Rebar: Reinforcing a Matching Estimator with Predictions from High-Dimensional Covariates

Adam Sales & Ben B Hansen & Brian Rowan

September 14, 2017

1 Introduction: Two Types of Neglected Data

Matching-based observational studies in education sciences often neglect data from the "remnant" of a match: untreated and un-matched subjects. That is, researchers will select a set of matched controls that most closely resemble the treated subjects, and discard data from the remnant, the unmatched controls.

Similarly, due to sample size and other modeling limitations, researchers will typically condition their experimental and observational studies on a small set of pre-treatment covariates that are deemed most relevant to the study—the variables thought most likely to pose a confounding threat. In many cases, reams of less-relevant data are available, perhaps from state longitudinal data systems or from other sources. These less relevant covariates are often discarded.

Conducting a causal analysis using only the matched sample and using only relevant covariates makes good statistical sense. The data from subjects that are not part of a match are likely to be distributed differently than data from the match. The process of matching encourages researchers to focus their analysis on the region of common support; the remnant is typically outside this region by construction. Including irrelevant variables into an analysis can swamp the sample, introduce over-fitting or extreme imprecision, and make impossible common statistical techniques such as ordinary least squares and logistic regression.

But these excluded data—the remnant and ostensibly irrelevant covariates—may also contain valuable information. Perhaps the distribution of the outcome conditional on covariates could be estimated with more precision by vastly increasing the sample size using discarded subjects. Perhaps discarded covariates are not so irrelevant, and capture important baseline differences between treated and untreated subjects.

This paper is an attempt to thread this needle with a new method that we call "remnant-based residualization," or "rebar." The idea of rebar is to, on the one hand, extract as much useful information as possible from the remnant and all available covariates, and on the other hand to preserve the most attractive properties of a good matching design. To implement rebar, we fit a machine learning prediction model to the unmatched controls—the "remnant"—predicting their outcomes in the control condition as a function of the entire set of covariates. Using this fitted model, we then generate predicted outcomes for the matched sample. Finally, instead of calculating the effect of the treatment on participants' outcomes themselves, we estimate the intervention's effect on the difference between participants' predicted outcomes under the control condition, and their actual outcomes, i.e. their prediction residuals—this is "residualization." The predictive model need not be correct in any sense, or consistent or unbiased for any particular parameter. It must only yield predictions that are closer, on average, to control potential outcomes than their mean.

Rebar builds thematically on prior work combining matching with outcome modeling, such as Rubin (1973) and Ho et al. (2007a), among others, alongside "doubly robust" estimation (e.g. Kang and Schafer, 2007). Its most direct antecedents are the papers of Rosenbaum (2002a) and Abadie and Imbens (2012), which suggest forms of residualization for matching estimators, and of Middleton and Aronow (2011), which does the same for weighting estimators. Our contribution to that literature is twofold: first, rebar is *remnant-based*: we argue here that residualization is well suited to recovering otherwise lost information from the remnant. Second, we demonstrate by simulation and example how rebar can exploit machine learning methods and high dimensional covariates without compromising the classical statistical properties of the match.

Rebar can supplement a wide range of matching analyses, and may be used alongside other outcome models and covariate adjustments.

The following section will review causal matching studies, and Section 3 will formally introduce rebar. There, we will discuss a possible threat to the validity of a matching design that rebar can introduce: if the distribution of outcomes, conditional on covariates, differs widely enough between the remnant the matched set, rebar might increase, rather than decrease bias. We will introduce a diagnostic called "proximal validation" that should detect such pathological cases, and suggest ways to tweak the algorithm if a researcher were to confront one.

Rebar can potentially reduce both the bias and the variance of causal estimates, by modeling otherwise unmodeled variation. That said, this paper will focus its attention on rebar's bias reducing properties. We will argue, with analytical results (Section 4), a simulation study (Section 5), and an empirical example (Section 6) that rebar is an effective method for reducing confounding bias from measured, but unmodeled, confounders in a high-dimensional dataset, without compromising the key advantages of matching.

2 Matching in Observational Studies: Review

In an observational study, let $i = 1, ..., n$ index n subjects, and let Z_i denote subject i's binary treatment assignment, and Y_i subject i's observed outcome of interest. Assuming non-interference (Cox, 1958), and following Neyman (1990) and Rubin (1974), let y_{Ti} and y_{Ci}

denote subject i's (perhaps counterfactual) responses were subject i treated and untreated, respectively. Then $Y_i = y_{Ti}Z_i + y_{Ci}(1 - Z_i)$. Further, let x_i be a vector of covariates measured prior to treatment. The potential outcomes y_C and y_T define treatment effects $\tau_i = y_{Ti} - y_{Ci}$ and a causal estimand

$$\tau_{ETT} = \mathbb{E}_Z[\boldsymbol{\tau}^T \boldsymbol{Z}/n_T] = \frac{\boldsymbol{\tau}^T \mathbb{E}\boldsymbol{Z}}{n_T}, \tag{1}$$

the expected average effect of the treatment on the treated. The expectation in (1) is taken conditional on the posited sampling scheme.

In a matching-based observational study, a researcher will create a new categorical variable, $\mathbf{M}$, considering subjects i and j to be *matched* to one another if $M_i = M_j$. (Subjects i with the property that $M_i \neq M_j$ for all $i \neq j$ are *unmatched*.) Researchers will choose M in such a way that matched subjects have similar covariate distributions x. Perhaps the most popular approach to matching is to use propensity scores (Rosenbaum and Rubin, 1983), $Pr(Z = 1|x)$, the probability of being assigned to treatment conditional on covariates x. In a propensity-score matching design, treated and untreated subjects are grouped into matches M with approximately equal estimated propensity scores. Other inexact matching techniques measure subjects' similarity in x using, for example, Mahalanobis distances (Rubin, 1980) or covariate balance tests (Diamond and Sekhon, 2013). Matched sets may contain any (positive) number of treated or untreated subjects (Rosenbaum, 1991).

Ideally, within any matched set, no subject's *a priori* probability of making its way into the treatment group was larger or smaller than any other's:

$$Pr(Z_i = 1|\boldsymbol{M}) = Pr(Z_j = 1|\boldsymbol{M}) \text{ whenever } M_i = M_j; \tag{2}$$

this is *perfect matching*. Under perfect matching in the sense of (2), matched comparisons are statistically equivalent to contrasts of treatment and control conditions in block- or paired randomized designs (e.g., Braitman and Rosenbaum, 2002; Rubin, 2008; Hansen, 2011).

A simple matching-based estimator compares average treated and untreated outcomes within each match. The average difference between treated and untreated subjects in matched set m is

$$t(\boldsymbol{Y}_m, \boldsymbol{Z}_m) = \frac{\boldsymbol{Y}_m^T \boldsymbol{Z}_m}{n_{Tm}} - \frac{\boldsymbol{Y}_m^T (\mathbf{1} - \boldsymbol{Z}_m)}{n_{Cm}}$$

where $\boldsymbol{Y}_m$ and $\boldsymbol{Z}_m$ are vectors of Y and Z, and n_{Tm} and n_{Cm} are the numbers of treated and untreated among subjects $\{i : \boldsymbol{M}_i = m\}$. Then a matching estimator is

$$\hat{\tau}_M(\boldsymbol{Y}) = \sum_m w_m t_m(Y, Z) \tag{3}$$

where weight $w_m = n_{Tm}/n_T$. Estimator $\hat{\tau}_M(\boldsymbol{Y})$ is unbiased for τ_{ETT} under perfect matching (2), or, more generally, if the difference in assignment probabilities is uncorrelated with control potential outcomes (Lemma 1 in the appendix). In practice neither of these will be exactly true, but researchers can hope for approximate unbiasedness, and explore their design's sensitivity to unmeasured (or unmodeled) bias (e.g. Gastwirth et al., 1998; Hosman et al., 2010).

Frequently, subjects who are not sufficiently similar in x to other units are left unmatched. We will refer to the set of unmatched untreated subjects as the *remnant* from a match. Typically, the remnant is discarded. While discarding data might seem unwise, there is good reason to discard the remnant. Since no suitable comparisons may be found between subjects in the remnant and treated subjects, any causal comparisons using the remnant necessarily involve modeling y_C as a function of $\boldsymbol{X}$. Moreover, the remnant typically occupies a mostly separate region of the distribution of $\boldsymbol{X}$ than the matched sample—hence its inability to be matched. Therefore, comparing outcomes from treated subjects with those from the remnant involves extrapolation, which can be highly sensitive to model specification. On the other hand, the remnant may contain information that is useful for modeling y_C.

An extensive, occasionally contentious literature discusses variable selection for propensity score models. This literature begins with Rubin and Thomas, who advised erring on the side of inclusiveness, striving to exclude only those covariates that a consensus of researchers believe to be unrelated to each outcome variable (1996, § 2.3); Rosenbaum's (2002b, p.76) view is similar. Later contributions argued that including variables only weakly related to outcomes may increase the mean squared error (MSE) of effect estimation (Brookhart et al., 2006; Austin, 2011). These additional losses can in principle take the form of bias, not only variance, even if the MSE-increasing variable was determined in advance of treatment assignment (Greenland, 2003; Sjölander, 2009; Pearl, 2009). Most recently, Steiner et al. (2015) argued via case study for including all available covariates, unless "strong substantive theory" (p. 573) suggests the presence of bias-amplifying covariates; ideally, researchers should include covariates from multiple domains, with each domain including as many covariates as possible. Pimentel et al. (2016) suggested conducting two analyses, each matching on a different set of covariates. Methods attempting to limit the MSE penalty by limiting propensity modeling variables to those that correlate with observed outcomes have been met with criticism of a different nature: In Rubin's view, in order to maximize objectivity, during matching researchers should keep outcome measurements in a virtual locked box, only to emerge once the matching structure and other study design elements have been determined (Rubin, 2008).

Rebar, the method of this paper, is compatible with either attitude to selection of propensity score variables; our illustration (§ 6) emphasizes this compatibility by adhering to the more restrictive of the two schools. Without reference to outcome associations, we select for inclusion in the propensity model those variables we felt that a consensus of scholars would be most likely to deem potential confounders. In this example as in many others, the number of potential confounders that could be addressed in this way was limited: when $p \geq n_T$ or $p \geq n_C$, then the treatment and control samples can ordinarily be separated by a hyperplane, in the space spanned by $\boldsymbol{X}$, with the result that common binary regression methods fail to fit (Agresti, 2013; Zorn, 2005); in the example of § 6, $n_T = 7$. This heightens the need for additional measures for confounder control, such as rebar.

3 Rebar: Using an Outcome Model to Reduce Bias in a Matching Design

The procedure we recommend is the following:

1. Using the full dataset, construct a match $\boldsymbol{m}$, perhaps based on a subset of available covariates, thereby dividing the sample into a matched sample and a remnant.
2. Using units in the remnant, construct an algorithm $\hat{y}_C(\cdot)$ to predict y_C as a function of the full matrix $\boldsymbol{X}$.
3. Assess the performance of $\hat{y}_C(\cdot)$ (See Section 3.1)
4. For all subjects i in the matched sample, use $\hat{y}_C(\cdot)$ to predict y_{Ci} as $\hat{y}_{Ci} = \hat{y}_C(x_i)$.
5. Construct prediction errors $e \equiv Y - \hat{y}_C(\boldsymbol{X})$ for all subjects in the matched sample.
6. Estimate treatment effects in the matched sample, substituting e for Y in the outcome analysis.

As in Rosenbaum (2002a), the model $\hat{y}_C(\cdot)$ relating $\boldsymbol{X}$ and y_C is an algorithmic model, rather than a statistical model. That is, it does not estimate parameters of a probability distribution, but rather generates deterministic predictions of y_C when given a vector x. Since this procedure relies on the residuals of a model fit to Y, we will refer to it as "residualization."

The predictions $\hat{y}_C(x)$ bear some similarity to prognostic scores (Hansen, 2008). Prognostic scores, which are analogous to propensity scores, are statistics that are sufficient for the relationship between y_C and x. They are commonly understood as predictions of y_C as a function of x (e.g. Pane et al., 2013). In fact, much of the intuition behind prognostic scores supports our use of $\hat{y}_C(x)$ here, though the prognostic score theory will not play a direct role in our argument.

Now as above, define residuals

$$e = Y - \hat{y}_C(x).$$

Then we may define "potential residuals": $e_C = y_C - \hat{y}_C(x)$ and $e_T = y_T - \hat{y}_C(x)$. Analogously to Y, the observed residuals are $e = Ze_T + (1 - Z)e_C$. Crucially,

$$e_{Ti} - e_{Ci} = \tau_i, \tag{4}$$

where τ_i as above is subject i's treatment effect, $y_{Ti} - y_{Ci}$. To see this, note that $y_C = \hat{y}_C(\boldsymbol{X}) + e_C$ and $y_T = \hat{y}_C(\boldsymbol{X}) + e_T = \hat{y}_C(\boldsymbol{X}) + e_C + \tau$. The prediction $\hat{y}_C(x)$ is based only on pre-treatment variables x, and not on treatment status Z from subjects in the matched sample. That being the case, it cannot be affected by treatment status—we would counterfactually estimate the same $\hat{y}_C(x)$ for alternative realizations of $\boldsymbol{Z}$ in the matched set. Therefore, we can write $e_{Ti} - e_{Ci} = y_{Ti} - \hat{y}_{Ci} - (y_{Ti} - \hat{y}_{Ci}) = y_{Ti} - y_{Ci} = \tau_i$: the treatment effect is manifest entirely in the residuals e_C and e_T, and not at all in $\hat{y}_C(x)$.

The prediction errors e, then, may replace Y in an outcome analysis. In particular, replace matched-set-specific treatment-control differences in Y, $t_m(Y, Z)$ with differences in e: $t_m(e, Z)$. That is, let

$$t_m(e, Z) = \bar{e}_{m,Z=1} - \bar{e}_{m,Z=0} = \frac{1}{n_{Tm}} \sum_{i:m_i=m} e_i Z_i - \frac{1}{n_{Cm}} \sum_{i:m_i=m} e_i(1 - Z_i)$$

then define

$$\hat{\tau}_{\text{rebar}} = \sum_m w_m t_m(e, Z) \tag{5}$$

Residualization, then, means revising a matching estimator by replacing outcomes Y with observed value/$\hat{y}_C(\cdot)$ differences; it aims to rid the dependent variable of variation that is not informative about treatment effects. Rosenbaum (2002a) precedes conventional hypothesis tests with a residualization step, using observations within the matched sample to fit the prediction model. If one instead trains one's prediction algorithm $\hat{y}_C(\cdot)$ using the remnant of the matching procedure, the method becomes compatible with common estimation (as well as hypothesis testing) techniques, and may offer larger numbers of observations for training $\hat{y}_C(\cdot)$. Such *remnant-based* residualization, briefly "rebar," is the topic of this paper.

3.1 Cross Validation and Proximal Validation: Assessing $\hat{y}_C(\cdot)$

Using the remnant to model outcomes as a function of covariates affords the researcher a great deal of flexibility. Researchers may use data from the remnant—both covariates and outcomes—to attempt a variety of prediction techniques, and choose the one which performs best. This is particularly important when the dimension of $\boldsymbol{X}$ is large, so formulating statistical models based on theory or first principles is hard or impossible; a variety of methods must be attempted. A useful tool in this regard is k-fold cross-validation (Efron and Gong, 1983), which can estimate the predictive accuracy of a model using data from the training sample. Cross-validation results may be examined for bias, variance, or other measures of predictive performance, but Proposition 3 (below) suggests a focus on prediction mean-squared-error. In the rebar case, cross validation using data from the remnant can estimate $MSE_{\text{remnant}} = \mathbb{E}_{i \in remnant}(\hat{y}_{Ci} - y_{Ci})^2$ or $R^2_{\text{remnant}} = 1 - MSE_{\text{remnant}}/Var_{\text{remnant}}(y_C)$.[1] These

[1] In defining MSE_{remnant} and R^2_{remnant} thusly, we briefly depart from our convention of conditioning on potential outcomes and instead treat them as random, drawn from the same superpopulation as the remnant. MSE_{remnant} and R^2_{remnant} do not play a role in the theoretical development of rebar, but are useful heuristics in practice.

results can be used both to pick a modeling technique and to pick tuning parameters. After modeling choices have been made, researchers arrive at an estimated prediction function $\hat{y}_C(\cdot) : \mathbb{R}^p \to \mathbb{R}$ that generates predictions $\hat{y}_C(\boldsymbol{X})$ as a function of covariates $\boldsymbol{X}$.

Cross-validation estimates an algorithm's predictive performance when applied to new cases drawn from the same population as the training set. Of course, this is manifestly not the case for rebar. Subjects in the matched sample are likely to be different from those in the remnant; a model fit and cross-validated in the remnant may not perform as well in the matched sample as that validation would suggest. Write $\mathcal{S}_M$ to denote the matched sample, i.e. $\{i : \exists j \neq i \text{ s.t. } M_i = M_j\}$. One expects $\text{MSE}_{\text{remnant}}$ to be less than $\text{MSE}_M = \{\sum_{i\in\mathcal{S}_M}(\hat{y}_{Ci} - y_{Ci})^2\}/|\mathcal{S}_M|$, and R^2_{remnant} to be less than R^2_M. This is unfortunate but far from fatal—the more information a prediction algorithm can learn about the matched sample from the remnant the better rebar can reinforce a causal design. Perfection is not necessary.

One does not expect MSE_M to exceed $\{\sum_{i\in\mathcal{S}_M}(y_{Ci} - \overline{y_C}_{\mathcal{S}_M})^2\}/|\mathcal{S}_M|$, although this can occur. In such cases rebar could do more harm than good. Even with perfect matching in the sense of (2), it could diminish efficiency; and if (2) is only approximately true, rebar could increase bias as well.

Fortunately, simple diagnostic tools can identify such pathological cases. Further, in many of *those* cases there are simple modifications to rebar that will improve its performance. To illustrate a diagnostic that we call "proximal validation," consider full matching within calipers of width c_0 in terms of a continuous variable or index, such as the propensity score. All control subjects within c_0 of a treated subject are matched, with remaining controls constituting the remnant. How well does an algorithm $\hat{y}_C(\cdot)$ fit in the remnant perform in the matched sample? To gauge $\hat{y}_C(\cdot)$'s performance, a researcher will subdivide the remnant into two groups by using caliper $c_1 > c_0$ to construct a new, larger matched set. The cases in the remnant that are matched under the more permissive caliper c_1 are "proximal" cases—

whether they are matched depends on the choice of caliper. The cases that remain unmatched even under c_1 are "distal" cases, unmatchable under either scheme. Proximal validation refits $\hat{y}_C(\cdot)$ using only data from subjects in the distal remnant, then examines its performance on the proximal portion of the remnant. If $\hat{y}_C(\cdot)$ performs poorly when extrapolated from the remnant to the matched set, it likely also performs poorly when extrapolated from distal cases to proximal cases within the remnant. In other words, proximal validation is a way to gauge the performance of $\hat{y}_C(\cdot)$ when its results are extrapolated in a way analogous to a matching design.

As compared to estimating MSE_M with rebar's MSE on the control group, proximal validation permits the analyst to keep matched subjects' outcomes in Rubin's (2008) virtual locked box, even as the rebar model is being validated and improved. Proximal validation is not limited to propensity-score full-matching designs with calipers; it may be used with any matching design that involves a quantitative restriction on allowable matches. The procedure, in general, will be to slightly relax that restriction, choose a second, more expansive match, and use the results to divide the remnant into proximal and distal portions.

If $\hat{y}_C(\cdot)$'s performance in proximal validation is discernibly worse than its cross-validation performance, the rebar routine should be modified. Suppose the mechanism selecting untreated units between the remnant and the matched sample is matching based on an estimated propensity score. In this case, the estimated propensity score itself can be incorporated into the prediction model $\hat{y}_C(\cdot)$—for instance, by including interaction terms between the columns of $\boldsymbol{X}$ and $\hat{\pi}$.

Another useful diagnostic test is to check covariate balance on the predictions $\hat{\boldsymbol{y}}_C(\boldsymbol{X})$. Since $\hat{y}_C(\boldsymbol{X})$ is a covariate, a successful matching design will ensure that its distributions are similar among treated and matched untreated subjects. Even though $\hat{y}_C(\boldsymbol{X})$ is a constructed variable, its balance can be tested in the same ways as balance on manifest variables, since the model behind it is fit without reference to the matched sample. If a balance test rejects the hypothesis of $\hat{y}_C(\boldsymbol{X})$ balance, researchers may revise either the prediction algorithm $\hat{y}_C(\cdot)$, the matching scheme, or both.

4 Rebar's Effects on Bias

To see the potential of rebar to reduce the bias of a matching estimator, note that the rebar estimator $\hat{\tau}_{\text{rebar}}$ can be expressed as the difference between two estimated treatment effects:

$$\hat{\tau}_{\text{rebar}} = \hat{\tau}_M(\boldsymbol{Y}) - \hat{\tau}_M(\hat{\boldsymbol{y}}_C) \tag{6}$$

the matching estimator of the effect of the treatment on Y, minus an estimate of the effect of the treatment on $\hat{y}_C(\boldsymbol{X})$. To see this, note that:

$$\begin{aligned} t_m(e, Z) =& \frac{1}{n_{Tm}} \sum_{i:\boldsymbol{M}_i=m} e_i Z_i - \frac{1}{n_{Cm}} \sum_{i:\boldsymbol{M}_i=m} e_i(1-Z_i) \\ =& \left(\frac{1}{n_{Tm}} \sum_{i:\boldsymbol{M}_i=m} Y_i Z_i - \frac{1}{n_{Cm}} \sum_{i:\boldsymbol{M}_i=m} Y_i(1-Z_i) \right) - \\ & \left(\frac{1}{n_{Tm}} \sum_{i:\boldsymbol{M}_i=m} \hat{y}_{Ci} Z_i - \frac{1}{n_{Cm}} \sum_{i:\boldsymbol{M}_i=m} \hat{y}_{Ci}(1-Z_i) \right) \\ \equiv& \Delta Y_m - \Delta \hat{y}_{Cm}. \end{aligned}$$

The expression in (6) follows by taking weighted averages of ΔY_m and $\Delta \hat{y}_{Cm}$. Of course, the treatment cannot have an effect on $\hat{y}_C(\boldsymbol{X})$, which is a function of pre-treatment covariates and a separate sample; any observed "effect" of the treatment on $\hat{y}_C(\boldsymbol{X})$ must be the result of covariate imbalance.

Two properties of the rebar estimate follow immediately.

Proposition 1.

$$bias(\hat{\tau}_{rebar}) = bias(\hat{\tau}_M(\boldsymbol{Y})) - \hat{\tau}_M(\hat{\boldsymbol{y}}_C)$$

Viewing $\hat{\tau}_M(\hat{\boldsymbol{y}}_C)$ as an estimate of $\hat{\tau}_{\text{rebar}}$'s bias, the effect of residualization is to subtract from the matching estimator an estimate of its bias. (As with other bias correction methods, it backfires when the bias is poorly estimated, an eventuality proximal validation aims to detect.)

Proposition 2. *Under perfect matching (2), $\hat{\tau}_{rebar}$ is unbiased for τ_{ETT}.*

This follows since, when treatment is essentially randomized within matches, $\mathbb{E}\hat{\tau}_M(\boldsymbol{Y}) = \tau_{ETT}$ and $\mathbb{E}\hat{\tau}_M(\hat{\boldsymbol{y}}_C) = 0$. So in a successful matching design, rebar does not introduce bias. Propositions 1 and 2 hold for any effect estimator $\hat{\tau}(\cdot)$ that is linear in outcomes Y, i.e. for which (6) holds.

4.1 An Upper Bound on the Bias of the Rebar Estimator

The closer, on average, predictions $\hat{y}(x)$ are to control potential outcomes in the matched set, the smaller the bias of $\hat{\tau}_{\text{rebar}}$ must be.

Proposition 3. *In a matching design, the squared bias of $\hat{\tau}_{rebar}$ can be bounded as*

$$bias(\hat{\tau}_{rebar})^2 \leq MSE_M \times C(n, \boldsymbol{n_T}, \boldsymbol{n_C})$$

where $MSE_M = \sum_{i \in matched}(\hat{y}_{Ci} - y_{Ci})^2/n_M$, n_M is the number of subjects in the matched set, and

$$C(n, \boldsymbol{n_T}, \boldsymbol{n_C}) = \frac{n}{n_T^2}\sum_m (n_{Cm} + n_{Tm}) \max\left(1, \frac{n_{Tm}}{n_{Cm}}\right)^2.$$

Equivalently,

$$\left(\frac{bias(\hat{\tau}_{rebar})}{SD(y_C)}\right)^2 \leq (1 - R_M^2) \times C(n, \boldsymbol{n_T}, \boldsymbol{n_C})$$

Where $SD(y_C)$ is the sample standard deviation of y_C in the matched set and R_M^2 is the prediction R^2 in the matched set, $1 - \sum_{i \in matched}(y_{Ci} - \hat{y}_{Ci})^2 / \sum_{i \in matched}(y_{Ci} - \bar{y}_{Cmatched})^2$.

A proof of proposition 3 appears in the Appendix.

Remark 1. *In a pair-matching design $C(n, \boldsymbol{n_T}, \boldsymbol{n_C}) = 4$.*

Therefore, the bias of $\hat{\tau}_{\text{rebar}}$ can be bounded as a function of the average squared error of the prediction algorithm in the matched set. Were it possible to perfectly predict all subjects' y_C values, their treatment effects could be estimated unbiasedly (exactly, in fact). More broadly, Proposition 3 suggests that prediction algorithms need not be based on a correct model to yield estimates with low bias. They must merely be accurate, on average. This, in turn, suggests that machine learning algorithms, whose central purpose tends to be prediction, can serve well as residualization mechanisms.

In practice, the bounds in Proposition 3 are unobservable, since they involve control potential outcomes in the matched set, which are only observable for the matched controls. Further, since the prediction algorithm $\hat{y}_C(\cdot)$ is fit in the remnant, the bounds are not directly estimable without strong assumptions. But based on cross-validation estimates of MSE_{remnant} and R^2_{remnant}, and an assessment of $\hat{y}_C(\cdot)$'s sensitivity to extrapolation from proximal validation, researchers can formulate reasonable guesses as to the values of MSE_M and R^2_M.

Proposition 3 assumes nothing about subjects' respective probabilities of treatment assignment within matches. In particular, it allows for a situation in which some subjects may be assigned to treatment with probability 1—this is a rather extreme violation of the stratified randomization assumption (2). Under weak assumptions about the distribution of treatment assignments, the bound in Proposition 3 may be considerably tightened. For instance, Rosenbaum (2002b) suggests a general model for sensitivity analysis for observational studies: the assumption that for some $\Gamma \geq 1$, if $m_i = m_j$—that is, i and j are in the same matched set—and $P_i = Pr(Z_i = 1)$ and $P_j = Pr(Z_j = 1)$, then

$$\frac{1}{\Gamma} \leq \frac{P_i(1-P_j)}{P_j(1-P_i)} \leq \Gamma. \tag{7}$$

That is, for matched subjects i and j, the ratio of the odds that i is selected for treatment to the odds that j is selected is bounded by $1/\Gamma$ and Γ. The following proposition uses the framework in (7) to tighten the bound in Proposition 3 in the simple case of a matched-pair design; an analogous result may hold for more complex designs, but we leave such an extension for future work.

Proposition 4. *In a pair-matching design, if (7) holds for some* $\Gamma \geq 1$*, then*

$$bias(\hat{\tau}_{rebar})^2 \leq MSE_M \times 4\left(\frac{\Gamma^{1/2}-1}{\Gamma^{1/2}+1}\right)$$

Equivalently,

$$\left(\frac{bias(\hat{\tau}_{rebar})}{SD(y_C)}\right)^2 \leq (1-R^2_M) \times 4\left(\frac{\Gamma^{1/2}-1}{\Gamma^{1/2}+1}\right)$$

A proof of Proposition 4 is given in the Appendix.

Remark 2. *For* $\Gamma = 6$*, which Rosenbaum (2002b, p. 114) characterized as "a high degree of insensitivity to hidden bias,"* $4\left(\frac{\Gamma^{1/2}-1}{\Gamma^{1/2}+1}\right) \approx 1.7$*. That is, a very weak assumption about the balance of treatment assignment probabilities in a matched pair design constricts the bound in Proposition 3 by more than half. If* $\Gamma = 3$*, the multiplier on* $(1-R^2_M)$ *is approximately one. On the other hand, as* $\Gamma \to \infty$*, the multiplier approaches 4, as in Remark 1.*

Propositions 3 and 4 show that by using data from the remnant and covariate matrix $\boldsymbol{X}$ to predict potential outcomes y_C, researchers can substantially bound the the bias of their treatment effect estimates. The closer the estimates are to the true values, on average, the lower the bound on the bias—the algorithm $\hat{y}_C(\cdot)$ need not be correct in any sense, only predictive.

5 A Simulation Study

This section presents a simulation study with two principal goals: to demonstrate rebar's potential to improve upon matching estimators under a variety of circumstances, and rebar's ability to interact with, and improve upon, a variety of matching designs and estimators. A second, smaller study examines rebar's performance under pathological circumstances.

5.1 Data Generating Models

The study imagined a researcher estimating the effect of a treatment Z on an outcome Y, using a sample of n =400 subjects, in the presence of p =600 covariates. While all of the covariates were potential confounders, the simulated researcher knew that five of the covariates—the first five columns of covariate matrix $\boldsymbol{X}$—predict both y_C and Z; prior background knowledge provided little guidance regarding the remaining 595.

The outcomes y_C were generated as a linear function of a fixed covariate vector x_i:

$$y_{Ci} = \mathbf{1}^T x_{i,1:5} + \boldsymbol{\beta}^T x_{i,6:600} + \epsilon_i \tag{8}$$

where the coefficients $\boldsymbol{\beta}$ were drawn from an exponential distribution with a rate of $\lambda = 5$ and ϵ was drawn from a standard normal distribution. A "treated" group was selected according to probabilities

$$Pr(Z_i = 1|x_i) = \text{logit}^{-1}(\alpha^* + \mathbf{1}^T x_{i,1:5} + \kappa\boldsymbol{\beta}^T x_{i,6:600}). \tag{9}$$

That is, the log odds of treatment assignment were linear in covariates. We chose the parameter α^* in such a way that, on average, n_T =50 were treated. As in (8), the coefficients for the first five columns of $\boldsymbol{X}$ in (9) were all set equal to 1. The coefficients of the other 595 columns in (9) were the same as in (8), multiplied by a factor κ which varied between simulation runs.

The factor κ controlled the amount of confounding after matching. When $\kappa = 0$, only the first five columns of $\boldsymbol{X}$ predicted Z, so estimates from a match based on those covariates were approximately unconfounded. When $\kappa > 0$, every column of $\boldsymbol{X}$ predicted both Z and y_C, and therefore confounded matching estimators that used only the first five columns of $\boldsymbol{X}$. As κ increased, so did the magnitude of the bias due to confounding after the match; the three values we assigned, κ =0, 0.1, 0.5, roughly correspond to zero, low, and high unmatched confounding.

A second parameter, ρ, controlled the covariance structure of $\boldsymbol{X}$, effectively controlling the ease of predicting y_C as a function of $\boldsymbol{X}$. In this simulation, ρ =0, 0.004, and 0.05. The rows of $\boldsymbol{X}$ were generated from a p =600-dimensional multivariate normal distribution, with a random covariance matrix whose eigenvalues we specified (it was generated with `R` code of Varadhan 2008). We set these eigenvalues ev_k, $k = 1, ...,$600, to decay exponentially: $ev_k = \exp\{-\rho k\}$. When $\rho = 0$, all eigenvalues were unity, and the columns of $\boldsymbol{X}$ are uncorrelated. As ρ increased, the columns of $\boldsymbol{X}$ became increasingly correlated: there was low-dimensional structure in $\boldsymbol{X}$. Prediction algorithms typically perform better when high-dimensional $\boldsymbol{X}$ can be summarized with a low-dimensional structure. During the simulation we recorded the estimated prediction R^2 from the cross-validation, and models fit to $\boldsymbol{X}$ with higher ρ fit substantially better.

Covariates $\boldsymbol{X}$ and coefficients $\boldsymbol{\beta}$ varied between scenarios (one random matrix x for each value of ρ and one random vector $\boldsymbol{\beta}$ for each value of κ) but were held fixed across simulation runs within scenarios. Outcomes Y and treatment assignments Z were generated anew in each simulation run. Each run, all ten effect estimates were computed using the same data.

5.2 Treatment-Effect Estimators

In each simulation run, we constructed four matches. Each of these matches, in turn, gave rise to two or three treatment-effect estimates; all in all, we compared 10 different estimators.

Matching Method	Matching Variables	Adjustment Method(s)
Optimal pair matching	$\boldsymbol{X}_{1:5}$	Rebar
Nearest Neighbor	$\boldsymbol{X}_{1:5}$	Bias adjusted bias adjusted+rebar
Coarsened Exact Matching	$\boldsymbol{X}_{1:5}$	Within-sample OLS within-sample OLS+rebar
High-dimensional pair match	$\boldsymbol{X}$	Rebar

Table 1: Summary of the matching and estimation methods in the simulation study.

These are summarized in Table 1.

Optimal Propensity Score Pair Matching

We estimated propensity scores using logistic regression, with $\mathbf{Z}$ regressed on the matching covariates, the first five columns of $\boldsymbol{X}$. Using these propensity scores, we constructed an optimal pair match without replacement—each treated subject was matched to a unique control subject in such a way that the total distance in propensity scores between matched subjects was minimized. (We used the `optmatch` package in `R` [Hansen 2007] and chose pair matching strictly for ease of interpretation; the application of § 6 uses `optmatch` to pair each treated subject to 1–4 controls.) We first estimated treatment effects via (3), the average difference in Y between treated subjects and their matched controls, without adjustment from an outcome model.

Next, we computed rebar-adjusted estimates. With the remnant from the pair-match as a training set, we used a combination of lasso (Tibshirani, 1996) and random forests (Breiman, 2001) to construct $\hat{y}_C(\cdot)$, a predictor of control potential outcomes y_C as a function of the entire covariate matrix $\boldsymbol{X}$. We implemented these in `R` with the `glmnet` and `randomForest` packages (Friedman et al., 2010; Liaw and Wiener, 2002), and tuned and combined them with the `SuperLearner` package (Polley and van der Laan, 2014) to minimize mean-squared-error. As outlined in Section 3, we used the fitted $\hat{y}_C(\cdot)$ to construct predictions $\hat{y}_C$ and prediction errors e in the matched set, and estimated the treatment effect as in equation (5).

Nearest-Neighbor Propensity Score Matching

Using the same propensity scores as in the optimal pair match, we constructed a "nearest-neighbor" match, as proposed by Abadie and Imbens (2006), and implemented by the `Matching` package in `R` (Sekhon, 2011). We used the "ATT" estimator of Abadie and Imbens (2006) to estimate the average of the differences between each treated subject's outcome and the average outcome of its matched controls. Next, we computed the "bias adjusted" estimator suggested in Abadie and Imbens (2012), using an ordinary least squares (OLS) outcome model fit to the matched sample.[2] Since OLS cannot be fit when the number of covariates exceeds the sample size, we used only the matching covariates for the bias adjustment. Finally, we combined this within-sample bias adjustment with rebar. As in optimal pair matching, we fit the lasso/random forest/SuperLearner algorithm to data from the remnant of the nearest neighbor match, predicting y_C as a function of the entire matrix $\boldsymbol{X}$, and computed $\hat{y}_C$ and e in the matched set. To estimate effects with both within-sample and rebar adjustment, we substituted e for Y in the bias-adjusted estimator.

Coarsened Exact Matching

We constructed a coarsened exact match, as described in Iacus et al. (2011) and implemented in `R` with the `cem` package (Iacus et al., 2015). We coarsened each of the first five columns of $\boldsymbol{X}$ with five bins, matched exactly on the coarsened covariates, and estimated treatment effects via (3). Next, we constructed a within-sample adjusted estimator along the lines of Ho et al. (2007b): using only data from the matched sample, we regressed Y on Z and the first five columns of $\boldsymbol{X}$, and recorded the coefficient on Z. Finally, we combined the within-sample adjustment with rebar. As in the optimal pair and nearest neighbor analyses, we used data from the remnant to fit a lasso/random forest/SuperLearner algorithm predicting y_C as a function of the entire $\boldsymbol{X}$, and generated predictions $\hat{y}_C$ and errors e in the matched set. To estimate effects, we regressed e on Z and the first five columns of X, and recorded

[2] Abadie and Imbens (2012) in fact suggest a more complicated regression routine that includes non-linear terms and interactions as the sample size grows, but in practice implement the routine with OLS; the `Matching` package similarly uses OLS.

the coefficient for Z.

High-Dimensional Pair Match

The first three matching designs, optimal pair matching, nearest neighbor matching, and coarsened exact matching, used only the first five columns of $\boldsymbol{X}$—the known confounders. However, when presented with a set of $p = 600$ covariates, many real-world researchers would not stop at the first five. Instead, they would try to incorporate additional covariates into their matches. The resulting iterative process of matching and balance checking is difficult or impossible to simulate; however, there are a number of automatic machine learning algorithms for estimating probabilities in high-dimensional spaces (e.g. McCaffrey et al., 2013; Lee et al., 2010). In this vein, in parallel to the rebar prediction model $\hat{y}_C(\cdot)$, we estimated high-dimensional propensity scores with random forest classification and lasso logistic regression, tuned and combined via the SuperLearner. We used these high-dimensional propensity scores to construct a second optimal pair match. As in the conventional pair match, we estimated effects using equation (3) and, fitting algorithm $\hat{y}_C(\cdot)$ to the remnant, we computed a rebar estimate.

5.3 Simulation Results

Figure 1 shows the results of the simulation, after 1000 simulation runs. Each row of Figure 1 corresponds to a value of κ; in the first row, κ =0, corresponding to no confounding from the covariates not used in the match, in the second row κ =0.1, corresponding to moderate confounding from the left-out covariates, and in the third row κ =0.5, corresponding to a high degree of confounding. Each column of Figure 1 corresponds to a different value of ρ: 0, 0.004, and 0.05. These correspond to datasets increasingly amenable to prediction algorithms; the top of the figure lists the average cross-validation R^2_{remnant} of $\hat{y}_C(\cdot)$ fit in the remnants from the pair matches. Each panel of Figure 1 displays boxplots of the ten treatment effect estimates, divided by the standard deviation of y_C.

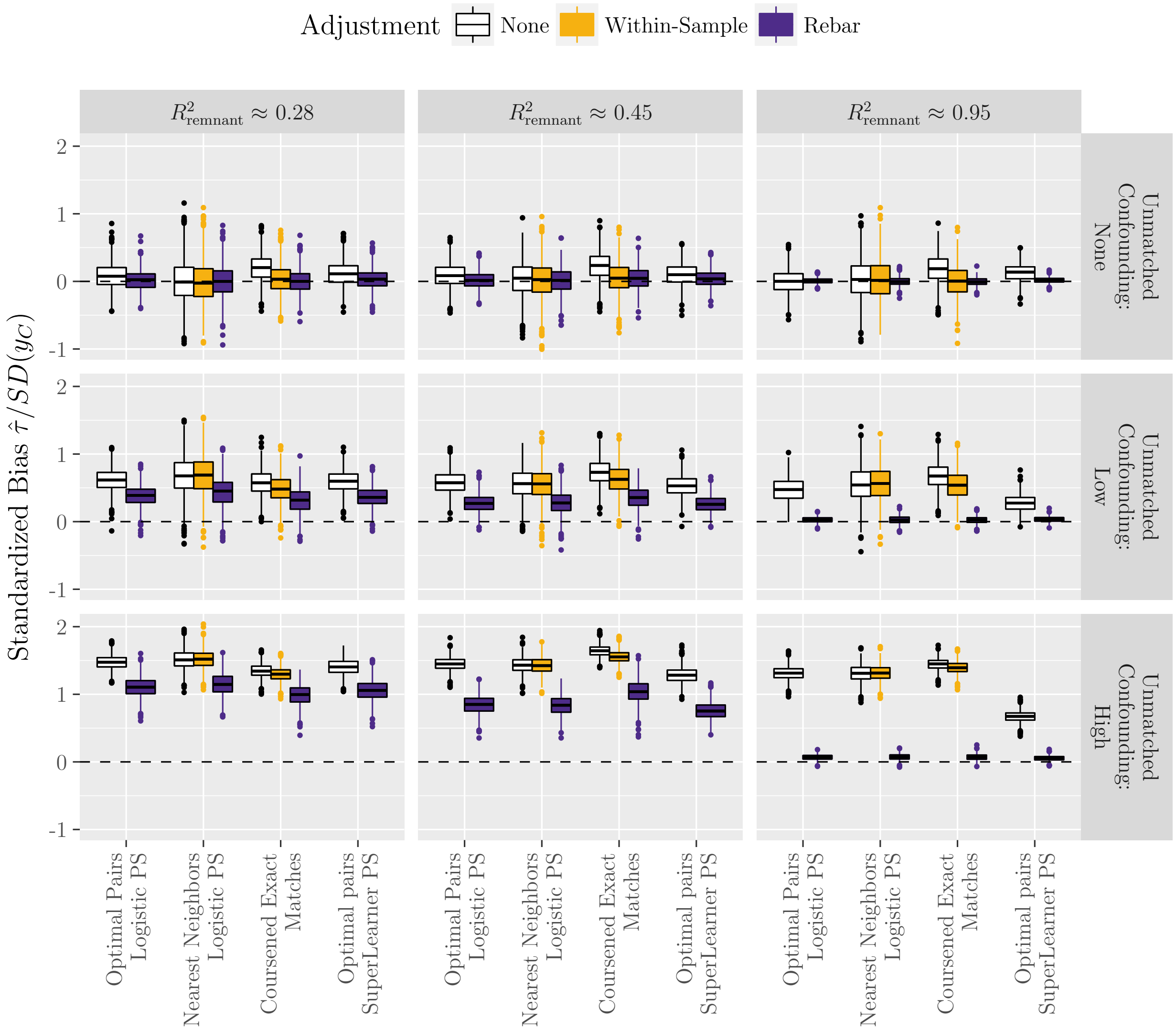

Figure 1: Boxplots of treatment effect estimates from 1000 simulation runs under the data generating models in Section 5.1. The true treatment effect of zero is indicated by a horizontal dotted line. The estimated treatment effects were divided by the standard deviation of y_C. The matching and outcome adjustment methods are described in Section 5.2 and Table 1; the rebar adjustments to the nearest neighbor and coarsened exact match were done alongside within-sample adjustment.The nine simulation scenarios, described in Section 5.1, are arranged in a matrix, with rows for κ =0, 0.1, and 0.5, and columns for ρ =0, 0.004, and 0.05. The R^2_{remnant} values listed are averages of prediction R^2 for $\hat{y}_C(\cdot)$ estimated using cross-validation within the remnant.

A number of patterns are apparent. When $\kappa = 0$, the covariates not used in the match did not pose a confounding threat, and all the estimators were approximately unbiased. Rebar reduced the variance of the effect estimates, subtly for the first two columns and dramatically in the third. As κ increased, all effect estimates became increasingly biased. However, rebar substantially reduced the bias. Rebar was similarly effective when used on its own and when used in conjunction with within-sample outcome model adjustments—that is, rebar had quite a bit to add even after other adjustments. Unsurprisingly, rebar's performance, both in terms of bias and variance reduction, improved with higher R^2_{remnant}—the closer, on average, the predictions $\hat{y}_C(\boldsymbol{X})$ are to y_C in the remnant (and, presumably, in the matched set, too), the more good rebar can do.

The high-dimensional propensity score match demonstrated that rebar can improve upon designs that incorporate all of $\boldsymbol{X}$.

This simulation study showed rebar's potential: rebar can substantially reduce both the bias and the variance of a matching estimator, especially in the presence of high-dimensional confounding and with an accurate prediction algorithm.

5.4 Rebar's Performance Under Non-Linearity

We conducted a parallel simulation study to investigate rebar's performance when the distribution of y_C, conditional on X, differs greatly between the remnant and the matched set. Since it is the match that determines which subjects are in the matched set and which are in the remnant, and the data generation occurs prior to the match, we could not set the distribution of y_C in the remnant exactly. Instead, we let the data generating model for y_C vary with $Pr(Z = 1)$, subjects' probabilities of being treated. To do so, we modified both the outcome model (8) and the treatment model (9). To select treated subjects, we chose those $2n_T$ with the highest linear predictors, as defined in equation (9), and assigned half to treatment. That left an "untreatable" group of subjects with $Pr(Z = 1) = 0$. For the

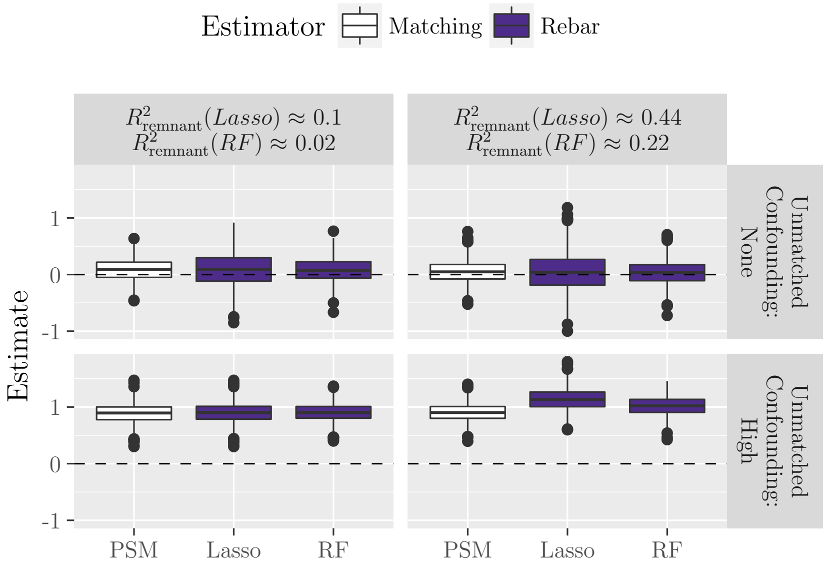

Figure 2: Boxplots of standardized treatment effect estimates from 1000 simulation runs under the data generating models in Section 5.4. The true treatment effect, indicated by a horizontal dotted line, is zero. The methods are optimal pair matching (PSM) and rebar-adjusted optimal pair matching, with y_C predicted using lasso or random forests (RF). The four simulation scenarios are arranged in a matrix, with rows for κ =0 and 0.5 and columns for $\rho = 0$ and 0.05. The R^2_{remnant} values listed are averages of prediction R^2 for $\hat{y}_C(\cdot)$ estimated using cross-validation within the remnant for lasso and random forest.

untreatable subjects, y_C was generated as in (8). For the $2n_T$ subjects with $Pr(Z = 1) = 0.5$, the outcomes were generated as $\overline{x\beta^*} - x\beta^* + \epsilon$, where β^* is the concatenation of a vector of five 1s with β and $\overline{x\beta^*}$ is the sample average of all subjects' $x\beta^*$. Finally, we transformed y_C to $-y_C$, so that the omitted variable bias would be positive, as in Section 5.3. In this study, the relationship between x and y_C for subjects who could be treated was precisely the opposite of the relationship for subjects who could not. The worry here was that $\hat{y}_C(\cdot)$ would be severely misleading, if fit in the remnant and extrapolated to the matched set.

The simulation results suggest that this is, indeed, a concern—in some cases. Figure 2 shows the results of rebar adjustment to optimal pair matching using two different rebar algorithms $\hat{y}_C(\cdot)$: lasso, which depends on a linear model, and random forest, which does not. Rebar adjustment with lasso worsened the bias and variance of the matching estimator, slightly for lower R^2_{remnant} values and considerably for higher R^2_{remnant}. On the other hand, rebar using random forests, which achieved much lower R^2_{remnant} values across the board, did little to no damage to the matching estimator. Apparently the matching routines were unable, in general, to perfectly identify the treatable control subjects with $Pr(Z = 1) = 0.5$, so both the remnant and the matched set contained subjects with outcomes drawn from both outcome models. By ignoring non-linearity, the lasso was able to fit the training sample more closely than random forests did; but because of their sensitivity to non-linearity, random forests extrapolated beyond the remnant more reliably than the lasso.

In summary, under data generating models combining nonlinear responses with limited propensity score overlap, rebar's performance depended on the prediction algorithm. Rebar adjustment via lasso increased the MSE of the matching estimator, while rebar adjustment via random forest caused little to no harm. It is unclear whether latent nonlinearities sufficient to undercut the lasso are to be expected in applications; proximal validation may flag some such situations.

6 Example Data Analysis: Evaluating Board Exam Systems

Board Exam Systems (BES) comprise a class of similar comprehensive educational reforms. BES are packages that a school can adopt: sets of rigorous curricula for all academic courses, corresponding sets of end-of-course exams, professional development and instructional guidance for teachers and systems of assistance for struggling students. Though uncommon in the United States, BES are common around the world, and several research studies have suggested that they improve student achievement (Bishop, 1997, 2000; Collier and Millimet, 2009)

Seven Arizona High Schools began implementing BES programs in the 2012–2013 school year: either the ACT Quality Core program or the Cambridge program. A pilot study sought to evaluate the results after one year, in part by estimating the effects of the BES programs on 10th-graders' end-of-year standardized test scores—specifically, the Arizona Instrument to Measure Standards, or AIMS. Here we present a simplified version of the study's estimate of the effect of BES on school-average 10th-grade AIMS Reading scores. The analysis we present here is intended to illustrate the rebar method, not to evaluate the effectiveness of BES programs in Arizona.

For Arizona high schools in our sample, we had four years of pre-treatment data. That is, data from four cohorts of students who preceded the adoption of BES—students set to graduate in 2011–2014. For each cohort, we had the total enrollment, the percents of students who were male, white, black, Hispanic, other race or ethnicity, receiving free or reduced-price lunches (FRL), special education (SPED), and English language learners (ELL), in addition to average 8th Grade and 10th Grade AIMS scores on writing, reading, math and science. We also had the percent of students in each cohort with missing AIMS English and Math scores. From these data, we computed composite AIMS scores by averaging the four AIMS components, and school "trends" for 10th grade math and reading scores: ordinary-least-squares slope estimates from the school-level regressions of school mean AIMS scores on a linear time variable. From the US National Center for Education Statistics Common Core of Data (NCES, 2013), we had a categorization of each school into one of 10 categories of urbanicity, ranging from urban to remote rural. All in all, there were 90 covariates, for a total of 509 high schools.

6.1 A Propensity Score Match

We constructed a propensity score match to estimate treatment effects. Since there were only $n_T = 7$ intervention schools, estimating propensity scores with logistic regression including all 90 predictors was not feasible. Instead, our propensity score model incorporated only a small subset of the covariates, those that we believed would be most recognizable as potential confounders to the end audience of the research. Specifically, we regressed schools' BES status on the percent FRL, white, SPED, Hispanic, and average and percent missing 8th and 10th grade AIMS scores for students in the cohort immediately prior to BES implementation (those set to graduate in 2014) along with estimated school trends in English and Math AIMS scores. Since this still gave more predictors than there were observations in the treatment group, we expected that classical logistic regression would fail to fit, so we instead used the Bayesian variant implemented in the `arm` library for `R` (Gelman and Su, 2015; Gelman et al., 2008).

We constructed optimal propensity-score matches, using the `R` `optmatch` package (Hansen, 2007) to minimize paired differences in the estimated log odds of assignment to treatment. Given the relatively large pool of available comparison schools, we disallowed the sharing of controls, as in nearest-neighbor matching or full matching, while permitting multiple matches per treatment schools. Rather than leaving the maximum number of matched comparisons per treatment unspecified, we restricted it to 4, a restriction that reduces the overall information content of the matched sample (Cinar and Zubizarreta, 2016) only modestly relative to matching without an upper limit on the number of matched controls per treatment. (Each matched set m makes a contribution to effective sample size comparable to $h(n_{Tm}, n_{Cm})$ matched pairs, where $h(n_{Tm}, n_{Cm}) = \{\frac{1}{2}(n_{Tm}^{-1} + n_{Cm}^{-1})\}^{-1}$ is the harmonic mean of n_{Tm} and n_{Cm} [Hansen, 2011; Cinar and Zubizarreta, 2016]. For $n_{Tm} = 1$ and $n_{Cm} \geq 1$, this contribution varies between 1 and 2, with $h(1, 4) = 1.6$.) If this left plausible matches for some treatment-group schools on the table, these eligible but unused comparisons would enhance the value of proximal validation, improving its ability to detect shortcomings of the extrapolation that underlies rebar.

Table 2 displays covariate balance for the variables in the propensity score model—standardized differences in covariate means and Z-scores—before and after matching. Covariate balance was assessed with the `xBalance` routine in the `RItools` package from `R` (Bowers et al., 2010). The `xBalance` routine also returns the results of omnibus balance tests, for the full sample and the matched sample. They returned p-values of 0.04 and 0.71, respectively. Evidently, the propensity score match controlled some covariate imbalance that was in the full sample.

6.2 Rebar to Adjust the Match

6.2.1 Estimating $\hat{y}_C(\cdot)$

After setting aside the treated schools and their untreated matches, there were 483 schools in the remnant. We considered four different predictive modeling strategies to construct $\hat{y}_C(\cdot)$: the lasso, random forests, ridge regression (Hoerl and Kennard, 1970; Venables and Ripley, 2002), and linear regression with weak priors for regularization (Gelman and Su, 2015), along with grand-mean prediction, all combined via the SuperLearner. The SuperLearner uses cross validation to estimate the prediction mean-squared-error of each of the modeling

	Std. Diff.		
	Unmatched		Matched
% FRL	1.06	**	0.08
% White	-0.97	*	0.02
% Sp.Ed.	-0.01		-0.19
% Hispanic	1.34	***	0.03
Urban	0.24		0.13
avg. AIMS Writing (8th)	0.31		-0.10
avg. AIMS Reading (8th)	0.42		-0.18
avg. AIMS Math (8th)	0.79	*	0.06
avg. AIMS Reading (10th)	-0.55		0.14
avg. AIMS Math (10th)	-0.27		0.05
avg. AIMS Writing (10th)	-0.46		-0.01
trend: AIMS English (10th)	-0.37		0.11
trend: AIMS Math (10th)	-0.42		0.10
% AIMS Eng. Missing	-0.27		-0.17
% AIMS Math Missing	-0.20		-0.22
$\hat{y}_C(x)$	-0.06		0.14

Table 2: Standardized differences testing balance on covariates from the propensity score model and predictions $\hat{y}_C(\boldsymbol{X})$ in the entire sample of schools and for the matched sample, conducted with the xBalance procedure.

algorithms in a library. Then, it constructs an "ensemble learner," predicting new values as a weighted average of the predictions from each of the algorithms, with the weights determined by the cross-validation results. These results are displayed in Table 3. The random forest dominated the other algorithms, with a prediction R^2 of 0.66, to the extent that its ensemble weight was 1.

6.2.2 Proximal Validation

To gauge how a model trained on the remnant might perform on the matched sample, we conducted proximal validation, described in Section 3.1. First, we constructed a second match, m^{big}, identical to the first, but allowing each treated subject to match at most 10 control subjects. Match m^{big} included an additional 31 control schools in the matched set—proximal schools—leaving 452 distal schools as a training set. We trained the SuperLearner

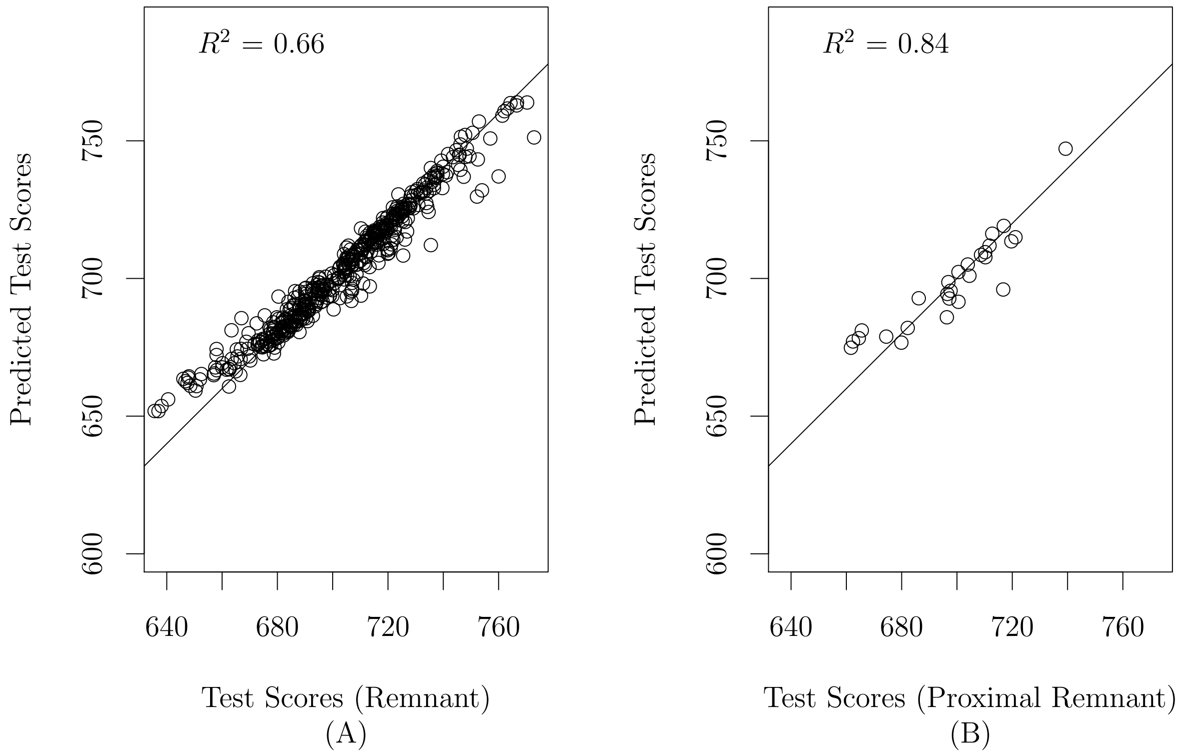

Figure 3: SuperLearner prediction accuracy: predictions ($\hat{y}_C(\boldsymbol{X})$) as a function of real test scores. (A) gives the results of the SuperLearner fit to, and tested against, the entire remnant. (B) shows the proximal validation results: the performance of the SuperLearner fit in the distal portion of the remnant and tested against the proximal portion. The figures also contain the $y = x$ line for comparison.

	Lasso	Random Forest	BayesLM	Ridge	Mean
RMSE	18.10	15.56	45.36	18.54	26.93
R^2	0.55	0.66	-1.85	0.52	-0.00
coefficient	0.00	1.00	0.00	0.00	0.00

Table 3: CV root-mean-squared error, R^2, and ensemble learner weight from the SuperLearner. The seven models displayed are the lasso, random forest, a linear model with weak priors on the coefficients ("BayesLM"), ridge regression, and a grand mean model

on the distal schools, and computed its prediction accuracy against the proximal schools. Somewhat surprisingly, the prediction models performed better when trained on the distal schools and tested on the proximal schools than when trained and tested on random subsets of the remnant in cross-validation. This may be a result of sampling error, or the fact that the distal set contains a number of outlier schools whose AIMS reading scores are particularly hard to predict. These schools will increase the estimated MSE reported by any validation method that includes them in its testing set. If there are no outlier schools in the proximal set, proximal validation will not suffer from this difficulty.

As an additional check of the identification assumption (2) for match m, we tested balance on $\hat{y}_C(\boldsymbol{X})$, in the same way as for other covariates: we tested if $\mathbb{E}\hat{\boldsymbol{y}}_C^T\boldsymbol{Z}/n_T = \mathbb{E}\hat{\boldsymbol{y}}_C^T(\boldsymbol{1}-\boldsymbol{Z})/n_C$. The resulting p-value from the `xBalance` routine was 0.5; the balance test on $\hat{y}_C(\boldsymbol{X})$ does not falsify (2).

6.2.3 Estimating Treatment Effects

	Estimate	SE	p-value	95% CI
PSM	5.91	4.98	0.48	(-10.4,22.53)
rebar	1.82	3.65	0.57	(-5.41,12.17)

Table 4: The average treatment effect on the treated τ_{ETT}, along with regression standard errors and permutational p-values and 95% confidence intervals, estimated with conventional propensity-score matching, as described in Section 6.1, and with rebar.

Finally, we calculated both τ_M, the matching estimator using Y, and $\hat{\tau}_{\text{rebar}}$, the rebar matching estimator; these are shown in Table 4 along with HC3 standard errors. To estimate p-values, we conducted permutation tests, permuting treatment indicators within matched sets and re-computing the estimates. Ninety-five percent confidence intervals were estimated by inverting the permutation test, as in Rosenbaum (e.g. 2002a). Neither the conventional method nor rebar detected a statistically significant effect. However, the rebar estimate resulted in a confidence interval with less than half the width of the conventional interval.

An anonymous reviewer suggested a post-hoc assessment of $\hat{y}_C(\cdot)$'s fit: estimating R^2_M by comparing y_C from within the match to corresponding predictions $\hat{y}_C$. The result was $\hat{R}^2_M$ =0.71.

7 Conclusion

In structural engineering, "rebar" abbreviates "reinforcement bar," a metal beam that is embedded in concrete. Concrete is resistant to compression, whereas rebar is resistant to tension; the combination of the two materials, rebar and concrete, is robust to a variety of threats. Similarly, the rebar method of this paper complements the use of matching for confounder control. Whereas matching typically focuses primarily on possible confounders' associations with the treatment variable, and typically leaves some subjects unmatched, rebar addresses bias by using the remnant from matching, the unmatched controls, to model possible confounders' associations with outcomes. The predictions that result, $\hat{y}_C(x)$, extract information about subjects' control potential outcomes from the covariates $\boldsymbol{X}$. The process of residualizing, that is, subtracting predictions $\hat{y}_C(x)$ from outcomes Y, can neutralize confounding from variables that the match failed to balance.

Residualizing using the remnant confers these benefits without compromising the statistical rationale for matching. Indeed, matching supplemented with rebar inherits a number of central attractions of the matching estimator. For instance, researchers with any level of statistical training can assess the success of the matching procedure by examining matched units' comparability on substantively meaningful baseline variables. Although it typically makes use of data from outside the range of common support—the set of subjects i for which $0 < Pr(Z_i = 1|x_i) < 1$—its final estimate $\hat{\tau}_{\text{rebar}}$ compares only matched subjects, observing any common support restrictions that the matching procedure observed. The procedure is compatible with postponing analysis involving outcomes until the process of matching is complete, as recommended by Rubin (2008). If matching succeeds in recreating a latent experiment, where subjects matched to each other were assigned to treatment randomly, then $\hat{\tau}_{\text{rebar}}$, like $\hat{\tau}_M$, is unbiased.

Generating predictions $\hat{y}_C(x)$ involves extrapolating from the remnant to the matched sample; in some circumstances, the method could worsen the quality of matched inferences. This risk is mitigated with the use of cross-validation, to limit overfitting of the prediction model, followed by proximate validation, which additionally detects biases specific to extrapolation from lower- into higher-propensity score regions of x-space. Both forms of validation are assisted by the presence of a sizable matching remnant, including at least controls that would have been suitable matches for some treatment group members. While compatible with any method of matching that leaves a positive fraction of the control reservoir unmatched, rebar is particularly attractive in observational studies with many more untreated than treated subjects.

We have focused on the capacity of rebar to reduce bias, but the method may have other benefits as well. For instance, the confidence interval from a rebar analysis of the BES data had less than half the width of the confidence interval from the corresponding matching analysis. Indeed, confidence interval widths and standard errors generally vary inversely with the variance of the outcome. Unless the rebar extrapolation is sufficiently unstable as to worsen MSE — within the matched sample, the mean-square difference between rebar's out-of-sample prediction and Y exceeds the variance of Y — confidence intervals based on e

are bound to be tighter than those based on Y alone. In addition, studies with more stable outcomes tend to have lower design sensitivity (Rosenbaum, 2010; Zubizarreta et al., 2013). Barring instability, the rebar analysis will be less sensitive to confounding from unmeasured or unmodeled variables. The relative stability of e and Y is reflected in the prediction R^2 of the rebar $\hat{y}_C(\cdot)$ when applied to the matched set, for which cross-validation and proximal validation can suggest a plausible range.

References

Abadie, A. and Imbens, G. W. (2006), "Large sample properties of matching estimators for average treatment effects," *econometrica*, 74, 235–267.

— (2012), "Bias-corrected matching estimators for average treatment effects," *Journal of Business & Economic Statistics*.

Agresti, A. (2013), *Categorical data analysis*, John Wiley & Sons.

Austin, P. C. (2011), "An introduction to propensity score methods for reducing the effects of confounding in observational studies," *Multivariate behavioral research*, 46, 399–424.

Bishop, J. H. (1997), "The effect of national standards and curriculum-based exams on achievement," *The American Economic Review*, 87, 260–264.

— (2000), "Curriculum-based external exit exam systems: Do students learn more? How?" *Psychology, Public Policy, and Law*, 6, 199.

Bowers, J., Fredrickson, M., and Hansen, B. (2010), "RItools: Randomization inference tools," *R package version 0.1-11. URL: http://www. jakebowers. org/RItools. html.*

Braitman, L. E. and Rosenbaum, P. R. (2002), "Rare Outcomes, Common Treatments: Analytic Strategies Using Propensity Scores," *Annals of Internal Medicine*, 137, 693–695.

Breiman, L. (2001), “Random forests,” *Machine learning*, 45, 5–32.

Brookhart, M. A., Schneeweiss, S., Rothman, K. J., Glynn, R. J., Avorn, J., and Stürmer, T. (2006), “Variable selection for propensity score models,” *American journal of epidemiology*, 163, 1149–1156.

Cinar, K. and Zubizarreta, J. R. (2016), “Maximizing the Information Content of a Balanced Matched Sample in a Study of the Economic Performance of Green Buildings,” *Annals of Applied Statistics*, in press.

Collier, T. and Millimet, D. L. (2009), “Institutional arrangements in educational systems and student achievement: a cross-national analysis,” *Empirical Economics*, 37, 329–381.

Cox, D. (1958), *The Planning of Experiments*, John Wiley.

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

Efron, B. and Gong, G. (1983), “A leisurely look at the bootstrap, the jackknife, and cross-validation,” *The American Statistician*, 37, 36–48.

Friedman, J., Hastie, T., and Tibshirani, R. (2010), “Regularization Paths for Generalized Linear Models via Coordinate Descent,” *Journal of Statistical Software*, 33, 1–22.

Gastwirth, J., Krieger, A., and Rosenbaum, P. (1998), “Dual and simultaneous sensitivity analysis for matched pairs,” *Biometrika*, 85, 907–920.

Gelman, A., Jakulin, A., Pittau, M. G., and Su, Y.-S. (2008), “A weakly informative default prior distribution for logistic and other regression models,” *The Annals of Applied Statistics*, 1360–1383.

Gelman, A. and Su, Y.-S. (2015), *arm: Data Analysis Using Regression and Multilevel/Hierarchical Models*, r package version 1.8-5.

Greenland, S. (2003), “Quantifying biases in causal models: classical confounding vs. collider-stratification bias,” *Epidemiology*, 14, 300–6.

Hansen, B. (2007), “Optmatch: Flexible, optimal matching for observational studies,” *New Functions for Multivariate Analysis*, 18.

— (2011), “Propensity score matching to extract latent experiments from nonexperimental data: A case study,” *Looking Back*, 149–181.

Hansen, B. B. (2008), “The Prognostic Analogue of the Propensity Score,” *Biometrika*, 95, 481–488, doi: 10.1093/biomet/asn004.

Ho, D. E., Imai, K., King, G., and Stuart, E. A. (2007a), “Matching as Nonparametric Preprocessing for Reducing Model Dependence in Parametric Causal Inference,” *Political Analysis*, 15, 199–236.

— (2007b), “Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference,” *Political analysis*, 15, 199–236.

Hoerl, A. and Kennard, R. (1970), “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, 12, 55–67.

Hosman, C. A., Hansen, B. B., and Holland, P. W. (2010), “The sensitivity of linear regression coefficients’ confidence limits to the omission of a confounder,” *Annals of Applied Statistics*, 4, 849–870.

Iacus, M., S., King, Gary, Porro, and Giuseppe (2015), *cem: Coarsened Exact Matching*, r package version 1.1.17.

Iacus, S. M., King, G., and Porro, G. (2011), "Causal inference without balance checking: Coarsened exact matching," *Political analysis*, mpr013.

Kang, J. D. and Schafer, J. L. (2007), "Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data," *Statistical science*, 523–539.

Lee, B. K., Lessler, J., and Stuart, E. A. (2010), "Improving propensity score weighting using machine learning," *Statistics in medicine*, 29, 337–346.

Liaw, A. and Wiener, M. (2002), "Classification and Regression by randomForest," *R News*, 2, 18–22.

McCaffrey, D. F., Griffin, B. A., Almirall, D., Slaughter, M. E., Ramchand, R., and Burgette, L. F. (2013), "A tutorial on propensity score estimation for multiple treatments using generalized boosted models," *Statistics in medicine*, 32, 3388–3414.

Middleton, J. A. and Aronow, P. M. (2011), "Unbiased estimation of the average treatment effect in cluster-randomized experiments," *Available at SSRN 1803849*.

NCES (2013), "Common core of data (CCD) [electronic resource]," Accessed 10/15/2013.

Neyman, J. (1990), "On the application of probability theory to agricultural experiments. Essay on principles. Section 9," *Statistical Science*, 5, 463–480, transl. by D.M. Dabrowska and T.P. Speed from 1923 Polish original.

Pane, J. F., Griffin, B. A., McCaffrey, D. F., and Karam, R. (2013), "Effectiveness of cognitive tutor algebra I at scale," *Educational Evaluation and Policy Analysis*, 0162373713507480.

Pearl, J. (2009), "Remarks on the method of propensity score," *Statistics in Medicine*, 28, 1415–1416.

Pimentel, S. D., Small, D. S., and Rosenbaum, P. R. (2016), "Constructed second control groups and attenuation of unmeasured biases," *Journal of the American Statistical Association*, 111, 1157–1167.

Polley, E. and van der Laan, M. (2014), *SuperLearner: Super Learner Prediction*, r package version 2.0-15.

Rosenbaum, P. (2002a), "Covariance adjustment in randomized experiments and observational studies," *Statistical Science*, 17, 286–327.

Rosenbaum, P. and Rubin, D. (1983), "The central role of the propensity score in observational studies for causal effects," *Biometrika*, 70, 41–55.

Rosenbaum, P. R. (1991), "A Characterization of Optimal Designs for Observational Studies," *Journal of the Royal Statistical Society*, 53, 597– 610.

— (2002b), *Observational Studies*, Springer-Verlag, 2nd ed.

— (2010), *Design of observational studies*, Springer New York.

Rubin, D. B. (1973), "The Use of Matched Sampling and Regression Adjustment to Remove Bias in Observational Studies," *Biometrics*, 29, 185–203.

— (1974), "Estimating the Causal Effects of Treatments in Randomized and Nonrandomized Studies," *J. Educ. Psych.*, 66, 688–701.

— (1980), "Bias reduction using Mahalanobis-metric matching," *Biometrics*, 293–298.

— (2008), "For objective causal inference, design trumps analysis," *Annals of Applied Statistics*, 2, 808–40.

Rubin, D. B. and Thomas, N. (1996), "Matching Using Estimated Propensity Scores: Relating Theory to Practice," *Biometrics*, 52, 249–64.

Sekhon, J. S. (2011), “Multivariate and Propensity Score Matching Software with Automated Balance Optimization: The Matching Package for R,” *Journal of Statistical Software*, 42, 1–52.

Sjölander, A. (2009), “Propensity scores and M-structures,” *Statistics in medicine*, 28, 1416–1420.

Steiner, P. M., Cook, T. D., Li, W., and Clark, M. (2015), “Bias reduction in quasi-experiments with little selection theory but many covariates,” *Journal of Research on Educational Effectiveness*, 8, 552–576.

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

Varadhan, R. (2008), “[R] how to randomly generate a n by n positive definite matrix in R,” Accessed: 2015-05-16.

Venables, W. N. and Ripley, B. D. (2002), *Modern Applied Statistics with S*, New York: Springer, 4th ed., iSBN 0-387-95457-0.

Zorn, C. (2005), “A solution to separation in binary response models,” *Political Analysis*, 13, 157–170.

Zubizarreta, J. R., Cerdá, M., and Rosenbaum, P. R. (2013), “Effect of the 2010 Chilean Earthquake on Posttraumatic Stress Reducing Sensitivity to Unmeasured Bias Through Study Design,” *Epidemiology (Cambridge, Mass.)*, 24, 79.

8 Appendix: Proofs of Propositions 3 and 4

8.1 The Bias of τ_M

Lemma 1. *In a matching design where the target of estimation is τ_{ETT}, the bias of matching estimator (3) is*

$$\mathbb{E}[\hat{\tau}_M(\boldsymbol{Y})] - \tau_{ETT} = \sum_m w_m \boldsymbol{y}_{Cm}^T(\frac{\boldsymbol{p}_m}{n_{Tm}} - \frac{\boldsymbol{1}-\boldsymbol{p}_m}{n_{Cm}})$$

where $\boldsymbol{y}_{Cm}$ is the vector of y_C values for all subjects for whom $m_i = m$: $\{y_{Ci}\}_{m_i=m}$, and $\boldsymbol{p}_m$ is a vector of probabilities of treatment assignment for subjects in m, given n_{Tm} and n_{Cm}: $P_i = Pr(Z_i = 1|n_{Tm}, n_{Cm})$.

Proof. All of the following expectations are taken conditional on $n_{C1}, ..., n_{CM}$ and $n_{T1}, ..., n_{TM}$.

$$\begin{aligned}
\mathbb{E}\hat{\tau}_M &= \mathbb{E}\sum_m w_m t_m(Y, Z) \\
&= \sum_m w_m \mathbb{E} t_m(Y, Z)
\end{aligned}$$

Next, for a particular match m,

$$\begin{aligned}
\mathbb{E}t_m(Y,Z) &= \mathbb{E}[\frac{1}{n_T}\boldsymbol{Y}_m^T\boldsymbol{Z}_m - \frac{1}{n_{Cm}}\boldsymbol{Y}_m^T(\boldsymbol{1}-\boldsymbol{Z}_m)] \\
&= \mathbb{E}[\frac{1}{n_T}\boldsymbol{y}_{\boldsymbol{Cm}}^T\boldsymbol{Z}_m - \frac{1}{n_{Cm}}\boldsymbol{y}_{\boldsymbol{Cm}}^T(\boldsymbol{1}-\boldsymbol{Z}_m)] + \mathbb{E}\frac{\boldsymbol{\tau}_m^T\boldsymbol{Z}_m}{n_{Tm}} \\
&= \boldsymbol{y}_{\boldsymbol{Cm}}^T\mathbb{E}[\frac{1}{n_{Tm}}\boldsymbol{Z}_m - \frac{1}{n_{Cm}}(\boldsymbol{1}-\boldsymbol{Z}_m)] + \frac{\boldsymbol{\tau}_m^T\mathbb{E}\boldsymbol{Z}_m}{n_{Tm}} \\
&= \boldsymbol{y}_{Cm}^T(\frac{\boldsymbol{p}_m}{n_{Tm}} - \frac{\boldsymbol{1}-\boldsymbol{p}_m}{n_{Cm}}) + \frac{\boldsymbol{\tau}_m^T\mathbb{E}\boldsymbol{Z}_m}{n_{Tm}}
\end{aligned}$$

Then note that $\sum_m w_m \frac{\boldsymbol{\tau}_m^T\mathbb{E}\boldsymbol{Z}_m}{n_{Tm}} = \tau_{ETT}$. □

8.2 Proof of Proposition 3

Proof. As in Lemma 1, the squared bias of $\hat{\tau}_{\text{rebar}}$ is

$$bias^2(\hat{\tau}_{\text{rebar}}) = \left[\sum_m w_m(\boldsymbol{y_{Cm}} - \boldsymbol{\hat{y}_{Cm}})^T \left(\frac{\boldsymbol{P_m}}{n_{Tm}} - \frac{\boldsymbol{1} - \boldsymbol{P_m}}{n_{Cm}}\right)\right]^2.$$

Let $\boldsymbol{y}_C$ and $\boldsymbol{\hat{y}}_C$ be length-n vectors, concatenations of $\boldsymbol{y}_{Cm}$ and $\boldsymbol{\hat{y}}_{Cm}$. For $i = 1, \cdots, n$ let $Q_i = w_{m_i}\left(\frac{P_i}{n_{Tm_i}} - \frac{1-P_i}{n_{Cm_i}}\right)$ and let $\boldsymbol{Q}$ be a concatenation of $\{Q_i\}$, a length-n vector. Since $0 \leq P_i \leq 1$, $|Q_i| \leq \max\left(\frac{1}{n_{Tm_i}}, \frac{1}{n_{Cm_i}}\right) w_{m_i}$. Then

$$\begin{aligned}
bias^2(\hat{\tau}_{\text{rebar}}) &= \left[(\boldsymbol{y_C} - \boldsymbol{\hat{y_C}})^T \boldsymbol{Q}\right]^2 \\
&\leq n\frac{||\boldsymbol{y_C} - \boldsymbol{\hat{y}_C}||^2}{n}||Q||^2 \text{ by Cauchy-Schwartz} \\
&\leq n\frac{||\boldsymbol{y_C} - \boldsymbol{\hat{y}_C}||^2}{n}\sum_i \max\left(\frac{1}{n_{Tm_i}}, \frac{1}{n_{Cm_i}}\right)^2 w_{m_i}^2 \\
&= \frac{||\boldsymbol{y_C} - \boldsymbol{\hat{y}_C}||^2}{n}\frac{n}{n_T^2}\sum_m (n_{Cm} + n_{Tm}) \max\left(1, \frac{n_{Tm}}{n_{Cm}}\right)^2
\end{aligned}$$

□

8.3 Proof of Proposition 4

Proof. The proof follows the form of the proof of Proposition 3, but exploits the fact that $Q_i \leq (\Gamma^{1/2} - 1)/(\Gamma^{1/2} + 1)/n_T$. This follows from two facts: first, in a matched pair design, if i is matched to j and $i \neq j$, $P_i = 1 - P_j$, so (7) can be re-written as $1/\Gamma \leq P_i^2/(1 - P_i)^2 \leq \Gamma$. Secondly, in a matched pair design, the term $P_i/n_{Tm_i} - P_j/n_{Cm_j}$ can be written as $2P_i - 1$. The result follows. □