Precise Unbiased Estimation in Randomized Experiments using Auxiliary Observational Data

Johann A. Gagnon-Bartsch*[1], Adam C. Sales*[2], Edward Wu[1], Anthony F. Botelho[3], Luke W. Miratrix[4], and Neil T. Heffernan[3]

[1]University of Michigan, Department of Statistics
[2]University of Texas, Austin, College of Education
[3]Worcester Polytechnic Institute, Learning Sciences and Technologies
[4]Harvard University, School of Education

February 2, 2020

Abstract

Randomized controlled trials (RCTs) are increasingly prevalent in education research, and are often regarded as a gold standard of causal inference. Two main virtues of randomized experiments are that they (1) do not suffer from confounding, thereby allowing for an unbiased estimate of an intervention's causal impact, and (2) allow for design-based inference, meaning that the physical act of randomization largely justifies the statistical assumptions made. However, RCT sample sizes are often small, leading to low precision; in many cases RCT estimates may be too imprecise to guide policy or inform science. Observational studies, by contrast, have strengths and weaknesses complementary to those of RCTs. Observational studies typically offer much larger sample sizes, but may suffer confounding. In many contexts, experimental and observational data exist side by side, allowing the possibility of integrating "big observational data" with "small but high-quality experimental data" to get the best of both. Such approaches hold particular promise in the field of education, where RCT sample sizes are often small due to cost constraints, but automatic collection of observational data, such as in computerized educational technology applications, or in state longitudinal data systems (SLDS) with administrative data on hundreds of thousand of students, has made rich, high-dimensional observational data widely available. We present a framework that allows one to employ machine learning algorithms to learn from the observational data, and use the resulting models to improve precision in randomized experiments. Importantly, there is no requirement that the machine learning models

*These authors contributed equally.

are "correct" in any sense, and the final experimental results are guaranteed to be exactly unbiased. Thus, there is no danger of confounding biases in the observational data leaking into the experiment.

1 Introduction

Randomized controlled trials (RCTs) are an increasingly common tool across scientific and commercial domains. RCTs have long been a prominent feature of the biomedical sciences and pharmaceutical industry, and are increasingly prevalent across the social sciences as well. Large internet companies such as Microsoft and Amazon conduct tens of thousands of RCTs (often referred to as "A/B tests") each year [Kohavi and Thomke, 2017], as do smaller start-up companies [Ries, 2011].

An example from the field of education is the ASSISTments TestBed [Heffernan and Heffernan, 2014, Ostrow et al., 2016]. ASSISTments is a computer-based learning platform used by over 50,000 students throughout the United States each year for homework and classwork. The TestBed is an A/B testing program designed for conducting education research that runs within ASSISTments, and has been made accessible to third-party education researchers. Using the TestBed, a researcher can propose A/B tests to run within ASSISTments. For example, a researcher may propose that on one night's homework, for a given topic such as "Adding Whole Numbers", or "Factoring Quadratic Equations," students are individually randomized into one of two conditions, e.g., video- or text-based instructional feedback. The researcher could then compare the relative impact of video- vs. text-based feedback on an outcome variable of interest such as homework completion. The anonymized data associated with the study, consisting of several levels of granularity and a breadth of covariates describing both historical pre-study and within-study student interaction, is made available to the researcher. The Testbed is currently host to over 100 such RCTs running in ASSISTments, and several of these RCTs have recently been analyzed [Fyfe, 2016, Walkington et al., 2019, McGuire et al., 2017, Koedinger and McLaughlin, 2016].

RCTs are famously free of confounding bias. Indeed, a class of estimators, often referred to as "design-based" [Schochet, 2015] or "randomization based" [Rosenbaum, 2002a] estimate treatment effects without assuming any statistical model other than whatever is implied by the experimental design itself. Design-based statistical estimators are typically guaranteed to be unbiased. Their associated inference—standard errors, hypothesis tests, confidence intervals—also come with guarantees regarding accuracy. In many cases, these apply regardless of the sample size or the characteristics of the data's parent distribution.

While RCTs can reliably provide unbiased estimates, they are often limited in terms of precision. The statistical precision of RCT-based estimates is inherently limited by the RCT's sample size, which itself is typically subject to a number of practical constraints. For instance, in one typical ASSISTments TestBed A/B test, a total of 294 students were randomized between two conditions, leading to a standard error of roughly four percentage points when estimating the effect on homework completion. This standard error is too large to either determine the direction of a treatment effect or rule out clinically meaningful effect

sizes.

In contrast, large observational datasets can frequently be brought to bear on some of the same questions addressed by an RCT. Analysis of observational data, unlike RCTs, typically requires a number of untestable modeling assumptions, chief among them the assumption of no unmeasured confounding. Consequently, treatment effect estimates from observational data cannot boast the same guarantees to accuracy as estimates from RCTs. That said, in many cases they boast a much larger sample—and, hence, greater precision—than equivalent RCTs.

In many cases, observational and RCT data coexist within the very same database. For instance, covariate and outcome data for a biomedical RCT may be drawn from a database of electronic health records, and that same database may contain equivalent records for patients who did not participate in the study and were not randomized. Along similar lines, covariate and outcome data for an RCT designed to evaluate the impact of an educational intervention might be drawn from a state administrative database, and that database may also contain information on hundreds of thousands of students who did not participate in the RCT. Also similarly, in the ASSISTments TestBed example, a given RCT is likely to consist of just a few hundred students assigned to a specific homework assignment, but the ASSISTments database contains data on hundreds of thousands of other ASSISTments users, many of whom may have completed similar homework assignments, or who may have even completed an identical homework assignment but in a previous time period. We refer to these individuals, who are non-participants of the RCT but who are in the same database, as the *remnant* from the study [Sales et al., 2018b].

This paper presents a novel method to estimate treatment effects in an RCT while incorporating high-dimensional covariate data, large observational remnant data, and machine learning prediction algorithms to improve precision. It does so without compromising the accuracy guarantees of traditional design-based RCT estimators, yielding unbiased point estimates and sampling variance estimates that are conservative in expectation; in fact our method is itself design-based, relying only on the randomization within the RCT to make these guarantees. In particular, the method prevents "bias leakage": bias that might have occurred due to differences between the remnant and the experimental sample, biased or incorrect modeling of covariates, or other data analysis flaws, does not leak into the RCT estimator. Our approach combines two recent causal methods: rebar [Sales et al., 2018a, Botelho et al., 2018], which incorporates high dimensional remnant data into RCT estimators, and LOOP [Wu and Gagnon-Bartsch, 2018], which uses machine learning algorithms for within-RCT covariate adjustment. This paper will focus on the challenge of precisely estimating treatment effects from a published set of 22 TestBed experiments [Selent et al., 2016], using prior log data from experimental participants and non-participants in the ASSISTments system.

The nexus of machine learning and causal inference has recently experienced rapid and exciting development. This has included novel methods to analyze observational studies [e.g. Diamond and Sekhon, 2013], to estimate subgroup effects [e.g. Duivesteijn et al., 2017, Künzel et al., 2018], or to optimally allocate treatment (e.g. Rzepakowski and Jaroszewicz

[2012], Zhao and Heffernan [2017]). Other developments share our goal, i.e. improving the precision of average treatment effect estimates from RCTs. These include Wager et al. [2016], Chernozhukov et al. [2018], Bloniarz et al. [2016], which uses the Lasso regression estimator [Tibshirani, 1996] to analyze experiments [also see Belloni et al., 2014], and the Targeted Learning [Van der Laan and Rose, 2011] framework, which combines ensemble machine-learning with semiparametric maximum likelihood estimation, however none of these methods make use of auxiliary observational data.

A large literature has explored the possibility of improving precision in RCTs by pooling the controls in the RCT with historical controls from observational datasets or from other similar RCTs. This literature dates back at least to Pocock [1976]; for more recent reviews see, e.g., Viele et al. [2014] and Lim et al. [2018]. Much of this work uses a Bayesian framework, although frequentist approaches exist as well [Yuan et al., 2019]. To the best of our knowledge, however, none of these methods offer design-based guarantees of unbiasedness, and indeed in many cases biases can be arbitrary large depending on the choice of historical controls. Other literature has sought to combine experimental and observational data for other purposes. In particular, recent work has sought to use observational data to extrapolate the results of an RCT to a larger population [Hartman et al., 2015, Kallus et al., 2018, Rosenman et al., 2018], and the estimand of interest is the population average treatment effect (PATE).

In this paper, our goal is to estimate the average treatment effect within the RCT, and our focus is on using observational data to improve the precision of the estimate. Our approach may be summarized as "first, do no harm," meaning that we prioritize retaining the advantages of randomized experiments highlighted above. In particular, we seek to ensure that our method (1) does not introduce any bias, (2) will not harm precision, and ideally will improve precision, and (3) does not require any additional statistical assumptions beyond those typically made in design-based analysis of RCTs.

The paper is organized as follows. Section 2 reviews background material, including design-based RCT analysis and covariate adjustment. Section 3 discusses incorporating remnant data, and presents our main methodological contribution. In Section 4 we apply the method to estimate treatment effects in the 22 TestBed experiments. Section 5 concludes.

2 Methodological Background

2.1 Causal Inference from Experiments

Consider a randomized experiment to estimate the average effect of a binary treatment T on an outcome Y. There are N subjects, indexed by $i = 1, \ldots, N$. Let $T_i = 1$ if subject i is assigned to treatment, and $T_i = 0$ if control. Let $\mathcal{T} = \{i \mid T_i = 1\}$ and $\mathcal{C} = \{i \mid T_i = 0\}$, and let $n_t = |\mathcal{T}|$ and $n_c = |\mathcal{C}|$.

Following Neyman [1923] and Rubin [1974], let potential outcomes y_i^t and y_i^c represent the outcome value Y_i that i would have exhibited if he or she had (perhaps counterfactually) been assigned to treatment or control, respectively. Observed outcomes are a function of

treatment assignment and potential outcomes:

$$Y_i = T_i y_i^t + (1 - T_i) y_i^c$$

Define the treatment effect for i as $\tau_i = y_i^t - y_i^c$. Our goal will be to estimate the average treatment effect (ATE), $\bar{\tau} \equiv \sum_i \tau_i / N = \bar{y}^t - \bar{y}^c$, where $\bar{y}^t = \sum_{i=1}^N y_i^t / N$ is the mean of y^t over all N units in the experiment and $\bar{y}^c$ is defined similarly.

If both y_i^c and y_i^t were known for each subject i, statistical modeling would be unnecessary—researchers could calculate $\bar{\tau}$ exactly, without error, by simply averaging observed τ. In practice, we never observe both y_i^c and y_i^t. Instead, we rely on the experimental setup to estimate and infer causation. Since the treatment and control groups are each random samples of the N participants, survey sampling literature provides design-based unbiased estimators of $\bar{y}^t$ and $\bar{y}^c$ based on observed Y and the known distribution of T. These estimators, and their associated inference, depend only on the experimental design, and not on modeling assumptions. The survey sample structure of randomized experiments allows us to infer counterfactual potential outcomes (at least on average) and estimate $\bar{\tau}$ as if τ_i were available for each i, albeit with sampling error.

We will use this framework to analyze the 22 TestBed experiments. These experiments are examples of "Bernoulli experiments," in which each T_i is an independent Bernoulli trial: $\mathbb{P}(T_i = 1) = p$, with $0 < p < 1$, and $T_i \perp\!\!\!\perp T_j$ if $i \neq j$. In the TestBed experiments, $p = 1/2$. Estimation and inference about $\bar{\tau}$ is based on the observed values of Y and T, and the known value of p.

Let $M_i = T_i y_i^c + (1 - T_i) y_i^t$ denote i's unobserved counterfactual outcome—when i is treated, $M_i = y_i^c$ and when i is in the control condition $M_i = y_i^t$. Then i's treatment effect may be expressed as $\tau_i = (-1)^{T_i}(M_i - Y_i)$, i.e., $\tau_i = M_i - Y_i$ if i is in the control group, or $\tau_i = Y_i - M_i$ if i is in the treatment group. Although M_i is, by definition, unobserved, it plays a central role in causal inference; its expectation,

$$m_i \equiv \mathbb{E} M_i = p y_i^c + (1 - p) y_i^t$$

will also play a prominent role in the method we are proposing.

Let

$$U_i = \begin{cases} \frac{1}{p} & T_i = 1 \\ -\frac{1}{1-p} & T_i = 0 \end{cases}$$

be subject i's signed inverse probability weights. Note that $\mathbb{E} U_i = 0$, and $\mathbb{E} U_i Y_i = \tau_i$. To see the latter, note that when $T = 1$, with probability p, $Y_i = y_i^t$ and $U_i Y_i = y_i^t / p$; when $T = 0$, with probability $1 - p$, $U_i Y_i = -y_i^c / (1 - p)$. Thus $U_i Y_i$ may be thought of as an unbiased estimate of τ_i, and $\hat{\tau}^{\text{IPW}} \equiv \sum_i U_i Y_i / N$ is an unbiased estimate of $\bar{\tau}$. In fact, $\hat{\tau}^{\text{IPW}}$ is identical to the "Horvitz-Thompson" estimator of, e.g., Aronow and Middleton [2013]

$$\hat{\tau}^{\text{IPW}} = \frac{1}{N} \sum_{i \in \mathcal{T}} \frac{Y_i}{p} - \frac{1}{N} \sum_{i \in \mathcal{C}} \frac{Y_i}{1 - p} \tag{1}$$

since it is the difference between the Horvitz-Thomson estimates of $\bar{y}^t$ and $\bar{y}^c$ [Horvitz and Thompson, 1952].

The sampling variance of $\hat{\tau}^{\text{IPW}}$ proceeds from the same principals. The variance of $U_i Y_i$ is

$$\mathbb{V}(U_i Y_i) = \left(y_i^t \sqrt{\frac{1-p}{p}} + y_i^c \sqrt{\frac{p}{1-p}} \right)^2 = \frac{m_i^2}{p(1-p)}$$

and $\mathbb{V}(\hat{\tau}^{\text{IPW}}) = \sum_i m_i^2 / [N^2 p(1-p)]$ because treatment assignments are independent. Note that because y_i^t and y_i^c are never simultaneously observed, $\mathbb{V}(\hat{\tau}^{\text{IPW}})$ is not identified. However, $\hat{\mathbb{V}}(\hat{\tau}^{\text{IPW}}) = \sum_i U_i^2 Y_i^2 / N^2$ is an upper bound, i.e., $\mathbb{E}\hat{\mathbb{V}}(\hat{\tau}^{\text{IPW}}) \geq \mathbb{V}(\hat{\tau}^{\text{IPW}})$. (See Aronow and Middleton 2013 for equivalent expressions for more general experimental designs.)

Strangely, $\hat{\tau}^{\text{IPW}}$ and $\mathbb{V}(\hat{\tau}^{\text{IPW}})$ are not translation-independent, i.e., adding a constant to each Y changes both the value of $\hat{\tau}^{\text{IPW}}$ and $\mathbb{V}(\hat{\tau}^{\text{IPW}})$ without changing the estimand $\bar{\tau}$. The more popular "simple difference" estimator [Neyman, 1923],

$$\hat{\tau}^{\text{SD}} = \frac{1}{n_t} \sum_{i \in \mathcal{T}} Y_i - \frac{1}{n_c} \sum_{i \in \mathcal{C}} Y_i = \bar{Y}_{\mathcal{T}} - \bar{Y}_{\mathcal{C}} \tag{2}$$

and its associated variance estimator

$$\hat{\mathbb{V}}(\hat{\tau}^{\text{SD}}) = \frac{S^2(Y_{\mathcal{C}})}{n_c} + \frac{S^2(Y_{\mathcal{T}})}{n_t} \tag{3}$$

where $S^2(Y_{\mathcal{C}}) = \sum_{i \in \mathcal{C}} (Y_i - \bar{Y}_{\mathcal{C}})^2 / (n_c - 1)$ is the sample variance of the control group and $S^2(Y_{\mathcal{T}})$ is defined similarly, do not have this undesirable property. Our presentation here focuses on $\hat{\tau}^{\text{IPW}}$ as a jumping-off point for subsequent methodological development, but $\hat{\tau}^{SD}$ will also play a prominent role.

2.2 Design-Based Covariate Adjustment

The reason for error when estimating τ is our inability to observe counterfactual potential outcomes M. As we have seen, randomized trials, coupled with design-based estimators like $\hat{\tau}^{\text{IPW}}$, use comparison groups and survey sampling theory to implicitly fill in this missing information. Baseline covariates—a vector $\boldsymbol{x}_i$ of data for subject i gathered prior to treatment randomization—may potentially help us improve upon this strategy. To see how, suppose a researcher has constructed algorithms $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$ designed to impute y^c and y^t, respectively, from $\boldsymbol{x}$. Then $\hat{M}_i = T_i \hat{y}^c(\boldsymbol{x}_i) + (1 - T_i)\hat{y}^t(\boldsymbol{x}_i)$ is an imputation of i's missing counterfactual outcome, and the researcher may estimate τ_i as $(-1)^{T_i}(\hat{M}_i - Y_i)$. In general, the bias of algorithms such as $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$ will be unknown without further assumptions, so these effect estimates may be inadvisable. On the other hand, imperfect or potentially biased imputations of potential outcomes can, *when combined with randomization*, yield substantial benefits.

The approach we will take to combining covariate adjustment with randomization follows Wu and Gagnon-Bartsch [2018], as will its presentation here. It has antecedents in Robins et al. [1994], Scharfstein et al. [1999], Robins [2000], Rosenbaum [2002b], Bang and Robins [2005], van der Laan and Rubin [2006], Tsiatis et al. [2008], Moore and van der Laan [2009],

Van der Laan and Rose [2011], Aronow and Middleton [2013], Belloni et al. [2014], Wager et al. [2016], and Chernozhukov et al. [2018], among others.

In a Bernoulli experiment, note that

$$\begin{aligned} U_i(Y_i - m_i) &= \begin{cases} \frac{1}{p}(y_i^t - py_i^c - (1-p)y_i^t) & T_i = 1 \\ -\frac{1}{1-p}(y_i^c - py_i^c - (1-p)y_i^t) & T_i = 0 \end{cases} \\ &= \begin{cases} \frac{p(y_i^t - y_i^c)}{p} & T_i = 1 \\ \frac{(1-p)(y_i^t - y_i^c)}{1-p} & T_i = 0 \end{cases} \\ &= \tau_i \end{aligned}$$

and this therefore suggests using imputations $\hat{y}^c(\boldsymbol{x}_i)$ and $\hat{y}^t(\boldsymbol{x}_i)$ to estimate m_i as $\hat{m}_i = p\hat{y}^c(\boldsymbol{x}_i) + (1-p)\hat{y}^t(\boldsymbol{x}_i)$, and then estimating τ_i as

$$\hat{\tau}_i \equiv U_i(Y_i - \hat{m}_i).$$

It turns out that $\hat{\tau}_i$ is unbiased if algorithms $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$ are constructed in such a way that

$$\{\hat{y}^c(\boldsymbol{x}_i), \hat{y}^t(\boldsymbol{x}_i)\} \perp\!\!\!\perp T_i. \tag{4}$$

Since $\boldsymbol{x}_i \perp\!\!\!\perp T_i$ by design, (4) is tantamount to requiring that T_i, and variables such as Y_i that depend on T_i, play no role in constructing algorithms $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$.

Under (4), $\hat{\tau}_i$ is indeed unbiased:

$$\mathbb{E}(\hat{\tau}_i) = \mathbb{E}(U_i Y_i) - \mathbb{E}(U_i \hat{m}_i) = \mathbb{E}(U_i Y_i) - \mathbb{E}(U_i)\mathbb{E}(\hat{m}_i) = \mathbb{E}(U_i Y_i) = \tau_i$$

where we use the facts that $\mathbb{E}(U_i) = 0$ and $\mathbb{E}(U_i Y_i) = \tau_i$. Finally, define the ATE estimate:

$$\hat{\tau} = \frac{1}{N}\sum_{i=1}^{N} \hat{\tau}_i = \frac{1}{N}\sum_{i \in \mathcal{T}} \frac{Y_i - \hat{m}_i}{p} - \frac{1}{N}\sum_{i \in \mathcal{C}} \frac{Y_i - \hat{m}_i}{1-p} \tag{5}$$

The unbiasedness of $\hat{\tau}$ for $\bar{\tau}$ follows from the unbiasedness of each of its summands, $\hat{\tau}_i$ for τ_i.

Crucially, this unbiasedness holds even if $\hat{y}^c(\boldsymbol{x}_i)$ and $\hat{y}^t(\boldsymbol{x}_i)$ are biased; algorithms $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$ need not be unbiased, consistent, or correct in any sense. As long as $\hat{y}^c(\boldsymbol{x}_i)$ and $\hat{y}^t(\boldsymbol{x}_i)$ are constructed to be independent of T_i, then $\hat{\tau}_i$ will be unbiased. The same cannot be said for regression-based covariate adjustment, the common technique of regressing Y on T and $\boldsymbol{x}$ [Freedman, 2008].

Compare the estimate $\hat{\tau}$ given in (5) to the estimate $\hat{\tau}^{\text{IPW}}$ given in (1). The only difference is that Y_i in (5) has been replaced by $Y_i - \hat{m}_i$ in (1). The goal of this covariate adjustment is to improve precision. Its success in this regard depends on the predictive accuracy of $\hat{y}^c(\boldsymbol{x}_i)$ and $\hat{y}^t(\boldsymbol{x}_i)$. Wu and Gagnon-Bartsch [2018] show that

$$\mathbb{V}(\hat{\tau}_i \mid \hat{m}_i) = \frac{(\hat{m}_i - m_i)^2}{p(1-p)}. \tag{6}$$

Accurate imputations of y_i^c and y_i^t, and hence of $\hat{m}_i$, yield precise estimation of τ_i. On the other hand, inaccurate imputations, i.e. when $(\hat{m}_i - m_i)^2$ is greater than m_i, will decrease precision—though, again, without causing bias.

2.3 LOOP

Successful covariate adjustment requires imputations $\hat{y}^c(\boldsymbol{x}_i)$ and $\hat{y}^t(\boldsymbol{x}_i)$ that are accurate and independent of T_i. To satisfy the independence condition, i's observed outcome Y_i, which is a function of T_i, cannot play a role in the construction of the algorithms $\hat{y}^c(\cdot)$ and $\hat{y}^t(\cdot)$; they must be trained using other data.

The "Leave-One-Out Potential outcomes," or "LOOP" estimator does so by fitting a separate imputation model for each experimental participant i, using data from the other participants. The algorithm proceeds as follows: for each i, one first drops observation i, and then use the remaining $N-1$ observations to construct imputation models for the control and treatment potential outcomes, denoted $\hat{y}^c_{-i}(\cdot)$ and $\hat{y}^t_{-i}(\cdot)$, respectively. These models may be fit by any method, for example linear regression or random forests [Breiman, 2001]. In particular, methods that allow for regularization to prevent overfitting may be used.

In this leave-one-out context,

$$\hat{m}_i = p\hat{y}^c_{-i}(\boldsymbol{x}_i) + (1-p)\hat{y}^t_{-i}(\boldsymbol{x}_i)$$

and the LOOP estimator is then again given by $\hat{\tau} = \sum_i \hat{\tau}_i/N$ as in (5). Note that in a Bernoulli experiment $\hat{m}_i \perp\!\!\!\perp T_i$ due to the fact that $\hat{m}_i$ is computed using $\boldsymbol{x}_i$ and a model fit without using observation i. It follows that $\hat{\tau}$ is unbiased. Other randomization designs will call for modifications to the algorithm (see, e.g., Wu and Gagnon-Bartsch [2019] for matched pair designs).

Wu and Gagnon-Bartsch [2018] provide an estimate for the variance of the LOOP estimator, building upon (6). Let

$$\hat{E}^2_c = \frac{1}{n_c}\sum_{i\in\mathcal{C}}\left[\hat{y}^c_{-i}(\boldsymbol{x}_i) - y^c_i\right]^2 \tag{7}$$

and define $\hat{E}^2_t$ similarly. Note $\hat{E}^2_c$ and $\hat{E}^2_t$ are leave-one-out cross validation mean squared errors. The estimated variance is then given by

$$\hat{\mathbb{V}}(\hat{\tau}) = \frac{1}{N}\left[\frac{p}{1-p}\hat{E}^2_c + \frac{1-p}{p}\hat{E}^2_t + 2\sqrt{\hat{E}^2_c\hat{E}^2_t}\right]. \tag{8}$$

Wu and Gagnon-Bartsch [2018] note that (8) will typically be somewhat conservative. This is due to the fact that $\mathbb{V}(\hat{\tau})$ is unidentifiable because the correlation of the potential outcomes is not estimable. This difficulty is not unique to the LOOP estimator; as noted in Section 2.1, similar comments apply to $\hat{\tau}^{\text{IPW}}$, and the same is true of $\hat{\tau}^{\text{SD}}$ as well [Neyman, 1923, Aronow et al., 2014].

Note that by simple algebraic inequality

$$\begin{aligned}\hat{\mathbb{V}}(\hat{\tau}) &\leq \frac{\hat{E}^2_c}{N(1-p)} + \frac{\hat{E}^2_t}{Np} \\ &\approx \frac{\hat{E}^2_c}{n_c} + \frac{\hat{E}^2_t}{n_t}\end{aligned} \tag{9}$$

which is similar in form to the variance estimate typically used in a two-sample t-test, namely $\frac{S^2(Y_\mathcal{C})}{n_c} + \frac{S^2(Y_\mathcal{T})}{n_t}$. In (9), $S^2(Y_\mathcal{C})$ and $S^2(Y_\mathcal{T})$ are replaced by $\hat{E}_c^2$ and $\hat{E}_t^2$. In other words, the sample variances are replaced by the estimated mean squared errors of the imputations.

A special case of the LOOP estimator occurs when the potential outcomes are imputed by simply taking the mean of the observed outcomes (after dropping observation i). That is, we set

$$\hat{y}^c_{-i}(\boldsymbol{x}_i) = \frac{1}{|\mathcal{C} \setminus i|} \sum_{j \in \mathcal{C} \setminus i} y^c_j$$

and similarly for $\hat{y}^t_{-i}(\boldsymbol{x}_i)$. Note that in this case the covariates are simply ignored. It can be shown that when the potential outcomes are mean-imputed in this manner the LOOP estimator $\hat{\tau}$ is exactly equal to the simple difference (2) estimator [Wu and Gagnon-Bartsch, 2018]. Moreover, in this special case $\hat{E}_c^2 = \frac{n_c}{n_c - 1} S^2(Y_\mathcal{C})$ and $\hat{E}_t^2 = \frac{n_t}{n_t - 1} S^2(Y_\mathcal{C})$ and thus the variance estimate given by (9) is nearly identical to the ordinary t-test variance estimate.

In short, when using mean imputation for the potential outcomes, LOOP essentially simplifies to ordinary simple difference estimation, as in a t-test. The effect estimate is identical, and the variance estimate is nearly identical. This is highly reassuring. Any imputation strategy that improves upon mean-imputation in terms of mean squared error will reduce the variance of the LOOP estimator relative to the simple difference estimator. Most modern machine learning methods employ some form of regularization to guard against overfitting, and thus typically perform no worse, or at least not substantially worse, than mean-imputation. Thus in practice there is relatively little risk that LOOP will hurt precision.

3 Incorporating the Remnant

Modern field trials are often conducted within a very data-rich context, in which rich high-dimensional covariate data is automatically, or already, collected for all experiment participants. For instance, in the TestBed experiments, system administrators have access to log data for every problem and skill builder each participating student worked before the onset of the experiment. In other contexts, such as healthcare or education, rich administrative data is often available. In fact, these covariates are available for a much wider population than just the experimental participants—in the TestBed case, there is log data for all ASSISTments users. In other education or healthcare examples, administrative data is often available for every student or patient in the system, not just for those who were randomized to a treatment or control condition. Often, as in the TestBed case, the outcome variable Y is also drawn from administrative or log data. We refer to subjects within the same data system in which the experiment took place—i.e. for whom covariate and outcome data are available—but who were not part of the experiment, as the "remnant" from the experiment. The remnant from a TestBed experiment consists of all ASSISTments users for whom log data is available but who did not participate in the experiment, of whom there are several hundred thousand.

3.1 Rebar: Covariate Adjustment Using the Remnant

Clearly, pooling data from the remnant with data from the experiment undermines the randomization. On the other hand, Sales et al. [2018a] argue that data from the remnant can play a role in covariate adjustment. In particular, an analyst may train an algorithm $\hat{y}^r(\cdot)$ to predict outcomes from covariates using data from the remnant, and then apply the trained algorithm to the experimental participants, using their covariates $\boldsymbol{x}$ to predict their outcomes. That is, for each participant i in the experiment, the analyst would compute $\hat{y}^r(\boldsymbol{x}_i)$, where $\hat{y}^r(\cdot)$ is trained using only data from the remnant.

It may not be clear precisely what $\hat{y}^r(\boldsymbol{x}_i)$ predicts. For example, if the experimental condition consists of some specific intervention while the control condition is "business as usual," and if everyone in the remnant experiences "business as usual," then we might view $\hat{y}^r(\boldsymbol{x}_i)$ as an imputation of subject i's control potential outcome. If, however, the subjects in the RCT are a non-representative sample of the overall population, then we may not even wish to view $\hat{y}^r(\boldsymbol{x}_i)$ as an imputation of subject i's control potential outcome, and the interpretation of $\hat{y}^r(\boldsymbol{x}_i)$ is more opaque.

Setting aside this issue for now, rebar proceeds as follows. First, one defines residuals $R_i \equiv Y_i - \hat{y}^r(\boldsymbol{x}_i)$. These residuals may be thought of as outcome variables, and in particular, R_i may be thought of as having two potential outcomes, r_i^c and r_i^t. If subject i is in control, then $R_i = y_i^c - \hat{y}^r(\boldsymbol{x}_i) = r_i^c$, and if i is in treatment, then $R_i = y_i^t - \hat{y}^r(\boldsymbol{x}_i) = r_i^t$. Importantly, note that $\hat{y}^r(\boldsymbol{x}_i)$ is invariant to treatment assignment, and $\hat{y}^r(\boldsymbol{x}_i)$ is therefore identical in both the control and treatment conditions. This is because the covariates $\boldsymbol{x}$ are pre-treatment, and because the function $\hat{y}^r(\cdot)$ is trained on the remnant and therefore unaffected by treatment assignments in the RCT. As a result, residualization leaves the estimand intact, i.e.,

$$r_i^t - r_i^c = \left[y_i^t - \hat{y}^r(\boldsymbol{x}_i)\right] - \left[y^c - \hat{y}^r(\boldsymbol{x}_i)\right] = y_i^t - y_i^c = \tau_i.$$

Therefore, the outcome Y may be replaced with the residualized outcome R in any unbiased estimator of the ATE, and the result will be an unbiased estimate of $\bar{\tau}$. For instance, treatment effects may be estimated by replacing Y_i with with R_i in the Horvitz-Thompson estimator (1); then, the rebar estimator is equivalent to $\hat{\tau}$ in (5), with $\hat{m}_i = \hat{y}^r(\boldsymbol{x}_i)$. Alternatively, treatment effects may be estimated as the simple difference of R between treatment and control groups, i.e.,

$$\hat{\tau}^{\text{Rebar}} = \frac{1}{n_t}\sum_{i\in\mathcal{T}} R_i - \frac{1}{n_c}\sum_{i\in\mathcal{C}} R_i = \bar{R}_{\mathcal{T}} - \bar{R}_{\mathcal{C}} \tag{10}$$

and with variance estimator

$$\hat{\mathbb{V}}(\hat{\tau}^{\text{Rebar}}) = \frac{S^2(R_{\mathcal{C}})}{n_c} + \frac{S^2(R_{\mathcal{T}})}{n_t}. \tag{11}$$

In what follows we will refer specifically to the simple difference (10) as "the rebar estimator."[1]

[1]Note that this differs slightly from the approach in Sales et al. [2018a], which seeks to estimate the average effect of an intervention on the treatment group only.

Importantly for practitioners, as long as only remnant data is used, $\hat{y}^r(\cdot)$ may be trained and assessed in any way. This process can be iterative, so that an analyst may train a candidate model, assess its performance (perhaps with $k-$fold cross-validation), modify the algorithm, and repeat until suitable performance is achieved. Any approach to modeling may be used, so long as no data from the RCT is used. The frequent problem of post-selection inference, which would be a serious concern if modeling were done instead on the RCT data (especially when the dimension of $\boldsymbol{x}$ is large and the sample size is small), does not apply here. Moreover, the guarantee that $\hat{\tau}^{\text{Rebar}}$ is unbiased does *not* require that $\hat{y}^r(\cdot)$ itself be unbiased, consistent, or "correct" in any sense; indeed, as hinted at above, it need not even be clear precisely what $\hat{y}^r(\cdot)$ is estimating. However $\hat{y}^r(\cdot)$ is fit to the remnant, the rebar estimator is guaranteed to be exactly unbiased due to the randomization in the RCT.

The goal of rebar is to improve precision. As noted above, the variance estimate of $\hat{\tau}^{\text{Rebar}}$ is $S^2(R_{\mathcal{C}})/n_c + S^2(R_{\mathcal{T}})/n_t$, while that of $\hat{\tau}^{\text{SD}}$ is $S^2(Y_{\mathcal{C}})/n_c + S^2(Y_{\mathcal{T}})/n_t$. We therefore desire $S^2(R_{\mathcal{C}}) < S^2(Y_{\mathcal{C}})$ and $S^2(R_{\mathcal{T}}) < S^2(Y_{\mathcal{T}})$. In other words, we wish for $\hat{y}^r(\boldsymbol{x})$ to capture at least some of the variation in Y, so that R is less variable than Y. This will be achieved in practice when $\hat{y}^r(\cdot)$ does indeed successfully predict outcomes in the RCT.

However, when $\hat{y}^r(\cdot)$ performs poorly in the experimental sample, rebar can actually *increase* variance. This will be the case if an algorithm trained in the remnant extrapolates poorly to the experimental sample—for instance, if the distribution of $\boldsymbol{x}$ or the distribution of Y differs substantially between the remnant and the RCT. We will see examples of rebar harming precision, sometimes substantially, in Section 4 when we analyze the ASSISTments trials.

That said, rebar is at least somewhat robust to systematic differences between the remnant and the RCT. Note, importantly, that if d is some fixed constant, then replacing $\hat{y}^r(\cdot)$ by $\tilde{y}^r(\cdot) \equiv \hat{y}^r(\cdot)+d$ does not have any impact on the rebar estimator. Both $\bar{R}_{\mathcal{C}}$ and $\bar{R}_{\mathcal{T}}$ will be decreased by d, and thus $\hat{\tau}^{\text{Rebar}}$ will be unchanged. Moreover, $S^2(R_{\mathcal{C}})$ and $S^2(R_{\mathcal{T}})$ will also both be unchanged. Thus, if outcomes in the RCT are systematically shifted from those in the remnant by a fixed constant amount, rebar can still perform well. What matters is that $S^2(R_{\mathcal{C}}) < S^2(Y_{\mathcal{C}})$ and $S^2(R_{\mathcal{T}}) < S^2(Y_{\mathcal{T}})$. It is *not* necessary that $\bar{R}_{\mathcal{C}} \approx 0$ or $\bar{R}_{\mathcal{T}} \approx 0$. Note that this property of rebar also suggests that we need not be overly concerned with whether we regard $\hat{y}^r(\cdot)$ as estimating y^c, y^t, or $\mathbb{E}Y$. If we suppose there is a constant treatment effect, then rebar would perform equally well if $\hat{y}^r(\cdot)$ is estimating y^c, y^t, or $\mathbb{E}Y$, as these would all simply be shifted versions of one another.

To summarize, the remnant is often much larger than the experimental sample, and may provide fertile ground for fitting outcome models, especially in the presence of rich high-dimensional covariates. Once a reasonably good outcome model has been obtained, rebar can be used to potentially improve precision. However, despite the "robustness" properties mentioned above, $\hat{y}^r(\cdot)$ must extrapolate reasonably well to the RCT, implying that the experimental sample cannot differ too substantially from the remnant. When $\hat{y}^r(\cdot)$ extrapolates poorly to the RCT, it is possible that rebar can actually harm precision. To make matters worse, the performance of $\hat{y}^r(\cdot)$ in the experimental sample—where it counts—may not be checked directly to select a best model, since outcomes from the RCT can not be touched

when fitting $\hat{y}^r(\cdot)$. Thus, covariate adjustment using rebar is risky.

3.2 ReLOOP: Flexibly Incorporating Remnant-Based Imputations

Our goal is to construct a method that, like rebar, is able to exploit data in the remnant, but, like LOOP, poses little risk of harming precision in practice. The basic idea is to predict experimental outcomes as in rebar, but to then use those predictions as a covariate in LOOP.

To begin, define $x_i^r \equiv \hat{y}^r(\boldsymbol{x}_i)$, i.e., we let x_i^r denote the predicted outcome for subject i using the predictive model $\hat{y}^r(\cdot)$ fit in the remnant. As this notation suggests, x_i^r may be thought of as simply an additional covariate, along with those in $\boldsymbol{x}_i$. In particular, x_i^r is invariant to treatment assignment. We may therefore include x_i^r as a covariate in LOOP. The hope is that x_i^r might be an especially helpful covariate.

As a simple example, we may run LOOP on the RCT data, but using only x_i^r as a covariate, and using linear regression to construct imputations $\hat{y}^c_{-i}(x_i^r)$ and $\hat{y}^c_{-i}(x_i^r)$. That is, for each i we let

$$\hat{y}^c_{-i}(x_i^r) = a^c_{-i} + b^c_{-i} x_i^r \tag{12}$$

where a^c_{-i} and b^c_{-i} are the intercept and slope coefficients, respectively, from a univariate regression of $Y_{\mathcal{C}\setminus i}$ on $x^r_{\mathcal{C}\setminus i}$. The expression for $\hat{y}^t_{-i}(x_i^r)$ would be analogous. We refer to this strategy as ReLOOP, or "remnant-based LOOP."

ReLOOP may be preferable to rebar because, for each observation i, the remaining $N-1$ observations help determine the best use of x_i^r in constructing $\hat{m}_i$. For example, suppose that the x^r are highly accurate imputations of the y^c in the RCT. In this case, we might expect $a^c_{-i} \approx 0$ and $b^c_{-i} \approx 1$ so that $\hat{y}^c_{-i}(x_i^r) \approx x_i^r$, or in other words, the predictions from the remnant would "pass through" the LOOP procedure largely unmodified, resulting in a rebar-like adjustment. However, in contrast to rebar, poor imputations x^r will not necessarily harm precision in ReLOOP. Consider the extreme case in which the x^r are pure noise. We would then expect $a^c_{-i} \approx \bar{Y}_{\mathcal{C}\setminus i}$ and $b^c_{-i} \approx 0$ so that $\hat{y}^c_{-i}(x_i^r) \approx \bar{Y}_{\mathcal{C}\setminus i}$. That is, we would revert approximately to mean-imputation, and the final estimator would therefore approximately equal the simple difference estimator. In other words, the role of x^r may be tempered according to the prediction accuracy of $\hat{y}^r(\cdot)$ in the RCT.

To further illuminate the relationship between ReLOOP, rebar, and the simple difference estimator, we define what we will refer to as the "generalized rebar estimator," of which both rebar and the simple difference estimator are special cases. Let b be some fixed constant. Then the generalized rebar estimator is

$$\hat{\tau}^{\text{GR}}(b) \equiv \frac{1}{n_t}\sum_{i\in\mathcal{T}}(Y_i - bx_i^r) - \frac{1}{n_c}\sum_{i\in\mathcal{C}}(Y_i - bx_i^r). \tag{13}$$

Note that when $b = 0$ this is equivalent to the simple difference estimator, and when $b = 1$ this is equivalent to the rebar estimator. To now relate this to ReLOOP, consider a modification of the ReLOOP procedure as described above in which the slope coefficient in the regression is constrained to be a fixed value b. Then, as shown in Proposition 1 below, ReLOOP is equivalent to the generalized rebar estimator.

Proposition 1. *Let*

$$\hat{y}^c_{-i}(x^r_i) = a^c_{-i} + b^c_{-i}x^r_i$$
$$\hat{y}^t_{-i}(x^r_i) = a^t_{-i} + b^t_{-i}x^r_i$$

as in (12), but constrain the slope coefficients to be some fixed value b. That is, for some fixed value of b, let

$$\begin{aligned}
b^c_{-i} &= b \\
b^t_{-i} &= b \\
a^c_{-i} &= \arg\min_a \sum_{j \in \mathcal{C} \setminus i} \left[Y_j - \left(a + bx^r_j\right)\right]^2 \\
a^t_{-i} &= \arg\min_a \sum_{j \in \mathcal{T} \setminus i} \left[Y_j - \left(a + bx^r_j\right)\right]^2.
\end{aligned}$$

Then the ReLOOP estimator is identical to $\hat{\tau}^{\mathrm{GR}}(b)$.

Proof. See Appendix A. □

Thus in particular, when the slope coefficients b^c_{-i} and b^t_{-i} are constrained to be 0, ReLOOP is identical to the simple difference-in-means estimator, and when the slope coefficients are constrained to be 1, ReLOOP is identical to the rebar estimator. However, by allowing b^c_{-i} and b^t_{-i} to be estimated from the RCT data, we are typically able to make more effective use of x^r. Indeed, as long as the potential outcomes y^c and y^t as well as the predictions x^r are reasonably well behaved (i.e., no extreme outliers or leverage points), then we might reasonably expect the ReLOOP estimator to nearly always outperform, or at least perform no worse than, rebar and the simple difference estimator. This is formalized in the following proposition:

Proposition 2. *Let* $(y^c_1, y^t_1, x^r_1), \ldots, (y^c_N, y^t_N, x^r_N)$ *be IID samples from a population in which* y^c, y^t, *and* x^r *have finite fourth moments, and where* $-1 < \mathrm{corr}(y^c, x^r) < 1$ *and* $-1 < \mathrm{corr}(y^t, x^r) < 1$. *Let* b *be a fixed constant. Let* $\hat{\mathbb{V}}[\hat{\tau}^{\mathrm{GR}}(b)]$ *denote the estimated variance of* $\hat{\tau}^{\mathrm{GR}}(b)$, *defined analogously to* (3) *and* (11). *Let* $\hat{\mathbb{V}}(\hat{\tau}^{\mathrm{ReLOOP}})$ *denote the estimated variance of ReLOOP, defined as in* (8), *with potential outcomes imputed as in* (12). *Then as* $N \to \infty$,

$$\frac{\hat{\mathbb{V}}(\hat{\tau}^{\mathrm{ReLOOP}})}{\hat{\mathbb{V}}[\hat{\tau}^{\mathrm{GR}}(b)]} \xrightarrow{p} \phi(b) \leq 1$$

where $\phi(b)$ *is some constant that depends on* b.

Proof. See Appendix A. □

Importantly, because the x^r are used as a covariate within LOOP, they do not necessarily need to accurately impute the potential outcomes in the RCT; rather, it suffices that they are merely predictive of the potential outcomes. If the RCT is systematically different from the

remnant, e.g., the potential outcomes in the RCT differ in scale from those in the remnant, the x^r will still be useful as long as they are correlated with the experimental potential outcomes. Indeed, counterintuitively, it is even possible for ReLOOP to achieve precision gains if the x^r are *anticorrelated* with outcomes in the RCT.

3.3 ReLOOP+: Combining Remnant-Based and Within-RCT Covariate Adjustment

ReLOOP effectively solves rebar's main deficiencies. However, when only the remnant-based predictions x^r are used, as in (12), ReLOOP largely neglects the RCT covariate data (except to the extent that x^r depends on $\boldsymbol{x}$ through $\hat{y}^r(\cdot)$). Neglecting the RCT covariate data may be suboptimal, especially when $\hat{y}^r(\cdot)$ is poorly predictive of outcomes in the RCT, perhaps due to systematic differences between the RCT and the remnant. Our goal in this section is to augment ReLOOP, so that it may also exploit the RCT covariate data.

Define

$$\tilde{\boldsymbol{x}}_i \equiv (x_{i1}, x_{i2}, ..., x_{ip}, x_i^r) \tag{14}$$

or in other words, $\tilde{\boldsymbol{x}}_i$ is $\boldsymbol{x}_i$ augmented with x_i^r. We may then run LOOP on the RCT data, using the augmented set of covariates $\tilde{\boldsymbol{x}}$ instead of $\boldsymbol{x}$. We refer to this estimation strategy as "ReLOOP+." The hope is that by including x_i^r we can exploit information in the remnant in much the same way that rebar and ReLOOP do, while also performing within-sample covariate adjustment, as in LOOP.

The imputation strategy within LOOP may be any learning algorithm that can predict y^c and y^t as a function of $\tilde{\boldsymbol{x}}_i$. Wu and Gagnon-Bartsch [2018] recommends using random forests; we will refer to ReLOOP+ with random forest imputations as "ReLOOP+RF."

The precision of the ReLOOP+ estimator will depend on the performance of the imputation strategy, and in particular, its ability to integrate information from the remnant, via x^r, with information from other covariates $\boldsymbol{x}$. On the one hand, x^r is a function of the other covariates and thus, in at least some sense, does not contain any additional information. However, the function $\hat{y}^r(\cdot)$ is fitted on the remnant, which may be much larger than the experimental sample, and thus $\hat{y}^r(\cdot)$ may be a more accurate imputation function than what we would be able to obtain using the RCT data alone. In this sense, x_i^r does contain additional information, which ReLOOP+ can exploit by heavily weighting x^r over the other covariates.

On the other hand, if the x^r are highly accurate, using them as a covariate within a nonparametric method like a random forest may be statistically inefficient relative to OLS. Therefore, it may not always be clear whether ReLOOP or ReLOOP+RF will perform better; it depends on the quality of the imputations x^r as well as the predictive power of the covariates in the experimental sample.

This suggests imputing potential outcomes within LOOP using a specialized ensemble learner [e.g. Opitz and Maclin, 1999]: a weighted average of OLS using just x_i^r, as in ReLOOP, and random forests using $\tilde{\boldsymbol{x}}$, as in ReLOOP+RF. In particular, let $\hat{y}_{-i}^{c,\text{LS}}(\tilde{\boldsymbol{x}}_i)$ be the least squares imputation defined in (12), i.e., the imputation used in ReLOOP; note in particular

that $\hat{y}_{-i}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_i)$ ignores all of the entries of $\tilde{\boldsymbol{x}}_i$ except x_i^r. Let $\hat{y}_{-i}^{c,\mathrm{RF}}(\tilde{\boldsymbol{x}}_i)$ denote the imputation from a random forest regression of $Y_{\mathcal{C}\setminus i}$ on $\tilde{\boldsymbol{x}}_{\mathcal{C}\setminus i}$. We then define an ensemble imputation

$$\hat{y}_{-i}^{c,\mathrm{EN}}(\tilde{\boldsymbol{x}}_i) = \gamma_i^c \hat{y}_{-i}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_i) + (1-\gamma_i^c)\hat{y}_{-i}^{c,\mathrm{RF}}(\tilde{\boldsymbol{x}}_i) \tag{15}$$

which is an interpolation between $\hat{y}_{-i}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_i)$ and $\hat{y}_{-i}^{c,\mathrm{RF}}(\tilde{\boldsymbol{x}}_i)$, where the interpolation parameter γ_i^c is such that $0 \leq \gamma_i^c \leq 1$ and is given by

$$\gamma_i^c = \underset{\gamma\in[0,1]}{\arg\min} \sum_{j\in\mathcal{C}\setminus i} \left[Y_j - \left(\gamma \hat{y}_{-i,j}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_j) + (1-\gamma)\hat{y}_{-i,j}^{c,\mathrm{RF}}(\tilde{\boldsymbol{x}}_j) \right) \right]^2$$

where $\hat{y}_{-i,j}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_j)$ is defined analogously to $\hat{y}_{-i}^{c,\mathrm{LS}}(\tilde{\boldsymbol{x}}_i)$, but with both observations i and j removed, and similarly for $\hat{y}_{-i,j}^{c,\mathrm{RF}}(\tilde{\boldsymbol{x}}_j)$. That is, the interpolation parameter γ_i^c is obtained empirically to minimize mean squared error, and is obtained from a leave-one-out procedure, which ensures that $\gamma_i^c \perp\!\!\!\perp T_i$, and thus $\hat{y}_{-i}^{c,\mathrm{EN}}(\tilde{\boldsymbol{x}}_i) \perp\!\!\!\perp T_i$. In other words, ReLOOP+ with potential outcomes imputed as (15) uses a leave-one-out procedure both to fit imputation models and to choose between them. We refer to this ensemble-based method as "ReLOOP+EN."

The imputation strategy (15) allows ReLOOP+EN to triangulate between ReLOOP and ReLOOP+RF, at the cost of estimating only one additional parameter (i.e., γ_i^c). ReLOOP+EN therefore combines the advantages of rebar and LOOP; see the simulation studies in Appendix B. Like rebar, ReLOOP+EN makes use of the remnant, and when the resulting imputations are sufficiently accurate, ReLOOP+EN returns approximately the same estimate as rebar. Like LOOP, ReLOOP+EN uses modern machine learning methods to adjust for covariates within the sample. Like both rebar and LOOP, ReLOOP+EN is design-based, and hence returns unbiased estimates and conservative inferences regardless of the joint distribution of potential outcomes and covariates, and without modeling assumptions beyond the experimental design itself.

4 Estimating Effects in 22 Online Experiments

4.1 Data from the ASSISTments TestBed

We apply and evaluate the methods described in this work to a set of 22 randomized controlled experiments run within the ASSISTments Testbed. As noted in the Introduction, ASSISTments is a free web-based learning platform used by real teachers and students for classwork and homework; the system contains primarily 6th, 7th, and 8th grade mathematics content (which is the basis of the 22 studies observed in this work), but also hosts content and users from other domains and grade levels ranging from early elementary school through college-level. The Testbed is a platform on which external researchers can propose and run studies within ASSISTments, randomizing student and teacher users in real-time into different software configurations. The hope is that by A/B testing, researchers can rigorously test educational or cognitive theories, while simultaneously guiding the improvement of ASSISTments' pedagogy.

Once a TestBed proposal is approved, based on Institutional Review Board and content quality criteria, its experimental conditions are embedded into an ASSISTments assignment. This is then assigned to students, either by a group of teachers recruited by the researcher or, more commonly, by the existing population of teachers using ASSISTments in their classrooms. As an example, consider an experiment comparing text-based hints to video hints. The proposing researcher would create the alternative hints and embed them into particular assignable content, otherwise referred to as a "problem set." Then, any time that a teacher assigns that problem set to his or her students, those students are randomized to one of the conditions, and, when they request hints, receive them as either text or video.

There are several types of problem sets that researchers can utilize when developing their experiments. In the case of the 22 experiments observed in this work, the problem sets are mastery-based assignments called skill builders. As opposed to more traditional assignments requiring students to complete all problems assigned, skill builders require students to demonstrate a sufficient level of understanding in order to complete the assignment. By default, students must simply answer three consecutive problems correctly without the use of computer-provided aid such as hints or scaffolding (a type of aid that breaks the problem into smaller steps). In this way, completion acts as a measure of knowledge and understanding as well as persistence and learning, as students will be continuously given more problems until they are able to reach the completion threshold. ASSISTments also includes a daily limit of ten problems to encourage students to seek help if they are struggling to reach the threshold.

After the completion of a TestBed experiment, the proposing researcher may download a dataset which includes students' treatment assignments and their performance within the skill builder, including an indicator for completion. Additionally, the dataset includes 30 aggregated features that describe student performance and activity recorded within the learning platform prior to random assignment for each respective experiment. We combined this data with disaggregated log data from students' individual prior assignments.

4.2 Imputations from the Remnant

We also gathered analogous data from a large remnant of students who did not participate in any of the 22 experiments we analyzed. Ideally, the remnant would consist of previous ASSISTments students who had worked on the skill builders on which the 22 experiments had been run. If that were the case, we would have considered 22 outcomes of interest, say Y_s, denoting completion of skill builder s. Unfortunately, due to labeling conventions in the ASSISTments database, this was not feasible. Instead, we used prior ASSISTments data to impute one outcome, completion of a generic skill builder.

Rather than use the entire set of past ASSISTments users to build a remnant, we selected students who resembled those who participated in the 22 experiments. Specifically, we first observed the collection of problem sets given to students in the experiments before being assigned. The remnant consisted of all other ASSISTments users who had been assigned to at least one of those assignments. In other words, the remnant consisted of students who did not participate in any of the 22 experiments, but had worked on some of the same content

as those who did. In all, the remnant consisted of 130,678 students. Sample sizes in the 22 experiments are given in Table 1.

Experiment	A	B	C	D	E	F	G	H	I	J	K
n_C	956	330	680	943	355	231	367	617	338	193	265
n_T	961	365	650	921	349	197	387	587	289	209	275
Experiment	L	M	N	O	P	Q	R	S	T	U	V
n_C	165	188	199	264	215	281	224	270	201	238	69
n_T	170	193	213	281	211	234	233	253	228	259	67

Table 1: Sample Sizes for each of the 22 TestBed A/B Tests

We gathered records of up to ten assigned skill builders for each student in the remnant, and for each skill builder recorded the number of problems the student started, completed, requested help on, and answered correctly, the total amount of time spent, and assignment completion (i.e., skill mastery). Then, we fit a type of recurrent neural network [e.g. Williams and Zipser, 1989] called Long-Short Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] to the resulting panel data. The model attempted to detect within-student trends in assignment completion and speed (i.e. the number of problems needed for skill mastery); please see Appendix C for further details. Using 10-fold cross validation within the remnant, we estimated the area under the ROC curve as 0.82 and a root mean squared error of 0.34 for the dependent measure of next assignment completion.

After fitting and validating the model in the remnant, we used it to predict skill builder completion for each subject in each of the 22 experiments. To do so, we gathered log data for each student from up to ten previous assigned skill builders. (Students in the experiments with no prior data were dropped from all analyses.) Using the model fit in the remnant, we predicted whether each student would complete his or her next assigned skill builder. The resulting predictive probabilities were used as x^r in the following analyses.

4.3 Results

In each of the 22 experiments, we calculated 5 different unbiased ATE estimates:

1. The simple difference
2. Rebar
3. ReLOOP
4. LOOP using only covariates supplied within TestBed
5. ReLOOP+E, using both x^r and provided TestBed covariates

These 5 methods are all design-based and unbiased, but they differ in their adjustment for covariates—both in the data they use for the adjustment, and in how the adjustment is effected. Each estimator has been described above: simple difference in Section 2.1, rebar

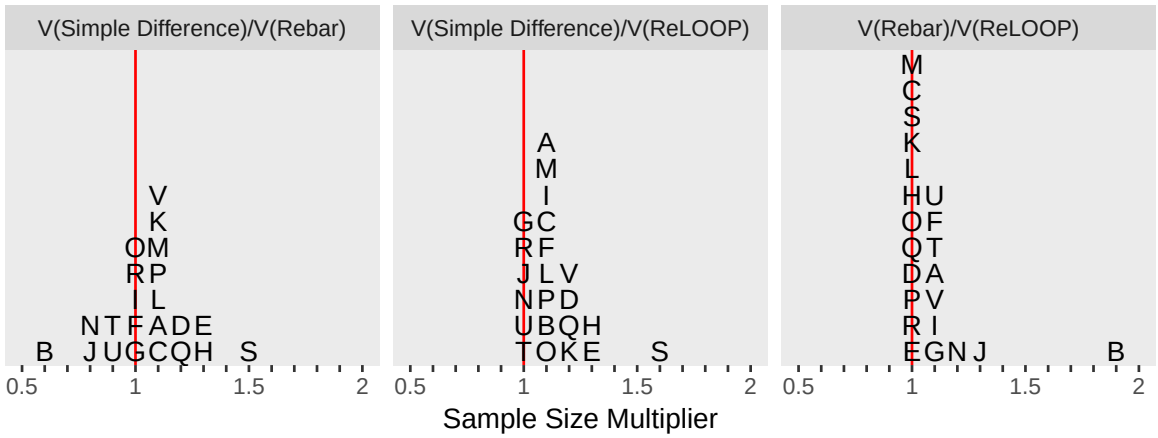

Figure 1: A labeled dotplot showing sample size multipliers (i.e. sampling variance ratios) comparing the simple difference, rebar, and ReLOOP estimators on the 22 ASSISTments TestBed experiments.

in Section 3.1, LOOP (without remnant data) in Section 2.3, ReLOOP in Section 3.2, and ReLOOP+E in Section 3.3.

Since each of these estimates is unbiased, we will focus on their estimated sampling variances. To aid interpretability, we will express contrasts between the sampling variances of two methods in terms of sample size. The estimated sampling variance of each estimator we consider is inversely proportional to sample size. Therefore, reducing the sampling variance of an estimator by, say, 1/2 is equivalent to doubling its sample size. Under that reasoning, the following discussion will refer to the ratio of estimated sampling variances as a "sample size multiplier."

4.3.1 Remnant-Based Adjustment: Simple Difference, Rebar, and ReLOOP

Figure 1 compares the simple difference, rebar, and ReLOOP estimators on the 22 ASSISTments TestBed experiments. Each letter in the figure corresponds to a sample size multiplier comparing two estimated sampling variances in a particular experiment. The labels are arranged in bins of width 0.1, and ordered vertically within bins so experiments with larger sample size multipliers are placed higher. The vertical line at 1.0 indicates experiments in which the two methods gave approximately equal sampling variances (i.e., a ratio between 0.95 and 1.05); labels to the right and left of the line indicate experiments in which the method whose variance is in the denominator or numerator of the fraction is lower.

The leftmost plot contrasts rebar with the simple difference estimator. In five experiments, the rebar and simple difference sampling variances were approximately equal, and in 12 experiments rebar outperformed the simple difference estimator. Notably, in one case (labeled experiment "S") rebar covariate adjustment was equivalent to a roughly 50% increase in sample size. On the other hand, in 5 experiments, rebar's sampling variance was higher than that of the simple difference estimator, most egregiously in experiment B, in

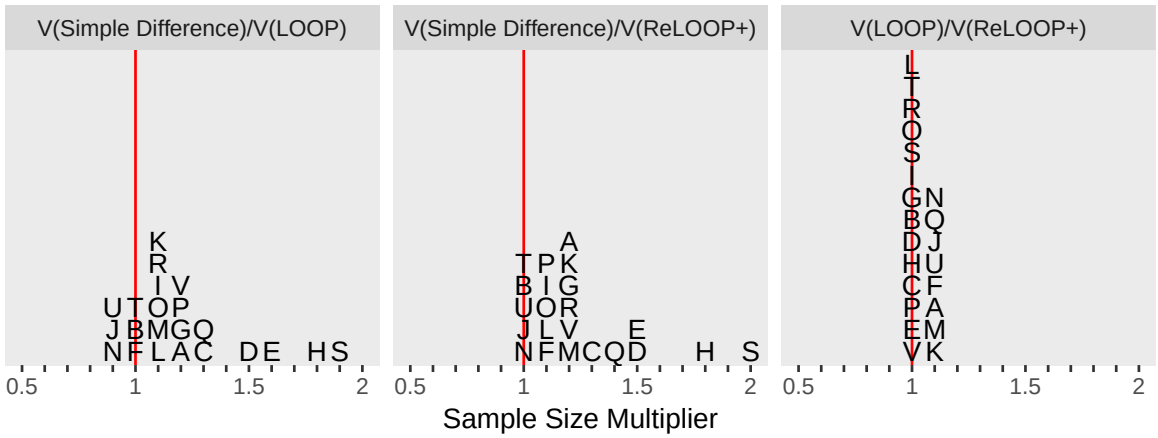

Figure 2: A labeled dotplot showing sample size multipliers (i.e. sampling variance ratios) comparing the simple difference, LOOP (without the remnant), and ReLOOP+ estimators on the 22 ASSISTments TestBed experiments.

which rebar covariate adjustment was equivalent to a roughly 40% *decrease* in sample size. In this case, apparently, the imputations from the model fit to the remnant were particularly inaccurate in the experimental sample. Because experimental outcomes played no role whatsoever in rebar covariate adjustment, the adjustment was blind to this inaccuracy, and was unable to anticipate the resulting increase in standard errors in those cases.

In contrast, the ReLOOP estimator incorporates information on imputation accuracy into its covariate adjustment. The middle panel of Figure 1 shows that across the board, ReLOOP standard errors were smaller or roughly equal to simple difference standard errors. In those cases in which rebar performed well, ReLOOP tended to perform even better. For instance, in experiment S, ReLOOP reduced the simple difference standard errors by a factor of 20%.

The rightmost panel of Figure 1 makes the comparison between rebar and ReLOOP explicit: ReLOOP sample variances dominated those of rebar. In roughly half of the experiments, rebar and ReLOOP performed similarly, and in the remaining half ReLOOP improved upon rebar. Proposition 2, above, guarantees that ReLOOP will dominate both the simple difference and rebar estimators in the limit as $N \to \infty$; Figure 1 gives examples of this property in finite samples.

4.3.2 Incorporating Standard Covariates

Figure 2 compares the simple difference, LOOP (without the remnant, and using random forests), and ReLOOP+ estimators on the 22 ASSISTments TestBed experiments. The left panel shows that in all but three cases, LOOP's sampling variance was less than or roughly equal to the simple difference sampling variance. In those three cases within-sample covariate adjustment via LOOP was equivalent to a roughly 10% decrease in sample size. In contrast, in 16 experiments LOOP improved upon the simple difference estimator. In four of those,

the improvement due to LOOP was equivalent to increasing the sample size by about 50% or more, and in the case of experiment S, by about 90%.

The middle and right panels of Figure 2 compare ReLOOP+EN to simple difference and LOOP, respectively. They show ReLOOP+EN dominating both the simple difference and LOOP estimators. Across all 22 experiments, the estimated sampling variances for the ReLOOP+EN estimates were lower or roughly equal to those of the other two estimates—in these datasets, ReLOOP+EN indeed appears to have incorporated the advantages of both its constituent methods, rebar and LOOP.

5 Discussion

Randomized experiments and observational studies have complementary strengths. Randomized experiments allow for unbiased estimates with minimal statistical assumptions, but often suffer from small sample sizes. Observational studies, by contrast, may offer huge sample sizes, but typically suffer from confounding biases which must be adjusted for, often through statistical modeling with questionable assumptions. In this paper we have attempted to combine the strengths of both. More specifically, we have sought to improve the precision of randomized experiments by exploiting the rich information available in a large observational dataset.

Our approach may be summarized as "first, do no harm." A randomized experiment may be analyzed by taking a simple difference in means, which on its own provides a valid design-based unbiased estimate. The only rationale for a more complicated analysis is to improve precision. Our goal has therefore been to ensure that, in attempting to improve precision by incorporating observational data, we have not actually made matters worse. In particular, we have sought to ensure that (1) no biases in the observational data may "leak" into the analysis, (2) we can reasonably expect to improve precsion, and not harm it, and (3) inference may be justified by the experimental randomization, without the need for additional statistical modeling assumptions.

References

Peter M Aronow and Joel A Middleton. A class of unbiased estimators of the average treatment effect in randomized experiments. *Journal of Causal Inference*, 1(1):135–154, 2013.

Peter M. Aronow, Donald P. Green, and Donald K. K. Lee. Sharp bounds on the variance in randomized experiments. *Ann. Statist.*, 42(3):850–871, 2014.

Heejung Bang and James M Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

Alexandre Belloni, Victor Chernozhukov, and Christian Hansen. Inference on treatment

effects after selection among high-dimensional controls. *The Review of Economic Studies*, 81(2):608–650, 2014.

Adam Bloniarz, Hanzhong Liu, Cun-Hui Zhang, Jasjeet S Sekhon, and Bin Yu. Lasso adjustments of treatment effect estimates in randomized experiments. *Proceedings of the National Academy of Sciences*, 113(27):7383–7390, 2016.

Anthony Botelho, Adam C Sales, Neil T Heffernan, and Thanaporn March Patikorn. The assistments testbed: Opportunities and challenges of online experimentation in intelligent tutors, 2018.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018.

Alexis Diamond and Jasjeet S Sekhon. Genetic matching for estimating causal effects: A general multivariate matching method for achieving balance in observational studies. *Review of Economics and Statistics*, 95(3):932–945, 2013.

Wouter Duivesteijn, Tara Farzami, Thijs Putman, Evertjan Peer, Hilde JP Weerts, Jasper N Adegeest, Gerson Foks, and Mykola Pechenizkiy. Have it both waysfrom a/b testing to a&b testing with exceptional model mining. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 114–126. Springer, 2017.

David A. Freedman. On regression adjustments to experimental data. *Advances in Applied Mathematics*, 40(2):180–193, 2008.

Emily R Fyfe. Providing feedback on computer-based algebra homework in middle-school classrooms. *Computers in Human Behavior*, 63:568–574, 2016.

Erin Hartman, Richard Grieve, Roland Ramsahai, and Jasjeet S Sekhon. From sample average treatment effect to population average treatment effect on the treated: combining experimental with observational studies to estimate population treatment effects. *Journal of the Royal Statistical Society Series A*, 10:1111, 2015.

Neil T Heffernan and Cristina Lindquist Heffernan. The assistments ecosystem: building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Daniel G Horvitz and Donovan J Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260): 663–685, 1952.

Nathan Kallus, Aahlad Manas Puli, and Uri Shalit. Removing hidden confounding by experimental grounding. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 10888–10897. Curran Associates, Inc., 2018.

Kenneth R Koedinger and Elizabeth A McLaughlin. Closing the loop with quantitative cognitive task analysis. *International Educational Data Mining Society*, 2016.

Ron Kohavi and Stefan Thomke. The surprising power of online experiments. *Harvard Business Review*, 95(5):74–+, 2017.

Sören R Künzel, Bradly C Stadie, Nikita Vemuri, Varsha Ramakrishnan, Jasjeet S Sekhon, and Pieter Abbeel. Transfer learning for estimating causal effects using neural networks. *arXiv preprint arXiv:1808.07804*, 2018.

Jessica Lim, Rosalind Walley, Jiacheng Yuan, Jeen Liu, Abhishek Dabral, Nicky Best, Andrew Grieve, Lisa Hampson, Josephine Wolfram, Phil Woodward, Florence Yong, Xiang Zhang, and Ed Bowen. Minimizing patient burden through the use of historical subject-level data in innovative confirmatory clinical trials: review of methods and opportunities. *Therapeutic innovation & regulatory science*, 52(5):546–559, 2018.

Patrick McGuire, Shihfen Tu, Mary Ellin Logue, Craig A Mason, and Korinn Ostrow. Counterintuitive effects of online feedback in middle school math: Results from a randomized controlled trial in ASSISTments. *Educational Media International*, 54(3):231–244, 2017.

Kelly L Moore and Mark J van der Laan. Covariate adjustment in randomized trials with binary outcomes: targeted maximum likelihood estimation. *Statistics in Medicine*, 28(1): 39–64, 2009.

J. Neyman. On the application of probability theory to agricultural experiments. essay on principles. section 9. *Statistical Science*, 5:463–480, 1923. 1990; transl. by D.M. Dabrowska and T.P. Speed.

David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198, 1999.

Korinn S Ostrow, Doug Selent, Yan Wang, Eric G Van Inwegen, Neil T Heffernan, and Joseph Jay Williams. The assessment of learning infrastructure (ali): the theory, practice, and scalability of automated assessment. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge*, pages 279–288. ACM, 2016.

Stuart J Pocock. The combination of randomized and historical controls in clinical trials. *Journal of chronic diseases*, 29(3):175–188, 1976.

Eric Ries. *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. Crown Books, 2011.

James M Robins. Robust estimation in sequentially ignorable missing data and causal inference models. In *Proceedings of the American Statistical Association*, volume 1999, pages 6–10. Indianapolis, IN, 2000.

James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association*, 89(427):846–866, 1994.

Paul R. Rosenbaum. Covariance adjustment in randomized experiments and observational studies. *Statistical Science*, 17(3):286–327, 2002a.

P.R. Rosenbaum. Covariance adjustment in randomized experiments and observational studies. *Statistical Science*, 17(3), 2002b.

Evan Rosenman, Art B Owen, Michael Baiocchi, and Hailey Banack. Propensity score methods for merging observational and experimental datasets. *arXiv preprint arXiv:1804.07863*, 2018.

D.B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology; Journal of Educational Psychology*, 66(5):688, 1974.

Piotr Rzepakowski and Szymon Jaroszewicz. Decision trees for uplift modeling with single and multiple treatments. *Knowledge and Information Systems*, 32(2):303–327, 2012.

Adam C Sales, Anthony Botelho, Thanaporn M Patikorn, and Neil T Heffernan. Using big data to sharpen design-based inference in a/b tests. In *Proceedings of the 11th International Conference on Educational Data Mining. International Educational Data Mining Society*, pages 479–486, 2018a.

Adam C Sales, Ben B Hansen, and Brian Rowan. Rebar: Reinforcing a matching estimator with predictions from high-dimensional covariates. *Journal of Educational and Behavioral Statistics*, 43(1):3–31, 2018b.

Daniel O. Scharfstein, Andrea Rotnitzky, and James M. Robins. Rejoinder. *Journal of the American Statistical Association*, 94(448):1135–1146, 1999.

Peter Z Schochet. Statistical theory for the RCT-YES software: Design-based causal inference for RCTs. NCEE 2015-4011. *National Center for Education Evaluation and Regional Assistance*, 2015.

Douglas Selent, Thanaporn Patikorn, and Neil Heffernan. Assistments dataset from multiple randomized controlled experiments. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pages 181–184. ACM, 2016.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

Anastasios A Tsiatis, Marie Davidian, Min Zhang, and Xiaomin Lu. Covariate adjustment for two-sample treatment comparisons in randomized clinical trials: a principled yet flexible approach. *Statistics in medicine*, 27(23):4658–4677, 2008.

Mark J Van der Laan and Sherri Rose. *Targeted learning: causal inference for observational and experimental data*. Springer Science & Business Media, 2011.

Mark J van der Laan and Daniel Rubin. Targeted maximum likelihood learning. *The International Journal of Biostatistics*, 2(1), 2006.

Kert Viele, Scott Berry, Beat Neuenschwander, Billy Amzal, Fang Chen, Nathan Enas, Brian Hobbs, Joseph G Ibrahim, Nelson Kinnersley, Stacy Lindborg, et al. Use of historical control data for assessing treatment effects in clinical trials. *Pharmaceutical statistics*, 13 (1):41–54, 2014.

Stefan Wager, Wenfei Du, Jonathan Taylor, and Robert J Tibshirani. High-dimensional regression adjustments in randomized experiments. *Proceedings of the National Academy of Sciences*, 113(45):12673–12678, 2016.

Candace Walkington, Virginia Clinton, and Anthony Sparks. The effect of language modification of mathematics story problems on problem-solving in online homework. *Instructional Science*, pages 1–31, 2019.

Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

Edward Wu and Johann A. Gagnon-Bartsch. The LOOP estimator: Adjusting for covariates in randomized experiments. *Evaluation Review*, 42(4):458–488, 2018.

Edward Wu and Johann A Gagnon-Bartsch. The p-loop estimator: Covariate adjustment for paired experiments. *arXiv preprint arXiv:1905.08450*, 2019.

Jiacheng Yuan, Jeen Liu, Ray Zhu, Ying Lu, and Ulo Palm. Design of randomized controlled confirmatory trials using historical control data to augment sample size for concurrent controls. *Journal of Biopharmaceutical Statistics*, 29(3):558–573, 2019.

Siyuan Zhao and Neil Heffernan. Estimating individual treatment effect from educational studies with residual counterfactual networks. In *Proceedings of the 10th International Conference on Educational Data Mining, EDM*, 2017.

A Propositions

Proposition 1. *Let*

$$\hat{y}^c_{-i}(x^r_i) = a^c_{-i} + b^c_{-i}x^r_i$$
$$\hat{y}^t_{-i}(x^r_i) = a^t_{-i} + b^t_{-i}x^r_i$$

as in (12), but constrain the slope coefficients to be some fixed value b. That is, for some fixed value of b, let

$$\begin{aligned}
b^c_{-i} &= b \\
b^t_{-i} &= b \\
a^c_{-i} &= \arg\min_a \sum_{j\in\mathcal{C}\setminus i} \left[Y_j - \left(a + bx^r_j\right)\right]^2 \\
a^t_{-i} &= \arg\min_a \sum_{j\in\mathcal{T}\setminus i} \left[Y_j - \left(a + bx^r_j\right)\right]^2.
\end{aligned}$$

Then the ReLOOP estimator is identical to $\hat{\tau}^{\mathrm{GR}}(b)$.

Proof. First note that

$$\begin{aligned}
a^c_{-i} &= \arg\min_a \sum_{j\in\mathcal{C}\setminus i} \left[\left(Y_j - bx^r_j\right) - a\right]^2 \\
&= \frac{1}{n_c - (T_i - 1)} \sum_{j\in\mathcal{C}\setminus i} \left(Y_j - bx^r_j\right)
\end{aligned}$$

and similarly

$$a^t_{-i} = \frac{1}{n_t - T_i} \sum_{j\in\mathcal{T}\setminus i} \left(Y_j - bx^r_j\right)$$

and therefore

$$\hat{m}_i = \begin{cases} \frac{(1-p)\sum_{j\in\mathcal{T}}\left(Y_j - bx^r_j\right)}{n_t} + \frac{p\sum_{j\in\mathcal{C}\setminus i}\left(Y_j - bx^r_j\right)}{n_c - 1} + bx^r_i & T_i = 0 \\ \frac{(1-p)\sum_{j\in\mathcal{T}\setminus i}\left(Y_j - bx^r_j\right)}{n_t - 1} + \frac{p\sum_{j\in\mathcal{C}}\left(Y_j - bx^r_j\right)}{n_c} + bx^r_i & T_i = 1. \end{cases}$$

Thus, in this case the ReLOOP estimator is given by

$$
\begin{aligned}
\hat{\tau} &= \frac{1}{N}\left[\sum_{i\in\mathcal{T}}\frac{1}{p}\left(Y_i-\hat{m}_i\right)+\sum_{i\in\mathcal{C}}\frac{-1}{1-p}\left(Y_i-\hat{m}_i\right)\right] \\
&= \frac{1}{N}\left\{\sum_{i\in\mathcal{T}}\frac{1}{p}\left[Y_i-\left(\frac{(1-p)\sum_{j\in\mathcal{T}\setminus i}\left(Y_j-bx_j^r\right)}{n_t-1}+\frac{p\sum_{j\in\mathcal{C}}\left(Y_j-bx_j^r\right)}{n_c}+bx_i^r\right)\right]+\right. \\
&\qquad \left.\sum_{i\in\mathcal{C}}\frac{-1}{1-p}\left[Y_i-\left(\frac{(1-p)\sum_{j\in\mathcal{T}}\left(Y_j-bx_j^r\right)}{n_t}+\frac{p\sum_{j\in\mathcal{C}\setminus i}\left(Y_j-bx_j^r\right)}{n_c-1}+bx_i^r\right)\right]\right\} \\
&= \frac{1}{N}\left\{\sum_{i\in\mathcal{T}}\frac{1}{p}\left[\left(Y_i-bx_i^r\right)-\left(\frac{(1-p)\sum_{j\in\mathcal{T}\setminus i}\left(Y_j-bx_j^r\right)}{n_t-1}+\frac{p\sum_{j\in\mathcal{C}}\left(Y_j-bx_j^r\right)}{n_c}\right)\right]+\right. \\
&\qquad \left.\sum_{i\in\mathcal{C}}\frac{-1}{1-p}\left[\left(Y_i-bx_i^r\right)-\left(\frac{(1-p)\sum_{j\in\mathcal{T}}\left(Y_j-bx_j^r\right)}{n_t}+\frac{p\sum_{j\in\mathcal{C}\setminus i}\left(Y_j-bx_j^r\right)}{n_c-1}\right)\right]\right\}.
\end{aligned}
$$

If we now define $Z_i = Y_i - bx_i^r$, then

$$
\begin{aligned}
\hat{\tau} &= \frac{1}{N}\left\{\sum_{i\in\mathcal{T}}\frac{1}{p}\left[Z_i-\left(\frac{(1-p)\sum_{j\in\mathcal{T}\setminus i}Z_j}{n_t-1}+\frac{p\sum_{j\in\mathcal{C}}Z_j}{n_c}\right)\right]+\right. \\
&\qquad \left.\sum_{i\in\mathcal{C}}\frac{-1}{1-p}\left[Z_i-\left(\frac{(1-p)\sum_{j\in\mathcal{T}}Z_j}{n_t}+\frac{p\sum_{j\in\mathcal{C}\setminus i}Z_j}{n_c-1}\right)\right]\right\} \\
&= \frac{1}{N}\left[\sum_{i\in\mathcal{T}}\left(\frac{Z_i}{p}-\frac{1-p}{p}\frac{\sum_{j\in\mathcal{T}\setminus i}Z_j}{n_t-1}-\frac{\sum_{j\in\mathcal{C}}Z_j}{n_c}\right)+\right. \\
&\qquad \left.\sum_{i\in\mathcal{C}}\left(-\frac{Z_i}{1-p}+\frac{\sum_{j\in\mathcal{T}}Z_j}{n_t}+\frac{p}{1-p}\frac{\sum_{j\in\mathcal{C}\setminus i}Z_j}{n_c-1}\right)\right] \\
&= \frac{1}{N}\left[\sum_{i\in\mathcal{T}}\frac{Z_i}{p}-\sum_{i\in\mathcal{C}}\frac{Z_i}{1-p}-\frac{1-p}{p}\frac{(n_t-1)\sum_{j\in\mathcal{T}}Z_j}{n_t-1}-\frac{n_t\sum_{j\in\mathcal{C}}Z_j}{n_c}+\right. \\
&\qquad \left.\frac{n_c\sum_{j\in\mathcal{T}}Z_j}{n_t}+\frac{p}{1-p}\frac{(n_c-1)\sum_{j\in\mathcal{C}}Z_j}{n_c-1}\right] \\
&= \frac{1}{N}\left[\sum_{i\in\mathcal{T}}\frac{Z_i-(1-p)Z_i}{p}-\sum_{i\in\mathcal{C}}\frac{Z_i-pZ_i}{1-p}-\frac{n_t\sum_{j\in\mathcal{C}}Z_j}{n_c}+\frac{n_c\sum_{j\in\mathcal{T}}Z_j}{n_t}\right] \\
&= \frac{1}{N}\left[\sum_{i\in\mathcal{T}}Z_i-\sum_{i\in\mathcal{C}}Z_i-\frac{n_t\sum_{j\in\mathcal{C}}Z_j}{n_c}+\frac{n_c\sum_{j\in\mathcal{T}}Z_j}{n_t}\right] \\
&= \frac{1}{N}\left[\frac{(n_c+n_t)\sum_{j\in\mathcal{T}}Z_j}{n_t}-\frac{(n_t+n_c)\sum_{j\in\mathcal{C}}Z_j}{n_c}\right] \\
&= \frac{\sum_{j\in\mathcal{T}}Z_j}{n_t}-\frac{\sum_{j\in\mathcal{C}}Z_j}{n_c}
\end{aligned}
$$

which is the same as (13). □

Proposition 2. *Let* $(y_1^c, y_1^t, x_1^r), \ldots, (y_N^c, y_N^t, x_N^r)$ *be IID samples from a population in which* y^c*,* y^t*, and* x^r *have finite fourth moments, and where* $-1 < \mathrm{corr}(y^c, x^r) < 1$ *and* $-1 < \mathrm{corr}(y^t, x^r) < 1$*. Let* b *be a fixed constant. Let* $\hat{\mathbb{V}}[\hat{\tau}^{\mathrm{GR}}(b)]$ *denote the estimated variance of* $\hat{\tau}^{\mathrm{GR}}(b)$*, defined analogously to* (3) *and* (11)*. Let* $\hat{\mathbb{V}}(\hat{\tau}^{\mathrm{ReLOOP}})$ *denote the estimated variance of ReLOOP, defined as in* (8)*, with potential outcomes imputed as in* (12)*. Then as* $N \to \infty$*,*

$$\frac{\hat{\mathbb{V}}(\hat{\tau}^{\mathrm{ReLOOP}})}{\hat{\mathbb{V}}[\hat{\tau}^{\mathrm{GR}}(b)]} \xrightarrow{p} \phi(b) \leq 1$$

where $\phi(b)$ *is some constant that depends on* b*.*

Proof. To do. □

B Simulations

Note: The notation in this section needs updating, and is not always consistent with the notation in the main text.

We examine the performance of ReLOOP and ReLOOP+ using simulations. In particular, we investigate the effects of varying sample size, the predictive power of the covariates, and the predictive power of the external predictions x^r. We generate our data using a model that is parameterized in such a way that we are able to independently vary these three quantities (sample size, the predictive power of the covariates, and the the predictive power of the external predictions).

We simulate a randomized experiment in which there are N subjects. For each subject i there are two covariates, $Z_{i,1}$ and $Z_{i,2}$, which are independent and Unif$(0, 10)$. The potential outcomes are generated from the following linear model:

$$\begin{aligned} a_i &= 2Z_{i,1} + Z_{i,2} + \delta_i \\ y_i^c &= \frac{a_i}{\sigma_a} \\ y_i^t &= y_i^c + 3 \end{aligned}$$

where $\delta_i \sim \mathrm{N}(0, \sigma^2)$ and $\sigma_a^2 \equiv \mathrm{Var}(a_i) = \frac{500}{12} + \sigma^2$. By generating our potential outcomes as above, we have defined our generative model so that the control potential outcomes have unit variance. We can alternatively write the observed outcome as:

$$Y_i = 3T_i + \frac{2}{\sigma_a} Z_{i,1} + \frac{1}{\sigma_a} Z_{i,2} + \epsilon_i$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2/\sigma_a^2)$.

For each observation, we also simulate external predictions $\tilde{t}_i$ and $\tilde{c}_i$ for y_i^t and y_i^c by taking the true y_i^t or y_i^c and adding a normally distributed noise term with mean 0 and variance σ_{ext}^2.

Again, our goal is to investigate variations in sample size, the predictive power of the covariates, and the predictive power of the external predictions. Sample size is directly indexed by N. We can index the predictive power of the covariates with

$$R_{int}^2 = 1 - \frac{\sigma^2}{\sigma_a^2}.$$

The subscript "int" is for "internal" covariates, i.e. not from the remnant. Similarly, the predictive power of our external prediction $\tilde{c}_i$ is

$$R_{ext}^2 = 1 - \frac{\sigma_{ext}^2}{\mathrm{Var}(\tilde{c}_i)} = 1 - \frac{\sigma_{ext}^2}{1 + \sigma_{ext}^2}.$$

Thus, given a desired R^2_{int} and R^2_{ext}, the corresponding values of σ^2 and σ^2_{ext} are:

$$\sigma^2 = \frac{1 - R^2_{int}}{R^2_{int}} \times \frac{500}{12}$$
$$\sigma^2_{ext} = \frac{1 - R^2_{ext}}{R^2_{ext}}.$$

We perform three sets of simulations. In each, we vary one of the quantities N, R^2_{int}, or R^2_{ext} while holding the other two fixed. For each set of simulations, we compare the following methods:

1. Simple difference estimator
2. LOOP: Uses the LOOP estimator including only the covariates Z_1 and Z_2. Uses a random forest as the imputation method.
3. ReLOOP* ("ReLOOP" in the main text): Uses the LOOP estimator, with OLS as the imputation method. Only includes the external predictions as a covariate.
4. LOOP w External ("LOOP+RF" in the main text): uses the LOOP estimator with external predictions $\tilde{c}$ and $\tilde{t}$ as a covariates in addition to Z_1 and Z_2. Uses a random forest as the imputation method.
5. ReLOOP ("ReLOOP+EN" in the main text): Interpolates between the previous two methods.

We use the following simulation procedure. For a given set of N, R^2_{int}, and R^2_{ext}, we perform $k = 1000$ trials. For each trial, we generate a set of covariates, potential outcomes, a treatment assignment vector, and external predictions as described above. We then produce an estimate of the variance of each method. Next, we average the estimated variance across the k trials. Finally, we plot the average variance for each of the adjustment methods relative to the variance of the simple difference estimator. That is, for each method (2) – (5) we plot (average variance of method) / (average variance of simple difference estimator).

Varying Sample Size For these simulations, we hold the predictive power of the covariates and external predictions constant and vary the sample size. We consider four scenarios: (1) $R^2_{ext} = 0.25, R^2_{int} = 0.25$; (2) $R^2_{ext} = 0.75, R^2_{int} = 0.25$; (3) $R^2_{ext} = 0.25, R^2_{int} = 0.75$; and (4) $R^2_{ext} = 0.75, R^2_{int} = 0.75$. In each scenario the sample sizes considered are $N = 30, 40, 50, 75, 100, 150, 200$. Results are in Figure 3.

We first note that all methods perform better than the simple difference estimator (the relative variances are all less than 1). Also, LOOP with external predictions usually outperforms ordinary LOOP, indicating that using the external predictions is typically beneficial. The exception is when the covariates are highly informative but the external predictions are not ($R^2_{int} = 0.25, R^2_{ext} = 0.75$, lower left panel) and the sample size is small, in which case LOOP with external predictions performs slighly worse than ordinary LOOP.

We also observe that ReLOOP does well at tracking its better performing component (ReLOOP* or LOOP with external predictions). It performs reasonably well at small sample sizes, and quickly converges to near optimal at larger sample sizes. In some cases ReLOOP+ performs better than either component individually.

Varying Predictive Power of External Prediction In these simulations, we hold the predictive power of the covariates and sample size constant and vary R^2_{ext}. We again consider four scenarios, with N fixed at either 30 or 60, and R^2_{int} fixed at either 0.25 or 0.75. The values of R^2_{ext} considered are $R^2_{ext} = 0.05, 0.15, ..., 0.85, 0.95$. Results are in Figure 4.

Once again, we observe that ReLOOP tends to perform at least as well as either of its two components. This is particularly true for $N = 60$, where ReLOOP closely follows (or drops below) the lower of the component lines. As expected, the three methods that incorporate the external predictions all improve as R^2_{ext} increases, while the performance of LOOP stays constant. We see that ReLOOP is outperformed by LOOP only when R^2_{ext} is much lower than R^2_{int}.

Varying Predictive Power of Covariates For this simulation, we hold the predictive power of the external predictions and sample size constant and vary $R^2_{int} = 0.05, 0.15, ..., 0.85, 0.95$. We consider four scenarios: (1) $N = 30, R^2_{ext} = 0.25$; (2) $N = 30, R^2_{ext} = 0.75$; (3) $N = 60, R^2_{ext} = 0.25$; and (4) $N = 60, R^2_{ext} = 0.75$. Results are in Figure 5.

Here the performance of ReLOOP stays constant, as it makes use only of the external predictions, not the covariates. The remaining methods all improve as R^2_{int} increases. As before, we can see that ReLOOP tracks the better performing component well (especially when $N = 60$) and is only outperformed by LOOP when R^2_{int} is much higher than R^2_{ext}.

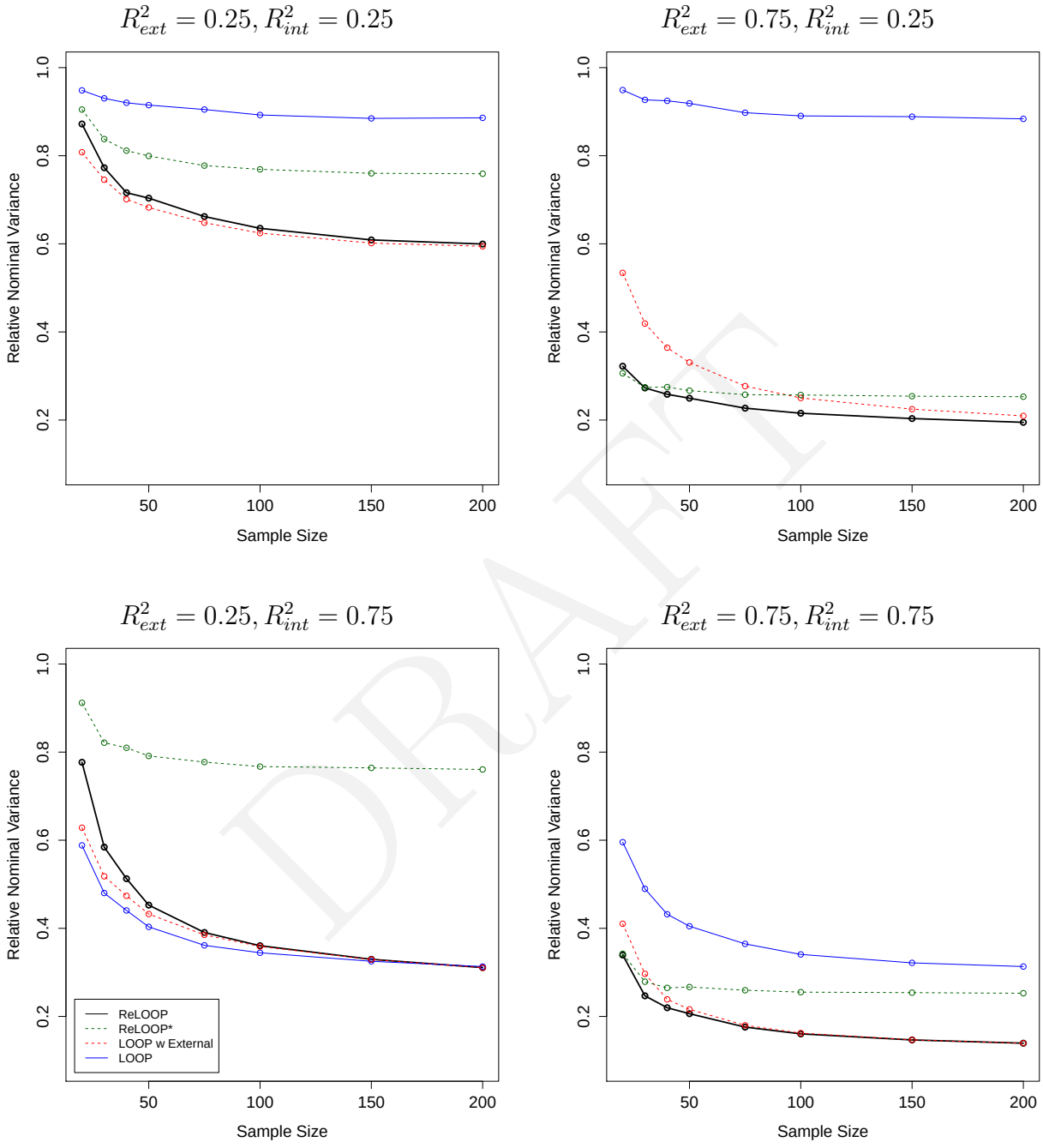

Figure 3: Varying sample size.

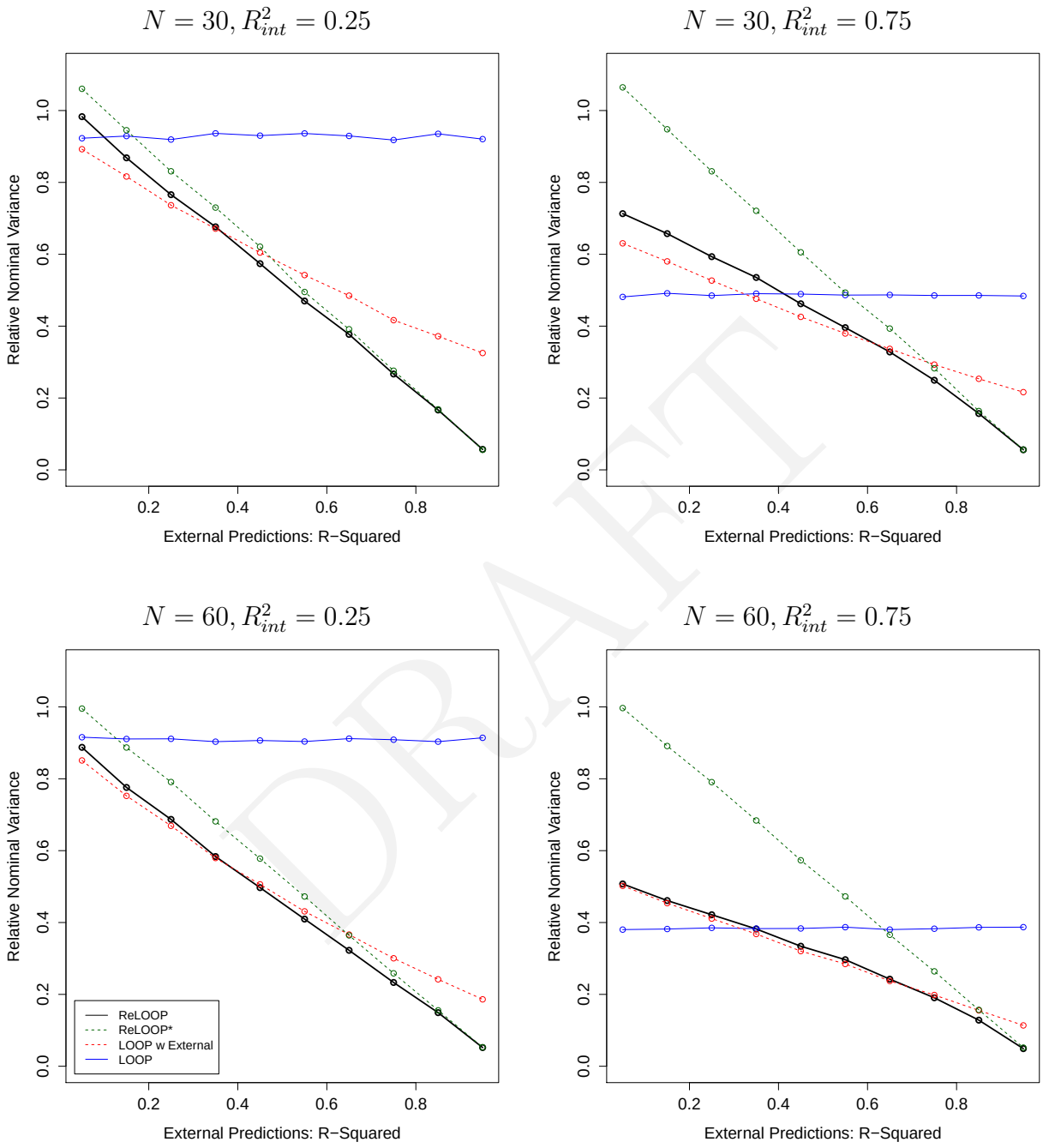

Figure 4: Varying R^2_{ext}

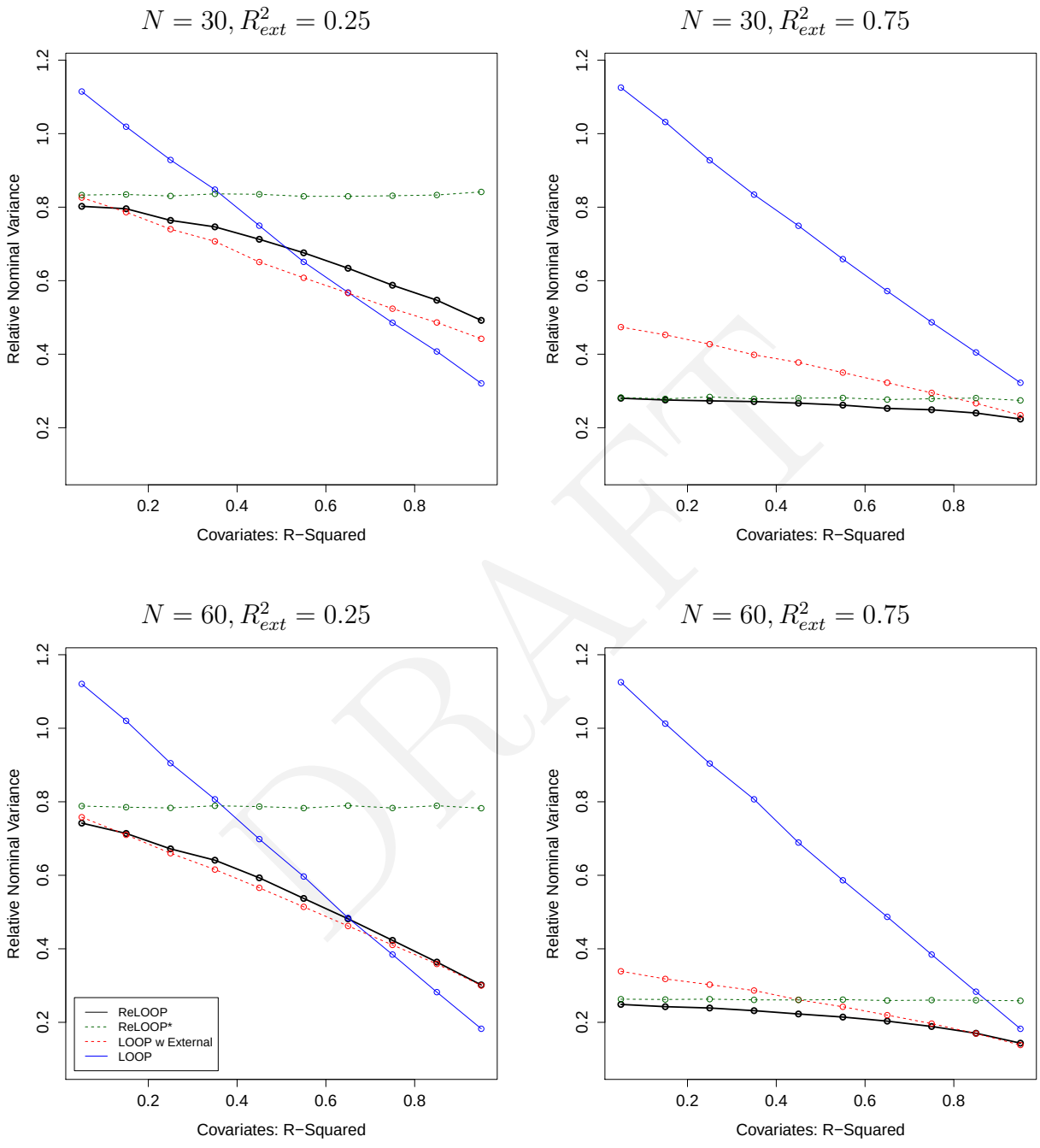

Figure 5: Varying R^2_{int}

C Deep Learning in the Remnant to Impute Completion

Note: This section needs citations and additional details

We used the remnant to train a variant of a recurrent neural network called a Long-Short Term Memory (LSTM) network to predict students' assignment completion. Unfortunately, we were unable to identify a large number of instances in the remnant of students working on the same skill builders as were in the 22 experiments; instead, we trained the LSTM model to predict a student's completion on whatever skill builder he or she worked on next. Specifically, we considered sequences of at most ten worked skill builders within each student's history, and attempted to predict that student's completion on an 11th skill builder.

Deep neural networks are essentially iterated generalized linear models: a set of outcomes are modeled as a function of a linear combination of latent "hidden" variables, which are themselves functions of previous layers of hidden variables. The process iterates until a bottom layer of hidden variables, which is a function of observed covariates. The LSTM model extends this logic to panel data: in each time step, the model combines information from the current observed time step with an aggregation of previous hidden layer outputs as well as an internal "cell memory" to best inform the models outcome estimates.

More precisely, the model is represented as several fully-connected layers, with a set of inputs feeding into one or more hidden layers, and then to an output layer corresponding with the observed dependent measures; this results in an $n \times m$ matrix of weights between layers corresponding to n nodes in a layer and m nodes in the subsequent layer. A nonlinear function is then commonly applied to the output of each layer; we apply a hyperbolic tangent (tanh) function to the output of the LSTM layer and a sigmoid function to the estimates produced by the output layer. We used 16 covariates to describe each single time step (representing a students performance on a single assignment), which then feeds into a hidden LSTM layer of 100 values, or units, which is used to inform an output layer of two units corresponding with two outcomes of interest: completion and inverse mastery speed—a continuous variable that equals the reciprocal of the number of problems a student worked, if they completed the assignment, and zero otherwise. Using the LSTM network to predict two outcomes is an example of multi-task learning, which attempts to reduce model overfitting by simultaneously observing multiple dependent measures, regularizing the model. Completion and inverse mastery speed together represent two different measures of student performance; including both prevents the model from overfitting to any one measure.

During training, the model uses an adaptive gradient descent method called Adam, optimizing model weights to minimize a simple sum of binary cross entropy loss and root mean squared error for the outcomes of completion and inverse mastery speed respectively. The training procedure involves the iterative update of model weights through gradient descent until a stopping criterion is met; in our case performance on a holdout set. Specifically, we used 30% of the training set to estimate the point at which when a 5-epoch moving average of calculated model error on this holdout set either plateaus (i.e. the difference in performance drops below a small threshold) or begins to increase, signifying overfitting; the use of a mov-

ing average helps to prevent the model from stopping too early due to small fluctuations in the difference of model error from one epoch to the next. We specified the LSTM model's hyperparameters based on previously successful model structures and training procedures within the context of education. We evaluate the model using a 10-fold cross validation to gain a measure of model fit (leading to an ROC area under the curve of 0.82 and root mean squared error of 0.34 for the dependent measure of next assignment completion) before then training the model on the full set of remnant data.

We then gave the trained model the sequences of assignment performances of students in the experimental set to gain an estimate of experiment completion for each student across each of the 22 experiments.