G. Imbens (2017): "The Econometrics of Randomized Experiments," in *Handbook of Field Experiments*, ed. by A. V. Banerjee and E. Duflo, North-Holland, vol. 1 of *Handbook of Economic Field Experiments*, 73 – 140.

Aufenanger, T. (2017): "Treatment Allocation for Linear Models with Covariate Information," Discussion paper, FAU.

Berk, R. A. (2008): *Statistical Learning from a Regression Perspective*, vol. 14, Springer.

Berry, D. A. (2011): "Adaptive Clinical Trials: The Promise and the Caution," *Journal of Clinical Oncology*, 29, 606–609, pMID: 21172875.

Breiman, L. (2001): "Random Forests," *Machine Learning*, 45, 5–32.

Bruhn, M. and D. McKenzie (2009): "In Pursuit of Balance: Randomization in Practice in Development Field Experiments," *American Economic Journal. Applied Economics*, 1, 200–232.

Bubeck, S. and N. Cesa-Bianchi (2012): "Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems," *Foundations and Trends in Machine Learning*, 5, 1–122.

Féraud, R., R. Allesiardo, T. Urvoy, and F. Clérot (2016): "Random Forest for the Contextual Bandit Problem," in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, ed. by A. Gretton and C. C. Robert, Cadiz, Spain: PMLR, vol. 51 of *Proceedings of Machine Learning Research*, 93–101.

Fernández-Delgado, M., E. Cernadas, S. Barro, and D. Amorim (2014): "Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?" *Journal of Machine Learning Research*, 15, 3133–3181.

Finn, J. D., J. Boyd-Zaharias, R. M. Fish, and S. B. Gerber (2007): *Project STAR and Beyond - Database User's Guide*, HEROS, Incorporated.

Fisher, R. A. (1926): "The Arrangement of Field Experiments." *Journal of the Ministry of Agriculture of Great Britain*, 33, 503–513.

Graham, B. S. (2008): "Identifying Social Interactions Through Conditional Variance Restrictions," *Econometrica*, 76, 643–660.

Greevy, R., B. Lu, J. H. Silber, and P. Rosenbaum (2004): "Optimal Multivariate Matching Before Randomization," *Biostatistics*, 5, 263–275.

Hahn, J., K. Hirano, and D. Karlan (2011): "Adaptive Experimental Design Using the Propensity Score," *Journal of Business & Economic Statistics*, 29, 96–108.

Holland, P. W. (1986): "Statistics and Causal Inference," *Journal of the American Statistical Association*, 81, 945–960.

Imbens, G. W. and D. B. Rubin (2015): *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*, Cambridge University Press.

Kasy, M. (2016): "Why Experimenters Might Not Always Want to Randomize, and What They Could Do Instead," *Political Analysis*, please email me for password to the MATLAB files, which generate optimal designs for your data.

Kempthorne, O. (1955): "The Randomization Theory of Experimental Inference," *Journal of the American Statistical Association*, 50, 946–967.

Kernan, W. N., C. M. Viscoli, R. W. Makuch, L. M. Brass, and R. I. Horwitz (1999): "Stratified Randomization for Clinical Trials," *Journal of Clinical Epidemiology*, 52, 19 – 26.

Krueger, A. B. (1999): "Experimental Estimates of Education Production Functions," *The Quarterly Journal of Economics*, 114, 497–532.

Lu, B., R. Greevy, X. Xu, and C. Beck (2011): "Optimal Nonbipartite Matching and Its Statistical Applications," *The American Statistician*, 65, 21–30.

Lu, T., D. Pál, and M. Pál (2010): "Contextual Multi-armed Bandits," in *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, 485–492.

Mahajan, A. and D. Teneketzis (2008): "Multi-Armed Bandit Problems," in *Foundations and Applications of Sensor Management*, Springer Science & Business Media, 121–151.

Moore, R. T. (2012): "Multivariate Continuous Blocking to Improve Political Science Experiments," *Political Analysis*, 20, 460–479.

R Development Core Team (2008): *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0.

Robbins, H. (1952): "Some Aspects of the Sequential Design of Experiments," *Bull. Amer. Math. Soc.*, 58, 527–535.

Rubin, D. (1974): "Estimating Causal Effects of Treatments in Experimantal and Observational Studies," *Journal of Educational Psychology*, 66, 688 – 701.

Schneider, S. O. and M. Schlather (2017): "A New Approach to Treatment Assignment for One and Multiple Treatment Groups," Tech. Rep. 228, Courant Research Centre: Poverty, Equity and Growth - Discussion Papers, Göttingen.

Sverdlov, O., ed. (2015): *Modern Adaptive Randomized Clinical Trials: Statistical and Practical Aspects*, vol. 81, CRC Press.

Tang, L., R. Rosales, A. Singh, and D. Agarwal (2013): "Automatic Ad Format Selection via Contextual Bandits," in *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, New York, NY, USA: ACM, CIKM '13, 1587–1594.

Thompson, W. R. (1933): "On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples," *Biometrika*, 25, 285–294.

Villar, S. S., J. Bowden, and J. Wason (2015): "Multi-armed Bandit Models for the Optimal Design of Clinical Trials: Benefits and Challenges," *Statist. Sci.*, 30, 199–215.

Wager, S. and G. Walther (2016): "Adaptive Concentration of Regression Trees, with Application to Random Forests," *arXiv preprint arXiv:1503.06388*.

Wang, C.-C., S. R. Kulkarni, and H. V. Poor (2005): "Bandit Problems with Side Observations," *IEEE Transactions on Automatic Control*, 50, 338–355.

Wilk, M. B. (1955): "The Randomization Analysis of a Generalized Randomized Block Design," *Biometrika*, 42, 70–79.

Wyner, A. J., M. Olson, J. Bleich, and D. Mease (2017): "Explaining the Success of AdaBoost and Random Forests as Interpolating Classifiers," *Journal of Machine Learning Research*, 18, 1–33.

A Simulations for larger experimental data set

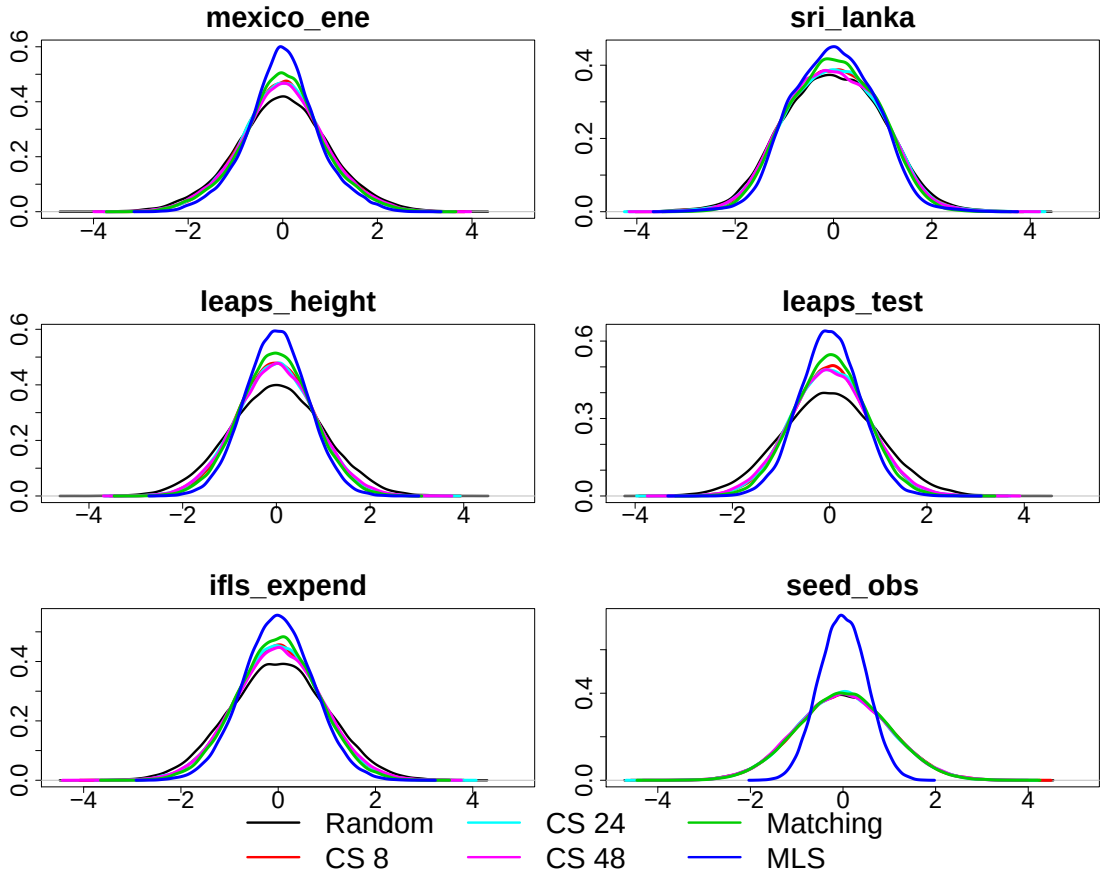


Figure 2: Distribution of difference in means estimator

In Section 3, I simulated the designs on experimental data sets of 32 units. In this section I increase the sample size to 96 units. The simulation approach remains the same. Figure 2 as well as Tables 3 and 4 show that also the performance of the designs remains very similar.

Some theoretical thoughts on this: As in Section 2.2, I assume random sampling from a large population. Consider a randomly drawn experimental sample U containing $\tilde{N}_s = k \cdot N_s$ units. Then w.l.o.g. U can be written as $U = (U^1, ..., U^k)$, where $U^j = (U^j_1, ..., U^j_{N_s})$ are sub samples of N_s experimental units for $j = 1, ..., k$.

The large population assumption ensures that $U^1, ..., U^k$ are independent and identically distributed. Regard the difference in means estimator on the sample U:

$$\hat{\tau} = \frac{2}{kN_s} \sum_{U_i^j \in U} W_i^j Y_i^j(1) - (1 - W_i^j) Y_i^j(0)) = \frac{1}{k} \sum_{U^j \in U} \hat{\tau}^j$$

Here $W_i^j \in \{0, 1\}$ indicates the treatment allocation of unit i out of sub sample j. Further $\hat{\tau}^j = \frac{2}{N_s} \sum_{U_i^j \in U^j} W_i^j Y_i^j(1) - (1 - W_i^j) Y_i^j(0))$ is the difference in means estimator in sub sample j. Whenever the allocation of treatments in sub sample j only depends on the units in U^j and not on the remaining sample, all $\hat{\tau}^j$ are independent for $j = 1, ..., k$. In this case:

$$\mathbb{V}[\hat{\tau}] = \frac{1}{k^2} \sum_{j=1}^{k} \mathbb{V}[\hat{\tau}^j] = \frac{1}{k} \mathbb{V}[\hat{\tau}^1]$$

Therefore the variance of the estimator of the large data set is given by $\frac{1}{k}$ times the variance on the small data set. Take a treatment allocation algorithm A that yields $\mathbb{V}[\hat{\tau}^1]_A = d \cdot \mathbb{V}[\hat{\tau}^1]_{cs}$, where $\mathbb{V}[\hat{\tau}^1]_{cs}$ is the variance of the estimator for random allocation and $0 < d < 1$. Then on the large data set:

$$\frac{\mathbb{V}[\hat{\tau}]_A}{\mathbb{V}[\hat{\tau}]_{cs}} = \frac{\mathbb{V}[\hat{\tau}^1]_A}{\mathbb{V}[\hat{\tau}^1]_{cs}} = d$$

This shows that the variance of the treatment estimator in terms of the variance for random allocation stays constant as the sample size increases.

For this result, I assumed that the treatment allocation of the sub sample U^j only depends on the units in U^j. This assumption will typically not be fulfilled. Take for example the matching design. To generate the same fraction of variances as in the simulation for 32 units, matching for the 96 unit sample should work as follows: Divide the 96 unit sample in three sub samples of 32 units. Then match units only inside the sub samples. Certainly this decreases performance of the matching design on large data sets, since better matches outside the subgroups are ignored. Consequently, $\frac{\mathbb{V}[\hat{\tau}^1]_A}{\mathbb{V}[\hat{\tau}^1]_{cs}}$ is typically only a lower bound on the fraction of variances on larger data sets $\frac{\mathbb{V}[\hat{\tau}]_A}{\mathbb{V}[\hat{\tau}]_{cs}}$.

Table 3 supports these theoretical considerations. For all data sets except for the SEED data, the variance of the estimator for all designs in terms of the variance for complete randomization either stay the same or slightly reduces compared to the simulation of Section 3.

Table 3: Performance of experimental design

]

	Variance of Estimator	Power	Number of Covariates	Training dataset size
mexico_ene:				
Random	1.00	0.71	30	332
CS.8	0.80	0.77	30	332
CS.24	0.80	0.77	30	332
CS.48	0.84	0.76	30	332
Matching	0.75	0.79	30	332
MLS	0.57	0.84	30	332
sri_lanka:				
Random1	1.00	0.69	34	299
CS.81	0.89	0.71	34	299
CS.241	0.91	0.71	34	299
CS.481	0.88	0.71	34	299
Matching1	0.77	0.74	34	299
MLS1	0.70	0.75	34	299
leaps_height:				
Random2	1.00	0.62	6	334
CS.82	0.67	0.77	6	334
CS.242	0.70	0.76	6	334
CS.482	0.72	0.74	6	334
Matching2	0.59	0.81	6	334
MLS2	0.44	0.90	6	334
leaps_test:				
Random3	1.00	0.65	7	334
CS.83	0.63	0.82	7	334
CS.243	0.64	0.81	7	334
CS.483	0.67	0.80	7	334
Matching3	0.53	0.87	7	334
MLS3	0.38	0.95	7	334
ifls_expend:				
Random4	1.00	0.72	7	334
CS.84	0.78	0.81	7	334
CS.244	0.77	0.81	7	334
CS.484	0.80	0.79	7	334
Matching4	0.68	0.84	7	334
MLS4	0.52	0.92	7	334
seed_obs:				
Random5	1.00	0.70	1476	594
CS.85	0.99	0.71	1476	594
CS.245	0.98	0.71	1476	594
CS.485	1.00	0.70	1476	594
Matching5	0.96	0.71	1476	594
MLS5	0.26	0.99	1476	594

Number of splits: 50, MC steps per split: 1000, Size of experimental dataset: 96

Table 4: Share of splits in which the algorithm performs best

	random	strata8	strata24	strata48	matching	strataForest
Variance of Estimator	0.00	0.00	0.01	0.01	0.11	0.88
Power	0.01	0.01	0.02	0.01	0.12	0.87

Number of splits: 300, MC steps per split: 1000, Size of experimental dataset: 96

B Heterogeneous treatment effects

In the simulations of Section 3, I assume a constant treatment effect. The motivation of Section 2.2 suggests that MLS performs particularly well in case of constant treatment effect, since in this case treated and untreated potential outcomes are perfectly correlated. To provide some evidence on the performance under heterogeneous treatment effects, this section presents some additional simulation that seeks to overtake the treatment effect heterogeneity from the data.

Simulation Procedure

For these simulations, I use experimental data sets. I match the experimental data on Mahalanobis distance, such that every unit from the treatment group has a match in the control group. Then I define the best k matches as the experimental sample. Out of the remaining observations, I only keep the control group and use it as the observational sample of untreated outcomes. In the group of the k best matches, every unit of the control group has an (almost) exact match in the treatment group. This means for every covariate vector X_i in this subgroup there is an untreated observation $Y_i(0)$ (the match in the control group) and a treated observation $Y_i(1)$ (the match in the treatment group). I simulate treatments by regarding $Y_i(1)$ instead of $Y_i(0)$. To ensure that the average treatment effect is equal to a given effect size c, I add a constant on the dependent variable of each treated outcome that is equal to c minus the current ATE in the sample. This does not affect the treatment effect heterogeneity.

In contrast to the simulations of Section 3, in the simulation of this section the experimental sample is not a random draw of the same population as the observational sample anymore. In order for the random forest predictions to be precise, the experimental data should contain the same signal as the observational data. If the experimental sample is entirely different regarding the connection between the covariates and the dependent variable, the MLS approach will not work anymore. This as well as the fact that I include treatment effect heterogeneity in this simulation should diminish the performance of the MLS design. As this section shows, the design nevertheless performs very well.

Data

I run this simulation on two data sets. The first data set is the SEED data already discussed in Section 3. For this simulation, I regard only a small subset of all covariates. The reason is that matches on 1,500 covariates in a data set of 1,700

units are necessarily very poor. In order to use the matched observation of treatment and control group as the potential outcomes of the same observation, those two units should look basically identical from the perspective of the experimental design. This means the covariates of the two units should be almost identical, such that any covariate-based allocation of treatments can not distinguish between the the treated and the untreated units of a match. In order to reduce the number of covariates, I run machine learning algorithms on the experimental data set to select the most important covariates with respect to the dependent variable. In particular, I run a LASSO regression to select the 18 most important categorical variables and a random forest to select the 31 most important ordinal variables. I neglect all covariates apart from those 49.

The second data set is the Tennessee STAR experiment (see Krueger (1999) or Graham (2008) for some related publications and Kasy (2016) for a comparison of experimental designs on this data). In 80 schools in Tennessee, the STAR experiment exogenously varies class sizes in from kindergarten to fourth grade and regards the effect on test performance in these as well as in later years. Out of this experiment, I regard the sample of subjects that have been in the first grade in the 1986-1987 school year. After deleting observations with many missings as well as outliers, this sub sample of the STAR experiment consist of around $5,600$ observations. The dependent variable in the simulated experiment is the average SAT test score at the end of grade 1, and the treatment variable is an assignment to a small class (13-17 students per teacher). Subjects that do not receive the treatment are either assigned to a regular class (22-25 students per teacher) or a regular class with special aid (see Finn et al. (2007) for more details). Unfortunately, the STAR experiment does not report many demographic variables on the subjects. I use seven covariates in the design: The age, gender and race of the subject, whether the subject has attended kindergarten, whether the subject received a free lunch in grade 1 (an indicator of a poor parenthood), and the school id as well as an urban/rural indicator for the school.

Results

On the STAR data, the 32 matches are perfect, i.e., subject of one match are identical in every covariate. For the SEED data there are no perfect matches, but I pick the best out of more than 800 matches.

Figure 3 shows the treatment effect heterogeneity in the experimental samples. The figure plots the treatment effect for each match, i.e., the treated outcome minus the untreated outcome. Especially the STAR data involves a considerable amount of heterogeneity in the treatment effects.

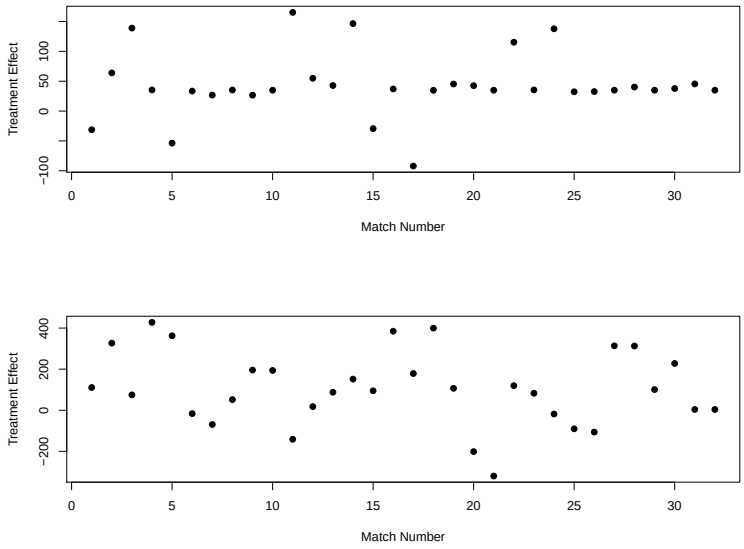


Figure 3: Treatment Effect by Match Number

Table 5 presents the variance of the difference in means estimator as well as the power. As in Section 3, the MLS design performs better in both measures. Surprisingly, the matching design in the SEED data performs far worse than complete randomization. At the first glance, this appears to contradict Section 2.2, which showed that stratification always results in a lower variance of the estimator than complete randomization. Note however, that this result holds only on average over a large number of experiments sampled randomly from a large population. In contrast to Section 3, in this section I regard only one (not random but selected) sample. The results should therefore be treated with care.

Table 5: Performance of experimental designs for heterogeneous treatment effects

	Variance of Estimator	Power	Number of Covariates	Training dataset size
star_32:				
Random	1.00	0.63	7	3948
CS.8	0.90	0.66	7	3948
Matching	0.89	0.68	7	3948
MLS	0.57	0.76	7	3948
seed_32:				
Random1	1.00	0.60	49	711
CS.81	0.89	0.63	49	711
Matching1	1.46	0.43	49	711
MLS1	0.21	1.00	49	711

Number of MC steps: 50000, Size of experimental dataset: 32