arXiv:1911.03071v3 [stat.ME] 24 Feb 2021

Balancing covariates in randomized experiments with the Gram–Schmidt Walk design

Christopher Harshaw, Fredrik Sävje, Daniel A. Spielman, and Peng Zhang

Yale University

February 25, 2021

Abstract

The design of experiments involves a compromise between covariate balance and robustness. This paper introduces an experimental design that admits precise control over this trade-off. The design is specified by a parameter that bounds the worst-case mean square error of an estimator of the average treatment effect. Subject to the experimenter's desired level of robustness, the design aims to simultaneously balance all linear functions of the covariates. The achieved level of balance is considerably better than what a fully random assignment would produce, and it is close to optimal given the desired level of robustness. We show that the mean square error of the estimator is bounded by the minimum of the loss function of a ridge regression of the potential outcomes on the covariates. One may thus interpret the approach as regression adjustment by design. Finally, we provide non-asymptotic tail bounds for the estimator, which facilitate the construction of conservative confidence intervals.

Keywords: Causal inference, covariate balance, experimental design, robustness, treatment effects.

We thank Edo Airoldi, Peter Aronow, Nicholas Christakis, David Pollard, Cyrus Samii and Johan Ugander for helpful comments and discussions. We thank Akshay Ramachandran for allowing us to include his proof of Lemma A7 in Supplement A, which is shorter than our original proof. This work was supported in part by NSF Grant CCF-1562041, ONR Awards N00014-16-2374 and N00014-20-1-2335, a Simons Investigator Award to Daniel Spielman, and an NSF Graduate Research Fellowship (DGE1122492) awarded to Christopher Harshaw. Computing research infrastructure was supplied by the Yale Center for Research Computing.

1 Introduction

The act of randomly assigning treatments to units in an experiment ensures that the distribution of the assignments contains no systematic biases. A comparison of responses between treatment groups will, in expectation, capture the average causal effect without error. But randomization does not ensure that the comparison captures the true effect for any specific assignment. It does not even ensure that the error of such an estimate is small. An alternative to randomization is to make the treatment groups as similar as possible in terms of observable characteristics. The aim is to maximizes balance and ensure that the only difference between the groups at the time of assignment is treatment itself. If successful, such an assignment would minimize the estimation error. The concern is that unobserved characteristics, including potential outcomes, may not be balanced even if the observed covariates are.

An idea that goes back to at least Efron (1971) is that the design of an experiment involves a compromise between balance and randomization. Randomization does not balance observed covariates as well as a non-random assignment that specifically targets such balance. But randomization provides protection against imbalances on unobserved characteristics. We must weigh the robustness granted by randomness against the possible gains in precision granted by balancing prognostically important covariates. All experimental designs implicitly resolve this trade-off, giving more weight to either balance or robustness.

We begin our investigation in this paper by elucidating and extending Efron's compromise. We show that the eigensystem of the covariance matrix of the assignment vector determines the mean square error of a treatment effect estimator. The characterization does not require assumptions on the potential outcomes, other than their existence, and it is valid in finite samples for any experimental design. It is the alignment of the eigenvectors of the covariance matrix with the vector of the potential outcomes that determines the precision. If we have some idea about the direction of the potential outcome vector before assignment, we may choose a design whose eigenvectors are appropriately aligned, and thus improve precision. From this perspective, the role of covariates is to inform us about possible directions of this vector, which would allow us to align the eigenvectors in practice. However, such alignment can be harmful if the covariates are uninformative. Specifically, as the alignment of the covariance matrix becomes more skewed, the design becomes less robust in terms of worst-case precision.

Building on these insights, the main contribution of the paper is the introduction of the Gram–Schmidt Walk design. The design is specified by a parameter that precisely controls the trade-off between balance and robustness. Experimenters may select this parameter to maximize robustness, in which case the design is fully randomized. Or, experimenters may select it to maximize expected covariate balance, in which case the design introduces potentially strong dependencies between the assignments. All intermediate choices between

two extremes are possible as well. The design then introduces less dependence in order to provide more robustness, at the cost of achieving less covariate balance. We describe an efficient implementation of the design that runs in polynomial time and provably satisfies the robustness and balance properties. This allows experimenters to draw assignments from the Gram–Schmidt Walk design in practice.

We investigate the statistical properties of the design in finite samples. The first main result, which we present in Section 7, is that the design parameter controls the worst-case mean square error relative to the minimax optimal design. Experimenters who desire a certain level of robustness may set the parameter to match their preferences. The design then balances the covariates conditional on meeting the specified robustness guarantee. If the parameter is set to maximize robustness, the design is minimax optimal.

The second result, which we present in Section 8, is a bound on the spectral norm of the covariance matrix of the vector of covariate imbalances. This bounds the imbalance of any linear function of the covariates. The imbalance is smaller than that of the fully randomized design roughly by a factor of the sample size divided by the number of covariates. We show in Section 9 that when the design parameter is set to maximize covariate balance, the balance of produced by the design is optimal in the sense that it is computationally intractable to achieve better balance. Additionally, regardless of computational considerations, we show that no other design can uniformly produce much better balance for a given robustness level than the Gram–Schmidt Walk design. The design is in this sense close to optimal, and meaningful improvements are only possible if assumptions are imposed on the covariates or the potential outcomes. These balance results allow experimenters to set the parameter according to the anticipated strength of the association between the covariates and the potential outcomes.

The bound on the linear imbalance suggests that the design performs well when the potential outcomes can be reasonably approximated by a linear function of the covariates. Section 10 demonstrates this formally by characterizing in finite samples the mean square error resulting from the design. We show that the error is bounded by the minimum of a ridge regression loss function when the covariates are used to predict the potential outcomes. This regression is never actually run. Instead, the design balances the covariates in such a way so that the estimator behaves as if the regression was run before the experiment was conducted, using information on all potential outcomes. The design parameter determines the amount regularization in the implicit regression.

Finally, Section 11 presents a tight, non-asymptotic tail bound on the sampling distribution of the treatment effect estimator. The proof of the tail bound introduces a new backwards induction technique for establishing concentration of martingales, which may be of independent interest. Together with an estimator of the ridge loss mentioned above, the tail bound allows for the construction of confidence intervals that are valid in finite samples. The finite sample validity comes at the cost of additional conservativeness compared to con-

ventional interval estimators based on large sample approximations. When experimenters select the design parameter to primarily focus on covariate balance, the conservativeness can be excessive.

A key discovery facilitating results in this paper is a way to translate the experimental design problem to a new type of problem in algorithmic discrepancy. A central problem of discrepancy theory is to partition a collection of vectors into two sets so that the sum of the vectors in each set is similar (Spencer, 1985). This problem directly corresponds to finding a treatment assignment that minimizes covariate imbalances. However, algorithms for discrepancy minimization generally aim to produce a single partition, corresponding to a single assignment. Experimental designs based on such algorithms would essentially be deterministic and thus afford essentially no robustness.

Our design is based on the Gram–Schmidt Walk algorithm of Bansal, Dadush, Garg and Lovett (2019). The algorithm is uniquely suited to the experimental design problem because it produces a random partition that has low expected discrepancy in every direction. To control the trade-off between balance and robustness, we instruct the algorithm to balance augmented covariates. The augmentation consists of a set of artificial covariates that are orthogonal between the units. Such covariates cannot be balanced, and they act instead to make the partition produced by the algorithm closer to fully random. Our use of the Gram–Schmidt Walk algorithm for experimental design requires us to perform a careful analysis of the statistical guarantees the algorithm provides.

2 Previous work

2.1 Arguments for and against randomization

The idea that experimenters should seek the most balanced treatment assignment predates the idea that they should randomize. For example, in a review of the experimental methods of the day, Student (1923) did not mention randomization even as a possibility. In a later paper, Student (1938) explicitly argues that randomization often is harmful because random assignments can only make the treatment groups less comparable than what they would be under the most balanced assignment. His conclusion is that the only role for randomization is to select between assignments that are equally balanced. The same conclusion, in slightly different incarnations, has been reached several times after this (see, e.g., Kiefer, 1959; Taves, 1974; Harville, 1975; Bertsimas et al., 2015; Kasy, 2016; Deaton & Cartwright, 2018; Kallus, 2018).

The conventional argument against designs that solely focus on balance is that confidence statements are hard to construct when the assignments are not sufficiently random. The argument goes back to at least Fisher (1925, 1926). He highlights that we do not need to choose between the two extremes. We can partially restrict the randomization to avoid

the most troublesome imbalances, but we allow some imbalances to persist to maintain enough granularity in the distribution of the design to construct well-motivated interval estimators and hypothesis tests. The insight has inspired a large number of experimental designs which fall on the continuum between the fully randomized and the maximally balanced designs. Examples include the matched pair design (Greevy et al., 2004; Imai et al., 2009), various stratified designs (Fisher, 1935; Higgins et al., 2016) and rerandomization (Lock Morgan & Rubin, 2012; Li et al., 2018).

Another argument against designs that solely focus on balance is robustness. Fisher talks about robustness in terms of unbiasedness, but the argument extends to precision. Wu (1981) appears to be the first paper to explicitly discuss the connection between randomization and robustness in this extended sense. He shows that the fully randomized design minimizes the worst-case mean square error. The result has been extended in various directions (Li, 1983; Hooper, 1989; Kallus, 2018; Bai, 2019; Basse et al., 2019).

Wu's minimax result demonstrates that there is no room to seek balance when robustness is our only objective. We may, however, accept a less robust design if it provides balance along dimensions that we believe are important. This is the compromise between balance and robustness mentioned in the introduction. Balancing covariates will improve precision if they are prognostically important, but it can be harmful if they are not. The idea can be traced back to Efron (1971), whose concept of "accidental bias" is closely related to our concept of robustness. This work is developed and extended by Kapelner et al. (2020). Our spectral interpretation of the experimental design problem in Section 3 further extends Efron's idea.

The two arguments for randomization are complementary. We may prefer an intermediate design that only partially restricts the randomization both because it facilitates confidence statements and because it provides robustness. The implications of the two arguments are, however, somewhat different. The amount of randomness needed for confidence statements is generally less than what considerations about robustness would suggest. For this reason, the primary focus in this paper is the robustness argument. However, the confidence intervals we describe are based on the distribution of the design, so we use the randomization as the sole basis for our inferences in line with the second argument.

2.2 Related experimental designs and analyses

The experimental design that is closest to the one we describe in this paper is rerandomization, which is a class of designs based on rejection sampling using a balance criterion for acceptance. Lock Morgan & Rubin (2012) provide an analysis of this approach when the balance criterion is based on a Mahalanobis distance, effectively imposing a whitening transformation on the covariates. They investigate the improvement in precision achieved by the approach under an assumption of additive treatment effects and an assumption of

normally distributed covariates and potential outcomes. In this setting, they show that rerandomization monotonically improves precision in the estimator as the balance criterion becomes stricter. However, the assumptions they impose implicitly remove the trade-off between robustness and balance discussed above, and it may therefore be difficult for experimenters to judge the relevance of the results for practice. Rerandomization also suffers from computational difficulties: to decrease precision by a constant factor one must reject a number of samples that is exponential in the number of covariates.

Li et al. (2018) relax the assumptions imposed by Lock Morgan & Rubin (2012) at the cost of studying the estimator's asymptotic distribution. The analysis is still restricted to balance criteria based on Mahalanobis distances. The authors show that the estimator has a non-normal asymptotic distribution under rerandomization in this more generally setting, emphasizing the concerns with the assumptions in the previous analysis. While Li et al. greatly improve our understanding of the rerandomization design, their analysis considers a balance criterion that is asymptotically fixed, meaning that an acceptable assignment must attain a Mahalanobis distance below some fixed threshold no matter the sample size. However, the computational resources required to find an assignment that satisfies a certain acceptance criterion generally depends on the sample size, so experimenters would adjust the criterion as the sample grows. It may therefore be difficult for experimenters to judge the relevance of the asymptotic results.

Li et al. (2018) note that it remains an open problem how to select the acceptance criterion. Using rerandomization, the compromise between balance and robustness is manifested in that a stricter acceptance criterion results in more balance at the cost of less randomness. Reminiscent of the investigation in this paper, Kapelner et al. (2019) study how to navigate this trade-off when using rerandomization. In particular, the authors investigate how to optimally select the acceptance criterion given a desired level of robustness. They use a model in which treatment effects are additive, and they assume that the unexplained parts of the units' potential outcomes can be seen as independent and identically distributed draws from some superpopulation. They provide an algorithm that produces an acceptance criterion that performs well given that the unexplained parts of the units' responses are not in the tails of the distribution they are assumed to be drawn from.

The design described in this paper is also related to the design introduced by Krieger et al. (2019). The authors construct an algorithm that makes local changes to an assignment that is generated fully at random. The purpose is to produce a new assignment that is more balanced. They show that if the covariate dimension is small, the number of changes needed to reach a highly balanced assignment is typically also small, so the final assignment vector is similar to the one that was generated at random. They provide heuristic arguments that this could give us reason to interpret the final assignments as being essentially random. Using three measures of randomness, they show that this is true asymptotically for a modified version of their algorithm under a set of regularity conditions.

Compared to these previous analyses, the analysis in this paper does not require assumptions on the potential outcomes or the covariates. In particular, the analysis does not require the covariates or potential outcomes to be normally distributed, nor the units to be drawn from a superpopulation, nor the treatment effects to be additive. Moreover, the analysis does not require whitening transformations on the covariates, so it applies no matter what type of covariates experimenters want to balance. Still, the analysis of both the precision and tail behavior of the estimator is valid in finite samples, and does not rely on large sample approximations. We use asymptotic illustrations only for expositional purposes to highlight and simplify features of the finite sample results. Hence, the understanding of the behavior of the Gram–Schmidt Walk design is both more precise and more relevant to practice than the understanding of existing experimental designs. Additionally, unlike rejection sampling algorithms, sampling an assignment from the Gram–Schmidt Walk design is practical due to its computationally efficiency.

3 Randomized experiments

3.1 Preliminaries

We consider an experiment with n units indexed by $i \in [n]$ and two treatment conditions. For example, the units could be patients in a clinical trial and the treatments could be a drug under evaluation and an inert substance, acting as placebo. We denote the treatment assigned to unit i as $z_i \in \{\pm 1\}$, and we collect all assignments in a vector $\boldsymbol{z} = (z_1, \ldots, z_n)$. The assignments are potentially generated at random, and we refer to the distribution of the vector $\boldsymbol{z}$ as the *design* of the experiment. The design is the sole source of randomness under consideration. We will initially consider designs for which each unit is equally likely to be assigned to either treatment, so that $\Pr(z_i = 1) = 1/2$ for all $i \in [n]$. We consider designs with unequal assignment probabilities in Section 13.1.

The responses of unit i when assigned to the two treatments are denoted a_i and b_i, which are taken to be real numbers. We refer these quantities as *potential outcomes* (Neyman, 1923). To ensure that the potential outcomes are well-defined, the treatment conditions must be specified to a sufficiently detailed level and the treatment assigned to one unit may not affect the response of other units (Rubin, 1980). The observed outcome for unit i is

$$y_i = \begin{cases} a_i & \text{if } z_i = 1, \\ b_i & \text{if } z_i = -1. \end{cases}$$

It will prove helpful to collect the potential and observed outcomes in vectors:

$$\boldsymbol{a} = (a_1, \ldots, a_n), \qquad \boldsymbol{b} = (b_1, \ldots, b_n), \qquad \boldsymbol{y} = (y_1, \ldots, y_n).$$

A unit's treatment effect is the contrast between its potential outcomes: $a_i - b_i$. The quantity of interest in this paper is the *average treatment effect*:

$$\tau = \frac{1}{n}\sum_{i=1}^{n}(a_i - b_i).$$

Treatment effects are fundamentally unobservable because we can observe at most one potential outcome for each unit. We seek to estimate the average effect using the observed outcomes and our knowledge about the design. The focus in this paper is the Horvitz–Thompson estimator (Narain, 1951; Horvitz & Thompson, 1952). For designs with $\Pr(z_i = 1) = 1/2$ for all units, the estimator can be written as

$$\widehat{\tau} = \frac{1}{n}\sum_{i\in Z^+}\frac{y_i}{0.5} - \frac{1}{n}\sum_{i\in Z^-}\frac{y_i}{0.5} = \frac{2}{n}\langle \boldsymbol{z}, \boldsymbol{y}\rangle,$$

where $Z^+ = \{i \in [n] : z_i = 1\}$ and $Z^- = \{i \in [n] : z_i = -1\}$ are the two treatment groups.

The quality of our inferences depends on the design of the experiment. The Horvitz–Thompson estimator is known to be unbiased and consistent for the average treatment effect for a large number of designs (see, e.g., Aronow & Middleton, 2013; Delevoye & Sävje, 2020). The goal is to pick one of these designs to ensure that the estimator is close to the average treatment effect also in finite samples. This is achieved if the design balances the potential outcomes, so that the mean of the potential outcomes in each treatment group tends to be close to the mean in the overall sample.

3.2 Spectral interpretation of experimental designs

To make the task ahead concrete, we will use the mean square error of the estimator as a measure of its performance. This error can be written as a quadratic form with respect to the covariance matrix of the assignment vector of the average of the two potential outcome vectors.

Lemma 1. *For any experimental design with* $\Pr(z_i = 1) = 1/2$ *for all* $i \in [n]$*, the mean square error of the Horvitz–Thompson estimator is*

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4}{n^2}\boldsymbol{\mu}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{\mu} \qquad \textit{where} \qquad \boldsymbol{\mu} = \frac{\boldsymbol{a} + \boldsymbol{b}}{2}.$$

The quadratic form demonstrates that the spectral properties of the covariance matrix are key to understanding the mean squared error. In particular, the eigenvectors and eigenvalues of the covariance matrix provide a basis which diagonalizes the matrix, thereby characterizing the quadratic form and the mean squared error. To see this, let $\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_n$ be

the normalized eigenvectors of $\mathrm{Cov}(\boldsymbol{z})$ with corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$. Because covariance matrices are symmetric, the eigenvectors form an orthonormal basis for the n-dimensional vector space. We can therefore write any vector $\boldsymbol{\mu}$ in $\mathbb{R}^n$ as a scaled linear combination of the eigenvectors:

$$\boldsymbol{\mu} = \|\boldsymbol{\mu}\| \sum_{i=1}^{n} w_i \boldsymbol{\eta}_i, \tag{1}$$

where $w_1, \ldots, w_n$ are the coefficients of the combination. Each coefficient w_i captures the alignment of the potential outcome vector with the corresponding eigenvector $\boldsymbol{\eta}_i$. In particular, we have that $w_i = \langle \boldsymbol{\mu}, \boldsymbol{\eta}_i \rangle / \|\boldsymbol{\mu}\|$.

We use equation (1) to expand the potential outcome vector $\boldsymbol{\mu}$ in the expression for the mean square error in Lemma 1. Because the eigenvectors form an orthonormal basis, we obtain

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4M}{n} \sum_{i=1}^{n} w_i^2 \lambda_i, \qquad \text{where} \qquad M = \frac{1}{n}\|\boldsymbol{\mu}\|^2 = \frac{1}{n}\sum_{i=1}^{n} \mu_i^2$$

is the second moment of the average of the potential outcomes. The decomposition shows that mean square error of a design is proportional to a convex combination of the eigenvalues of the covariance matrix of its assignment vector. The combination is convex because the squared coefficients $w_1^2, \ldots, w_n^2$ are non-negative and sum to one. Recall that these coefficients reflect the alignment of the potential outcome vector with the corresponding eigenvectors of the covariance matrix.

Representing the mean square error as a weighted average of eigenvalues provides several useful insights. One insight is that the relative performance of different designs does not depend on the magnitude of the potential outcomes. The mean square error will increase as the second moment M grows larger, but it will do so proportionally for all designs. The performance of a design is instead determined by the direction of the vector $\boldsymbol{\mu}$ relative to the eigenvectors of the covariance matrix that the design produces. To select a well-performing design, we should ensure that $\boldsymbol{\mu}$ is aligned to eigenvectors with small eigenvalues and nearly orthogonal to eigenvectors with large eigenvalues.

A second insight is that if the potential outcome vector is parallel to some eigenvector i, then $w_i = 1$, and the mean square error is determined by the corresponding eigenvalue. This gives us a simple way to characterize the worst- and best-case performance of a design over all possible potential outcomes. The worst-case is when $\boldsymbol{\mu}$ is parallel with the eigenvector corresponding to the largest eigenvalue, in which the case the mean square error is equal to that eigenvalue scaled by $4M/n$. Similarly, the best-case performance is determined by the smallest eigenvalue.

For example, the *group-balanced randomization design* divides the units into two treatment groups of fixed and equal size uniformly at random. This design is sometimes called

complete randomization, but we avoid that name because the assignments produced by the design are not independent, and assigning treatment independently at random is arguably more random. Indeed, the covariance matrix produced by the group-balanced design is $\text{Cov}(\boldsymbol{z}) = (n\boldsymbol{I} - \mathbf{1}\mathbf{1}^\intercal)/(n-1)$, where $\boldsymbol{I}$ is the identity matrix and $\mathbf{1}$ is a vector of ones. The smallest eigenvalue of this matrix is zero, corresponding to the eigenvector $\mathbf{1}$. Thus, the best-case mean squared error achieved by the design is zero. This occurs when the potential outcome vector is parallel with the vector of ones, so that all elements of $\boldsymbol{\mu}$ are the same. The largest eigenvalue is $n/(n-1)$, so the worst-case mean squared error is $4M/(n-1)$. This occurs when the potential outcome vector is orthogonal to the vector of ones, so that the sum of the elements of $\boldsymbol{\mu}$ is zero.

A third insight is that the sum of the eigenvalues is equal to n for all designs. This is because the sum is equal to the trace of the covariance matrix, and the diagonal elements of the matrix are fixed at one for designs with uniform first-order assignment probabilities. This highlights that it is not possible to improve the performance of a design in all directions simultaneously. One may improve performance for some vectors $\boldsymbol{\mu}$ only by making it worse for others.

An implication of the third insight is that there is an inherent trade-off between potential performance and robustness when designing an experiment. The most robust design is one that focuses equally on all possible directions of the potential outcome vector, so that the eigenvalues are uniform: $\lambda_1 = \cdots = \lambda_n = 1$. Such a design minimizes the worst-case performance because the mean square error is constant at $4M/n$ for all potential outcome vectors. This is achieved by any design for which the assignments are pair-wise independent, so that $\text{Cov}(\boldsymbol{z}) = \boldsymbol{I}$. A design with non-uniform eigenvalues will outperform the fully randomized design for some directions the potential outcome vector $\boldsymbol{\mu}$ could lie in. But by necessity, one or more eigenvalues will be larger than one, so the precision is worse in the corresponding directions. The more we skew the alignment of the covariance matrix, the better the design will perform if the vector $\boldsymbol{\mu}$ predominately lies in one of the targeted directions, but the precision worsens if it does not.

Another implication of the third insight is that all experimental designs are admissible. For any two designs with distinct covariance matrices, there exists potential outcome vectors so that either design outperforms the other. For this reason, searches for an universally optimal design will not be fruitful. Designs must instead be judged with respect to a particular potential outcome vector, or with respect to a particular set of such vectors.

3.3 Covariate balance

If we somehow knew the potential outcome vector $\boldsymbol{\mu}$, we would choose a design for which the smallest eigenvector of its covariance matrix is aligned with that vector, thereby making the mean square error as small as possible. In practice, however, the potential outcomes

are unknown, so this recipe cannot be followed.

If we are completely agnostic about the potential outcomes, designs with independent assignments appear to be the most appropriate choice because they put equal focus on all possible vectors. But prior substantive knowledge could motivate the use of other designs. We may, for example, know that the potential outcomes are positive. The group-balanced design will typically be a better choice in this case because a positive potential outcome vector is always somewhat aligned with the vector of ones, which is its smallest eigenvector.

A common type of prior knowledge is the existence of associations between the potential outcomes and auxiliary data observed before the experiment is run. Let $\boldsymbol{x}_i \in \mathbb{R}^d$ be a vector of d covariates for unit i, and let $\boldsymbol{X}$ be an n-by-d matrix whose rows are the covariate vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. If these covariates are predictive of the potential outcomes, we may use them as proxies for the outcomes. By balancing the covariates, we hope to indirectly balance potential outcomes as well. If we restrict our attention to linear associations, the spectral perspective provides motivation for this approach.

We refer to the column space of $\boldsymbol{X}$ as the span of the covariates. To say that that covariates are linearly predictive of the potential outcomes is the same as saying that the potential outcome vector is close to this span. Hence, if we know that such a linear association exists, we have some information about the direction of the potential outcomes. By aligning the smallest eigenvectors of the covariance matrix with the column space of $\boldsymbol{X}$, we make the corresponding quadratic form, and thus the mean square error, small for all potential outcomes $\boldsymbol{\mu}$ which are linearly predicted by the covariates.

To see this analytically, let $\boldsymbol{\beta} = \arg\min_{\boldsymbol{\beta}} \|\boldsymbol{\mu} - \boldsymbol{X}\boldsymbol{\beta}\|$ be a linear function of the covariates that best predicts the potential outcomes. If the covariates are linearly independent, this function is unique and given by $\boldsymbol{\beta} = (\boldsymbol{X}^\mathsf{T}\boldsymbol{X})^{-1}\boldsymbol{X}^\mathsf{T}\boldsymbol{\mu}$, but the argument applies also under linear dependence. In either case, the function $\boldsymbol{\beta}$ has no causal meaning, and it should be interpreted simply as describing a projection. Let $\widehat{\boldsymbol{\mu}} = \boldsymbol{X}\boldsymbol{\beta}$ be the predicted potential outcomes and let $\boldsymbol{\varepsilon} = \boldsymbol{\mu} - \widehat{\boldsymbol{\mu}}$ be the errors of the predictions.

We can now write $\boldsymbol{\mu} = \widehat{\boldsymbol{\mu}} + \boldsymbol{\varepsilon}$. This decomposes the potential outcomes into two projections: one onto to the span of the covariates and one onto the orthogonal complement of that span. The decomposition allows us to also decompose the mean square error:

$$\frac{n^2}{4}\,\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \widehat{\boldsymbol{\mu}}^\mathsf{T}\,\mathrm{Cov}(\boldsymbol{z})\widehat{\boldsymbol{\mu}} + \boldsymbol{\varepsilon}^\mathsf{T}\,\mathrm{Cov}(\boldsymbol{z})\boldsymbol{\varepsilon} + 2\widehat{\boldsymbol{\mu}}^\mathsf{T}\,\mathrm{Cov}(\boldsymbol{z})\boldsymbol{\varepsilon}.$$

It is not possible to directly minimize this expression because the potential outcomes are not observed, so $\widehat{\boldsymbol{\mu}}$ and $\boldsymbol{\varepsilon}$ are not known, but the decomposition provides guidance.

We know that $\widehat{\boldsymbol{\mu}}$ is in the span of the covariates. Thus, to make the first term small, we should seek to align the smallest eigenvectors of the covariance matrix with the columns of

$\boldsymbol{X}$. This ensures that

$$\widehat{\boldsymbol{\mu}}^\intercal \operatorname{Cov}(\boldsymbol{z})\widehat{\boldsymbol{\mu}} = \boldsymbol{\beta}^\intercal \boldsymbol{X}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{\beta}^\intercal \operatorname{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\boldsymbol{\beta}$$

is small compared to the magnitude of $\boldsymbol{\beta}$. Similarly, to make the second term small, we should seek to align the smallest eigenvectors with the orthogonal complement of the span.

The span of the covariates and its orthogonal complement make up the entire vector space, so we cannot simultaneously align the smallest eigenvectors of the covariance matrix to both of these subspaces. Without any prior knowledge about the potential outcomes, the appropriate choice of design would again be to focus equally on all directions. However, we now have some indication of the direction of the vector because we know, or presume to know, that the covariates are predictive of the potential outcomes. This implies that $\|\widehat{\boldsymbol{\mu}}\|^2$ is a large share of $\|\boldsymbol{\mu}\|^2$ relative to the number of dimensions of the covariate span. The implication is that we should focus on the first term and align the smallest eigenvectors with the columns of $\boldsymbol{X}$ to a larger degree than what the dimensionality of the span would suggest. However, such alignment could make the mean square error larger if, contrary to our perceived knowledge, $\|\boldsymbol{\varepsilon}\|^2$ is a disproportionately large share of $\|\boldsymbol{\mu}\|^2$.

The spectral interpretation does not tell us the weight we should give to balance and robustness when we design an experiment. This should be governed by the experimenter's preferences and prior substantive knowledge. But the interpretation highlights exactly what the trade-off consists of. To make the design robust, we should make the matrix $\operatorname{Cov}(\boldsymbol{z})$ small in all directions. By minimizing the spectral norm of this matrix, we minimize the worst-case mean square error of the design. To make the design perform well when the covariates are predictive of the potential outcomes, we should instead make the matrix $\operatorname{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ small. By minimizing the spectral norm of this second covariance matrix, we maximize the balance of all linear functions of the covariates. Unless the covariate vectors are orthogonal between the units, these two objectives are in opposition. Making the norm of one of the matrices smaller will generally make the norm of the other matrix larger.[1]

3.4 Realizable designs and drawing assignments

The spectral interpretation helps us understand the behavior of a particular design, but by itself, it does not help us construct a design. The difficulty is that covariance matrices $\operatorname{Cov}(\boldsymbol{z})$ by necessity belong to the convex hull of $\{\boldsymbol{v}\boldsymbol{v}^\intercal : \boldsymbol{v} \in \{\pm 1\}^n\}$, and some eigensystems are not realizable in this set of matrices. In other words, after an experimenter selects a desired covariance matrix that aligns the eigensystem in an appropriate way, it is often impossible to construct an experimental design that realizes that matrix. A related, more practical concern is that an efficient algorithm for drawing assignments from

[1]The objectives are, however, not in complete opposition, and the design that minimizes the spectral norm of $\operatorname{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ may still be quite random.

a specific design may not exist, because its support generally will contain exponentially many assignments. Experimental design is inherently both a statistical and computational problem.

There are currently three common approaches to solving this problem. The first is to impose ad hoc restrictions on the design in an effort to make it easy to analyze and computationally tractable to draw assignments. Implicitly, these restrictions also impose restrictions on the eigensystem in a way that we rarely have reason to believe is appropriate or useful. For example, the matched pair design restricts $\mathrm{Cov}(\boldsymbol{z})$ to be block diagonal with $n/2$ blocks. The second approach is rerandomization, which tackles the problem by rejection sampling. Besides being tremendously computationally inefficient, it is generally not clear how a particular acceptance criterion translates to a particular eigensystem, making the analysis of the design difficult without assumptions or large sample approximations. The final approach is to select a single assignment vector that approximately maximizes some measure of covariate balance. Provided that the sign of the vector is random, the resulting covariance matrix is rank one, telling us that this type of designs is minimally robust. For example, a naive application of algorithms in the discrepancy theory literature to minimize covariate imbalance would yield such a design.

The challenge ahead is to construct a design that allows experimenters to precisely control the trade-off between balance and robustness, and to precisely control which directions in the vector space to balance. At the same time, the design should be computationally tractable and amenable to statistical analysis without imposing assumptions or using large sample approximations.

4 The Gram–Schmidt Walk design

The Gram–Schmidt Walk design is specified by one parameter $\phi \in [0, 1]$ that determines how the trade-off between balance and robustness is resolved. When the parameter ϕ is closer to zero, the design places greater emphasis on covariate balance, at the expense of robustness. In particular, as ϕ decreases, the spectral norm of the matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ decreases so that better covariate balance is achieved. At the same time, the spectral norm of the matrix $\mathrm{Cov}(\boldsymbol{z})$ increases so the worst-case performance becomes worse.

The role of the parameter is to construct a matrix of *augmented covariates*. Using the raw covariate matrix $\boldsymbol{X}$ as input, an $(n + d)$-by-n matrix is constructed as

$$\boldsymbol{B} = \begin{bmatrix} \sqrt{\phi}\boldsymbol{I} \\ \xi^{-1}\sqrt{1-\phi}\boldsymbol{X}^\intercal \end{bmatrix},$$

where $\boldsymbol{I}$ is the n-by-n identity matrix and $\xi = \max_{i\in[n]} \|\boldsymbol{x}_i\|$ is the maximum row norm of the covariate matrix. The purpose of the factor ξ is to ensure that the two constituent matrices

are on comparable scales. The parameter ϕ controls the degree to which the augmented covariates resemble the raw covariates. When ϕ is zero, the augmented covariates coincide with the raw covariates, so balancing $\boldsymbol{B}$ is the same as making $\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$ small. When ϕ is one, the augmented covariates ignore the raw covariates, and we will see that balancing $\boldsymbol{B}$ is then the same as making $\mathrm{Cov}(\boldsymbol{z})$ small. Intermediate values interpolate between the two extremes.

We use the Gram–Schmidt Walk algorithm by Bansal et al. (2019) to balance the augmented covariates. The procedure was designed to algorithmically achieve the existence results of Banaszczyk (1998), which was an important open problem in algorithmic discrepancy theory (Matoušek, 1999; Chazelle, 2000). The algorithm partitions a set of high-dimensional vectors so that the discrepancy of the partition concentrates with high probability around zero. For our purposes, an important property of the algorithm is that its output approaches a fully random partition as its input vectors approach orthogonality.

An assignment vector is sampled from the design by calling a slightly modified version of the Gram–Schmidt Walk algorithm with the augmented covariate matrix $\boldsymbol{B}$ as input. The construction of the assignment vector will be essential to understand the properties of the design. We therefore describe our variant of the algorithm in detail. Our version differs from the original only in that we select the pivot units at random.

The assignment vector is constructed over several iterations. We extend our notation so that $\boldsymbol{z}_t$ denotes the assignment vector at iteration t, and $\boldsymbol{z}_t(i)$ denotes the ith coordinate at the same iteration.

1. Initialize a vector of *fractional assignments* $\boldsymbol{z}_1 \leftarrow (0, 0, \ldots, 0)$ and an index $t \leftarrow 1$.
2. Select a unit p uniformly at random from $[n]$. This is the initial *pivot* unit.
3. While $\boldsymbol{z}_t \notin \{\pm 1\}^n$:
   (a) Create the set $\mathcal{A} \leftarrow \{i \in [n] : |\boldsymbol{z}_t(i)| < 1\}$.
   (b) If $p \notin \mathcal{A}$, select a new pivot p from $\mathcal{A}$ uniformly at random.
   (c) Compute a *step direction* as

$$\boldsymbol{u}_t \leftarrow \operatorname*{arg\,min}_{\boldsymbol{u} \in U} \|\boldsymbol{B}\boldsymbol{u}\|,$$

   where U is the set of all $(u_1, \ldots, u_n) \in \mathbb{R}^n$ such that $u_p = 1$ and $u_i = 0$ for all $i \notin \mathcal{A}$.
   (d) Set $\delta^+ \leftarrow |\max \Delta|$ and $\delta^- \leftarrow |\min \Delta|$ where $\Delta = \{\delta \in \mathbb{R} : \boldsymbol{z}_t + \delta \boldsymbol{u}_t \in [-1, 1]^n\}$.
   (e) Select a *step size* at random according to

$$\delta_t \leftarrow \begin{cases} \delta^+ & \text{with probability } \delta^-/(\delta^+ + \delta^-), \\ -\delta^- & \text{with probability } \delta^+/(\delta^+ + \delta^-). \end{cases}$$

(f) Update the fractional assignments: $\boldsymbol{z}_{t+1} \leftarrow \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$.

(g) Increment the index: $t \leftarrow t + 1$.

4. Output $\boldsymbol{z}_t \in \{\pm 1\}^n$ as the assignment vector.

The goal of the algorithm is to produce assignments that tend to balance the difference between the within-group sums of the augmented covariate vectors. Unit i's augmented covariate vector is the ith column of $\boldsymbol{B}$. This is a scaled concatenation of the unit's raw covariates and a unit-unique indicator variable:

$$\boldsymbol{b}_i = \begin{bmatrix} \sqrt{\phi}\boldsymbol{e}_i \\ \xi^{-1}\sqrt{1-\phi}\boldsymbol{x}_i \end{bmatrix},$$

where $\boldsymbol{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ is the ith basis vector of dimension n. The difference between the within-group sums of the vectors is

$$\boldsymbol{B}\boldsymbol{z} = \sum_{i=1}^{n} z_i \boldsymbol{b}_i = \sum_{i \in Z^+} \boldsymbol{b}_i - \sum_{i \in Z^-} \boldsymbol{b}_i,$$

where, as above, Z^+ and Z^- are the two treatment groups.

The algorithm takes on this balancing problem using a relaxation. It extends the assignments from the integral values $\{\pm 1\}$ to the interval $[-1, 1]$. We refer to assignments in the interior of this interval as fractional. The algorithm constructs the assignments by iteratively updating a vector of fractional assignments $\boldsymbol{z}_t$. The initial fractional assignments are zero: $\boldsymbol{z}_1 = \boldsymbol{0}$. This means that the augmented covariate vectors start out perfectly balanced, because $\boldsymbol{B}\boldsymbol{z}_1 = \boldsymbol{B}\boldsymbol{0} = \boldsymbol{0}$. The initial assignments are not acceptable, however, because they are not integral. The only acceptable outputs are assignments $\boldsymbol{z}_t \in \{\pm 1\}^n$. As the algorithm updates the fractional assignments, the fundamental tension is between maintaining good balance, as measured by $\boldsymbol{B}\boldsymbol{z}_t$, and making the assignments integral. As we move towards integrality, balance becomes harder to maintain. The algorithm navigates this tension by updating the assignments in a direction that does not increase the imbalances too much while ensuring that the update is large enough to be a sizable step towards integrality.

The fractional assignments are updated by

$$\boldsymbol{z}_{t+1} \leftarrow \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t.$$

The update $\delta_t \boldsymbol{u}_t$ is comprised of a step size δ_t and a step direction $\boldsymbol{u}_t$. The algorithm selects the step direction to minimize the imbalance of the update as measured by the magnitude

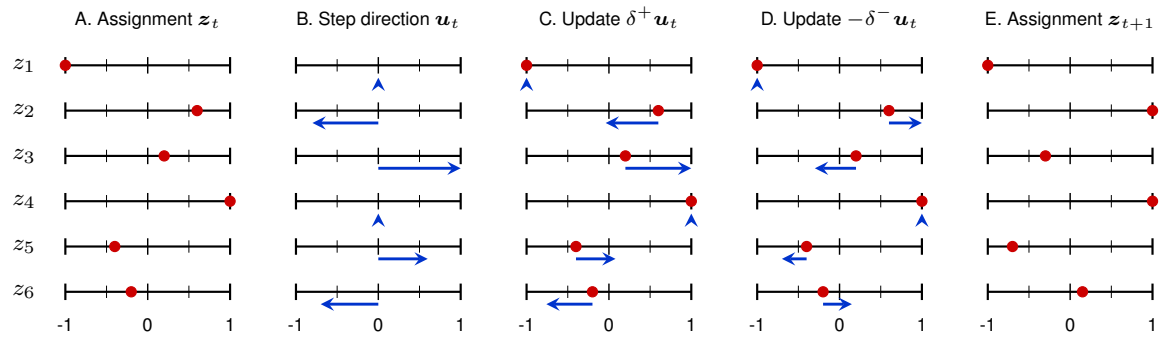

Figure 1: Illustration of one iteration of the Gram–Schmidt Walk design.

of the balance of the augmented covariate vectors:

$$\|\boldsymbol{B}\boldsymbol{u}_t\| = \left\|\sum_{i=1}^{n} u_i \boldsymbol{b}_i\right\|,$$

where we have suppressed the iteration index on the coordinates of $\boldsymbol{u}_t = (u_1, \ldots, u_n)$ to ease the notation. As the update is additive, we have

$$\boldsymbol{B}\boldsymbol{z}_{t+1} = \boldsymbol{B}\boldsymbol{z}_t + \delta_t \boldsymbol{B}\boldsymbol{u}_t,$$

so making $\|\boldsymbol{B}\boldsymbol{u}_t\|$ small helps keep $\|\boldsymbol{B}\boldsymbol{z}_{t+1}\|$ small.

The update direction is selected under two constraints. The first is that the coordinates corresponding to units that already have integral assignments are zero. That is, we impose $u_i = 0$ for all $i \notin \mathcal{A}$. The purpose is to ensure that these units maintain their integral assignments. The second constraint is that the coordinate for one unit $p \in \mathcal{A}$, which we call the pivot, is one: $u_p = 1$. The pivot fills two purposes. The first purpose is to avoid the trivial solution $\boldsymbol{u}_t = \boldsymbol{0}$. The second purpose, which we discuss in Section 6 below, is to avoid compounding imbalances in the updates.

With the step direction in hand, the algorithm randomly selects the step size δ_t to be one of two candidate values: δ^+ and δ^-. The candidate values, one positive and one negative, are the largest scaling factors δ_t such that the updated assignment vector $\boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$ is in the cube $[-1, 1]^n$. This ensures that the updated assignments are valid fractional assignments. It also ensures that at least one unit with an assignment in the interior of the interval is given an integral assignment at each iteration. The procedure is repeated until an integral assignment vector is reached.

Figure 1 provides an illustration of the algorithm. Panel A depicts the fractional assignments as an update iteration starts in the third step. Panel B depicts the selected update direction. This direction depends on the augmented covariates, which are not illustrated in

the figure. Panels C and D show the two possible updates given by the two candidate step sizes. Panel E depicts the randomly updated assignment vector at the end of the iteration.

5 Assignment probabilities and unbiasedness

In Section 3, we considered designs for which each unit is equally likely being assigned to either treatment. In this section, we prove that the Gram–Schmidt Walk design belongs to this class. To show this, we begin by demonstrating a martingale property of the assignment procedure.

Lemma 2. *The sequence of fractional assignments $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots$ forms a martingale.*

Proof. Recall that the fractional assignments are updated as $\boldsymbol{z}_{t+1} = \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$. Consider the conditional expectation of the assignments updated at iteration t:

$$\mathrm{E}[\boldsymbol{z}_{t+1} \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t] = \boldsymbol{z}_t + \mathrm{E}[\delta_t \boldsymbol{u}_t \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t].$$

By the law of iterated expectations,

$$\mathrm{E}[\delta_t \boldsymbol{u}_t \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t] = \mathrm{E}\big[\mathrm{E}[\delta_t \mid \delta_t^+, \delta_t^-]\boldsymbol{u}_t \;\big|\; \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t\big],$$

because δ_t is conditionally independent of $(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_t, \boldsymbol{u}_t)$ given (δ_t^+, δ_t^-). The step size δ_t takes the values δ_t^+ and δ_t^- with probabilities inversely proportional to their magnitudes, so

$$\mathrm{E}[\delta_t \mid \delta_t^+, \delta_t^-] = \delta_t^+ \left(\frac{\delta_t^-}{\delta_t^+ + \delta_t^-}\right) - \delta_t^- \left(\frac{\delta_t^+}{\delta_t^+ + \delta_t^-}\right) = 0.$$

It follows that the expected update is zero: $\mathrm{E}[\delta_t \boldsymbol{u}_t \mid \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t] = \boldsymbol{0}$. □

The martingale property implies that the expectation of the integral assignments sampled from the design is zero: $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{z}_1 = \boldsymbol{0}$. This yields the following corollaries.

Corollary 1. *Under the Gram–Schmidt Walk design,* $\Pr(z_i = 1) = 1/2$ *for all* $i \in [n]$.

Corollary 2. *The Horvitz–Thompson estimator is unbiased for the average treatment effect under the Gram–Schmidt Walk design.*

The first corollary follows from that $\mathrm{E}[z_i] = \Pr(z_i = 1) - \Pr(z_i = -1) = 0$ for all units. The second corollary follows from the fact that the estimator is unbiased for all designs with uniform first-order assignment probabilities (Aronow & Middleton, 2013).

The relation $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{z}_1$ holds for any initial fractional assignments, which provides control over the first moment of the assignment vector. We use this insight to extend the design to non-uniform assignment probabilities in Section 13.1.

6 Balancing augmented covariates

While the augmented covariates themselves are of little interest to experimenters, the design's ability to balance these covariates will prove fundamental to understand its behavior and the properties of the estimator. An upper bound on the maximum imbalance of the augmented covariates allows us to characterize the precision of the design and its ability to balance the raw covariates. We derive this upper bound in this section.

For ease of interpretation, our focus in this section, as well as in the subsequent sections, is performance and balance in a mean square sense. It is possible to extend these results to statements about tail probabilities using the concentration result in Theorem 4.

Definition 1. The *Loewner order* is a partial order on symmetric matrices where $\boldsymbol{A} \preceq \boldsymbol{B}$ if $\boldsymbol{B} - \boldsymbol{A}$ is positive semidefinite. Equivalently, $\boldsymbol{A} \preceq \boldsymbol{B}$ if $\boldsymbol{v}^\intercal \boldsymbol{A} \boldsymbol{v} \leq \boldsymbol{v}^\intercal \boldsymbol{B} \boldsymbol{v}$ for all vectors $\boldsymbol{v}$.

Theorem 1. *Under the Gram–Schmidt Walk design, the covariance matrix of the vector of imbalances for the augmented covariates $\boldsymbol{B}\boldsymbol{z}$ is bounded in the Loewner order by the orthogonal projection onto the subspace spanned by the columns of $\boldsymbol{B}$:*

$$\operatorname{Cov}(\boldsymbol{B}\boldsymbol{z}) \preceq \boldsymbol{P} = \boldsymbol{B}\big(\boldsymbol{B}^\intercal \boldsymbol{B}\big)^{-1} \boldsymbol{B}^\intercal.$$

Sketch of proof. We will show that $\boldsymbol{v}^\intercal \operatorname{Cov}(\boldsymbol{B}\boldsymbol{z}) \boldsymbol{v} \leq \boldsymbol{v}^\intercal \boldsymbol{P} \boldsymbol{v}$ for all vectors $\boldsymbol{v} \in \mathbb{R}^{n+d}$. In Supplement A, we derive an expression for $\operatorname{Cov}(\boldsymbol{z})$ in terms of the step directions and sizes used by the algorithm in Section 4. This allows us to write the quadratic form as

$$\boldsymbol{v}^\intercal \operatorname{Cov}(\boldsymbol{B}\boldsymbol{z}) \boldsymbol{v} = \boldsymbol{v}^\intercal \operatorname{E}\left[\sum_{t=1}^{T} \delta_t^2 \boldsymbol{B}\boldsymbol{u}_t \boldsymbol{u}_t^\intercal \boldsymbol{B}^\intercal\right] \boldsymbol{v} = \operatorname{E}\left[\sum_{t=1}^{T} \delta_t^2 \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right],$$

where T is the final iteration of the algorithm. Note that T is random.

The first part of the proof is to rearrange the terms of this sum. To do so, we define a pivot phase S_i as the set of iterations t for which unit i was the pivot. A unit's pivot phase is random and it may be the empty set if the unit was assigned an integral assignment without being chosen as the pivot. We can now write

$$\operatorname{E}\left[\sum_{t=1}^{T} \delta_t^2 \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right] = \sum_{i=1}^{n} \operatorname{E}\left[\sum_{t \in S_i} \delta_t^2 \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right].$$

In the supplement, we show that the expected sum of the squared step sizes within a pivot phase is bounded by one. This is a consequence of the fact that the same unit is kept as pivot until it is given an integral assignment. Together with the fact that each column of $\boldsymbol{B}$ has norm of at most one, this allows us to bound the contribution of each pivot phase

to the overall quadratic form as

$$\mathrm{E}\left[\sum_{t\in S_i}\delta_t^2\langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right] \leq \mathrm{E}\big[\boldsymbol{v}^\mathsf{T}\boldsymbol{P}_i\boldsymbol{v}\big],$$

where $\boldsymbol{P}_i$ denotes the projection onto a subspace that contains the updates $\boldsymbol{B}\boldsymbol{u}_t$ generated in the pivot phase S_i.

Bansal et al. (2019) show that the updates $\boldsymbol{B}\boldsymbol{u}_t$ and $\boldsymbol{B}\boldsymbol{u}_s$ are orthogonal if the iterations t and s are in different pivot phases. In the supplement, we extend this result to show that the subspaces corresponding to different pivot phases are orthogonal and their union is the column space of $\boldsymbol{B}$, so that $\sum_{i=1}^n \boldsymbol{P}_i = \boldsymbol{P}$ with probability one. We conclude that

$$\sum_{i=1}^n \mathrm{E}\big[\boldsymbol{v}^\mathsf{T}\boldsymbol{P}_i\boldsymbol{v}\big] = \boldsymbol{v}^\mathsf{T}\,\mathrm{E}\left[\sum_{i=1}^n \boldsymbol{P}_i\right]\boldsymbol{v} = \boldsymbol{v}^\mathsf{T}\boldsymbol{P}\boldsymbol{v}. \qquad \square$$

We provide a detailed proof of the theorem in Supplement A. This proof interprets the procedure as implicitly constructing a random basis for the column space of $\boldsymbol{B}$. This reveals the connection between the Gram–Schmidt Walk and its namesake, the Gram–Schmidt orthogonalization procedure.

The projection matrix $\boldsymbol{P}$ is small: it has n eigenvalues that are one and d eigenvalues that are zero. Theorem 1 therefore demonstrates that the design, as intended, balances the augmented covariates. A more direct way to see this is to consider the variance of linear functions of the augmented covariates. For every $\boldsymbol{v} \in \mathbb{R}^{n+d}$,

$$\mathrm{Var}(\boldsymbol{v}^\mathsf{T}\boldsymbol{B}\boldsymbol{z}) = \boldsymbol{v}^\mathsf{T}\,\mathrm{Cov}(\boldsymbol{B}\boldsymbol{z})\boldsymbol{v} \leq \boldsymbol{v}^\mathsf{T}\boldsymbol{P}\boldsymbol{v} \leq \|\boldsymbol{v}\|^2,$$

where the inequalities follow from Theorem 1 and the fact that the identity matrix upper bounds the projection matrix in the Loewner order.

The utility of achieving balance on the augmented covariates may become clear when one considers the structure of the covariance matrix. This matrix can be written in block form as

$$\mathrm{Cov}(\boldsymbol{B}\boldsymbol{z}) = \begin{bmatrix} \phi\,\mathrm{Cov}(\boldsymbol{z}) & \xi^{-1}\sqrt{\phi(1-\phi)}\,\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z}, \boldsymbol{z})^\mathsf{T} \\ \xi^{-1}\sqrt{\phi(1-\phi)}\,\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z}, \boldsymbol{z}) & \xi^{-2}(1-\phi)\,\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z}) \end{bmatrix}.$$

The blocks are scaled versions of the terms in the mean square error decomposition in Section 3.3. Theorem 1 thus bounds the components that determine the behavior of the design. We explore the consequences of Theorem 1 in the subsequent sections.

7 Controlling the worst-case mean square error

We introduced the Gram–Schmidt Walk design in Section 4 by saying that it is specified by a parameter $\phi \in [0, 1]$ that controls the robustness of the design. This section proves this property. In particular, the following theorem demonstrates that the parameter limits how much worse the Gram–Schmidt Walk design can perform relative to the mean square error of the minimax optimal design.

Theorem 2. *The worst-case mean squared error under the Gram–Schmidt Walk design is upper bounded by the ratio between the minimax optimum and the design parameter. That is, for all potential outcome vectors $\boldsymbol{\mu} = (\boldsymbol{a} + \boldsymbol{b})/2$, all covariate matrices $\boldsymbol{X}$, and all parameter values $\phi \in (0, 1]$,*

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] \leq \frac{4M}{\phi n} \qquad \textit{where} \qquad M = \frac{1}{n}\sum_{i=1}^{n} \mu_i^2.$$

Proof. We concluded in Section 3.2 that the worst-case mean square error of any design is $4\lambda_{\boldsymbol{z}} M/n$ where $\lambda_{\boldsymbol{z}}$ is the largest eigenvalue of $\mathrm{Cov}(\boldsymbol{z})$. We can obtain a bound on this largest eigenvalue by using the matrix inequality in Theorem 1. The upper left n-by-n block of $\mathrm{Cov}(\boldsymbol{B}\boldsymbol{z})$ is $\phi\,\mathrm{Cov}(\boldsymbol{z})$. The corresponding block of the projection matrix $\boldsymbol{P}$ in Theorem 1 is $\phi\boldsymbol{Q}$ where

$$\boldsymbol{Q} = \big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^{\intercal}\big)^{-1}.$$

If $\boldsymbol{A} \preceq \boldsymbol{B}$, then any two principal submatrices corresponding to the same row and column set S satisfy the inequality $\boldsymbol{A}_S \preceq \boldsymbol{B}_S$. It follows that $\mathrm{Cov}(\boldsymbol{z}) \preceq \boldsymbol{Q}$, which in turn implies that the largest eigenvalue of $\mathrm{Cov}(\boldsymbol{z})$ is bounded by the largest eigenvalue of $\boldsymbol{Q}$. Lemma A9 in Supplement A shows that this eigenvalue is at most $1/\phi$. □

The theorem shows how the parameter ϕ controls the robustness of the design. Recall from Section 3.2 that the mean square error of the minimax optimal design is $4M/n$ where M is the second moment of the potential outcomes. We know that the worst-case performance of the Gram–Schmidt Walk design generally is worse than the minimax optimal design because it aims to balance the covariates, which the minimax design does not. The parameter limits how much worse the design may perform relative to minimax. In particular, the relative worst-case performance is bounded by $1/\phi$. For example, if we were to use $\phi = 4/5$, the mean square error under the Gram–Schmidt Walk design would be at most 25% higher than the error under the minimax design. Of course, our hope is that the design will perform considerably better than this. The point here is that the design guarantees a certain level of performance even when there is no association between the covariates and the outcomes, in which case balancing the covariates can be harmful.

Theorem 2 also allows us to characterize the asymptotic behavior of the estimator under the design. A sufficient condition for the estimator to converge in mean square to the average treatment effect is $M = o(\phi n)$. When the second moment is asymptotically bounded, the worst-case rate of convergence is $\sqrt{\phi n}$. Hence, if we do not let the parameter approach zero, the estimator is ensured to be root-n consistent. These convergence rates are the worst-case rates, so they may be conservative. But it is reassuring that the design performs well asymptotically no matter the association between the covariates and the potential outcomes. This includes high-dimensional regimes in which the number of covariates grows at a faster rate than the number of units.

8 Balancing covariates

8.1 Measuring covariate balance

We continue our investigation of the statistical properties of the design by characterizing its ability to balance the covariates. Our focus will be on the matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$. To see how this matrix captures covariate balance, observe that the vector $\boldsymbol{X}^\intercal \boldsymbol{z}$ is the difference between the sums of the covariate vectors within the two treatment groups:

$$\boldsymbol{X}^\intercal \boldsymbol{z} = \sum_{i=1}^{n} z_i \boldsymbol{x}_i = \sum_{i \in Z^+} \boldsymbol{x}_i - \sum_{i \in Z^-} \boldsymbol{x}_i.$$

The expectation of this vector is zero, $\mathrm{E}[\boldsymbol{X}^\intercal \boldsymbol{z}] = \boldsymbol{0}$, so the diagonal of $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ is the mean square covariate difference between the treatment groups for each covariate. A common metric of covariate imbalance is the mean square error of a treatment effect estimator when we substitute a covariate for the outcome (see, e.g., Li et al., 2018). The diagonal elements of $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$, rescaled with $4/n^2$, are exactly this imbalance metric. The off-diagonal elements of $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ capture cross-imbalances, which are important to consider because imbalances in different covariates could be correlated, meaning that the imbalances compound.

Considering the whole matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$, rather than only its diagonal elements, allow us to characterize the balance of any linear function of the covariates. Let $\boldsymbol{\theta} \in \mathbb{R}^d$ be some arbitrary linear function of the covariates. The imbalance on this linear function for a particular assignment $\boldsymbol{z}$ is given by

$$\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z} = \sum_{i \in Z^+} \boldsymbol{\theta}^\intercal \boldsymbol{x}_i - \sum_{i \in Z^-} \boldsymbol{\theta}^\intercal \boldsymbol{x}_i,$$

and the expected imbalance of the function, in terms of mean square differences, is

$$\mathrm{E}\big[(\boldsymbol{\theta}^\mathsf{T}\boldsymbol{X}^\mathsf{T}\boldsymbol{z})^2\big] = \mathrm{Var}(\boldsymbol{\theta}^\mathsf{T}\boldsymbol{X}^\mathsf{T}\boldsymbol{z}) = \boldsymbol{\theta}^\mathsf{T}\,\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})\boldsymbol{\theta}.$$

In other words, imbalance of the linear function is intimately linked to the imbalance of $\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$. To ease the interpretation, we will translate our results on the balance of $\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$ to the corresponding results on an arbitrary linear function $\boldsymbol{\theta}$ of the covariates throughout this section.

8.2 Bounding covariate imbalances

The following lemma characterizes the covariate balance using the result in Theorem 1.

Lemma 3. *Under the Gram–Schmidt Walk design, the covariance matrix capturing covariate imbalances is bounded in the Loewner order by*

$$\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z}) \preceq \frac{\xi^2}{1-\phi}\boldsymbol{H} \qquad \textit{where} \qquad \boldsymbol{H} = \boldsymbol{X}^\mathsf{T}\left(\boldsymbol{X}\boldsymbol{X}^\mathsf{T} + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}\right)^{-1}\boldsymbol{X}.$$

Proof. The proof follows a similar structure as the proof of Theorem 2, in that we also here extract the principal submatrices from the matrix inequality in Theorem 1. The lower right d-by-d block of $\mathrm{Cov}(\boldsymbol{B}\boldsymbol{z})$ is $\xi^{-2}(1-\phi)\,\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$. The corresponding d-by-d block of the matrix bound $\boldsymbol{P} = \boldsymbol{B}(\boldsymbol{B}^\mathsf{T}\boldsymbol{B})^{-1}\boldsymbol{B}^\mathsf{T}$ is

$$\xi^{-2}(1-\phi)\boldsymbol{X}^\mathsf{T}\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}\big)^{-1}\boldsymbol{X}.$$

Thus, one obtains the lemma by dividing both sides by $\xi^{-2}(1-\phi)$ and rearranging the right hand side expression. □

The matrix $\boldsymbol{H}$ in Lemma 3 resembles the "hat matrix" often considered in the linear regression analysis. But there are two important differences. First, the conventional hat matrix is a projection onto the subspace spanned by the columns of the covariate matrix. The current matrix is instead a projection onto the subspace spanned by the rows. That is, $\boldsymbol{H}$ captures the leverages of the covariates rather than the leverages of the units. Second, $\boldsymbol{H}$ uses $\boldsymbol{X}\boldsymbol{X}^\mathsf{T} + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}$ for the normalization in the projection rather than the conventional $\boldsymbol{X}^\mathsf{T}\boldsymbol{X}$. We can interpret this normalization as a type of regularization. Indeed, $\boldsymbol{H}$ is the hat matrix for a ridge regression with $\xi^2\phi(1-\phi)^{-1}$ as the regularization parameter and $\boldsymbol{X}^\mathsf{T}$ as the regressors.

Similar to conventional leverage scores, we can interpret the diagonal of $\boldsymbol{H}$ as measures of how easy it is to recreate a column of $\boldsymbol{X}$ using linear combinations of the other columns. In this sense, they are measures of outlierness of the covariates. The implication is that

outlying covariates will be balanced to a lesser degree than non-outlying covariates. More generally, functions of the covariates that give more weight to covariates that are outliers will be balanced to a lesser degree. One intuition for this is that balancing a non-outlying covariate will tend to improve balance also for other non-outliers. There is, however, a tension between balance on non-outliers and outliers, and less focus will be given to the outliers because of the lack of complementarities.

Through Lemma 3, we immediately obtain a bound on the imbalance of linear functions of the covariates.

Corollary 3. *Under the Gram–Schmidt Walk design, the imbalance of any linear function* $\boldsymbol{\theta}$ *of the covariates is bounded by*

$$\mathrm{E}\big[(\boldsymbol{\theta}^\mathsf{T}\boldsymbol{X}^\mathsf{T}\boldsymbol{z})^2\big] \leq \frac{\xi^2}{1-\phi}\boldsymbol{\theta}^\mathsf{T}\boldsymbol{H}\boldsymbol{\theta}.$$

The corollary accounts for the distribution of the covariates among the units in the sample through the matrix $\boldsymbol{H}$. It is therefore adaptive; if some linear functions are balanced to a greater degree than others, perhaps because they give more weight to non-outlying covariates, the bound will reflect this. The bound does not involve any outcomes, so it may also be computed before the experiment is run. Experimenters may thereby inspect these bounds and select the design parameter according to their desired level of balance.

8.3 Worst-case imbalance on linear functions

The bound in Corollary 3 is different for each linear function $\boldsymbol{\theta}$. To make the bound more interpretable, we can take the supremum over all linear functions of a certain magnitude, which gives a bound on the worst-case imbalance. This allows us to bound the worst-case imbalance by a quantity that does not depend on the covariates other than through ξ.

Proposition 1. *The imbalance of any linear function* $\boldsymbol{\theta}$ *of the covariates is bounded by*

$$\mathrm{E}\big[(\boldsymbol{\theta}^\mathsf{T}\boldsymbol{X}^\mathsf{T}\boldsymbol{z})^2\big] \leq \frac{\xi^2}{1-\phi}\|\boldsymbol{\theta}\|^2.$$

Proof. By virtue of being a regularized hat matrix, all eigenvalues of $\boldsymbol{H}$ are at most one. This implies that $\boldsymbol{\theta}^\mathsf{T}\boldsymbol{H}\boldsymbol{\theta} \leq \|\boldsymbol{\theta}\|^2$ for all $\boldsymbol{\theta} \in \mathbb{R}^d$. □

The worst-case bound in Proposition 1 decreases monotonically with ϕ, indicating less imbalance as the parameter approaches zero. There can, however, be some linear functions whose expected imbalances are not monotone in the parameter. We will return to this point later in this section.

To better understand Proposition 1, we can compare this bound with the balance achieved under the fully randomized design. Let $\boldsymbol{z}_{\text{FR}}$ denote the assignment vector when the treatments are independent. We have $\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big] = \|\boldsymbol{X}\boldsymbol{\theta}\|^2$ for such a design. The following corollary considers this balance in relation to the balance achieved by the Gram–Schmidt Walk design.

Corollary 4. *Suppose that a sequence of covariate matrices $\boldsymbol{X}$ and linear functions $\boldsymbol{\theta}$ satisfies $\xi = \mathcal{O}(\sqrt{d\log(n)})$ and $\|\boldsymbol{\theta}\|^2/\|\boldsymbol{X}\boldsymbol{\theta}\|^2 = \mathcal{O}(1/n)$. Then the relative imbalance between the Gram–Schmidt Walk design and the fully randomized design is*

$$\frac{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big]}{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big]} = \mathcal{O}\bigg(\frac{d\log(n)}{(1-\phi)n}\bigg).$$

The first condition of the corollary stipulates that the set of covariates does not contain extreme outliers asymptotically. For example, the condition holds with a probability tending to one if the covariates were drawn from a subgaussian distribution. The second condition stipulates that we do not consider linear functions $\boldsymbol{\theta}$ that are trivially balanced by all designs. The simplest such trivial function is $\boldsymbol{\theta} = \mathbf{0}$, because we then have $\boldsymbol{X}\boldsymbol{\theta} = \mathbf{0}$, and all design would be equally balanced. In a low-dimensional setting where d and $\boldsymbol{\theta}$ are fixed in the asymptotic sequence, the second condition reduces to $\|\boldsymbol{X}\boldsymbol{\theta}\|^2 = \Omega(n)$. This states that the covariates of the units that are added in the asymptotic sequence are not self-balancing with respect to the linear function under consideration, meaning that the magnitude of $\boldsymbol{\theta}^\intercal \boldsymbol{x}_i$ tends to be bounded away from zero. When $\boldsymbol{\theta}$ is not fixed asymptotically, the sequence of functions is trivial if $\|\boldsymbol{X}\boldsymbol{\theta}\|^2$ is small relative to the magnitude of the functions.

Corollary 4 tells us that as long as we do not let ϕ approach one asymptotically, meaning that we instruct the design to at least partially focus on achieving balance, the relative imbalance between the Gram–Schmidt Walk design and the fully randomized design is on the order of the number of units divided by the number of covariates, $\mathcal{O}(d/n)$, disregarding the logarithmic factor. In a low-dimensional setting where d is bounded, this means that the rate of convergence of the imbalance of the function $\boldsymbol{\theta}$ improves from $\sqrt{n}$ under the fully randomized design to n under the Gram–Schmidt Walk design.

8.4 Characterizing the covariate imbalance

The worst-case bound in Proposition 1 is uninformative when ϕ is close to one. This is because it ignores that greater regularization imposed by the parameter ϕ will make the matrix $\boldsymbol{H}$ smaller. Using a singular value decomposition, we can write the matrix $\boldsymbol{H}$ in a way that better characterizes the covariate imbalance than the previous section, at the same time as being more interpretable than the bounds in Section 8.2.

Proposition 2. *Under the Gram–Schmidt Walk design, the covariance matrix of* $\boldsymbol{X}^\intercal \boldsymbol{z}$ *is bounded in the Loewner order by*

$$\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z}) \preceq \left(\phi(\boldsymbol{X}^\intercal \boldsymbol{X})^\dagger + (1-\phi)(\xi^2 \boldsymbol{\Pi})^\dagger\right)^\dagger,$$

where $\boldsymbol{\Pi}$ *is the orthogonal projection onto the rows of the covariate matrix* $\boldsymbol{X}$ *and* $\boldsymbol{A}^\dagger$ *denotes the pseudo-inverse of* $\boldsymbol{A}$*.*

The proposition demonstrates that the bound on the covariance matrix in Lemma 3 is a weighted harmonic mean of two d-by-d matrices. The first is the Gram matrix $\boldsymbol{X}^\intercal \boldsymbol{X}$ of the covariate matrix. This is the value the covariance matrix takes when the assignments are pair-wise independent. The second matrix $\xi^2 \boldsymbol{\Pi}$ is a scaled version of the projection onto the span of the covariate vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. When the covariate vectors span the entire vector space, $\boldsymbol{\Pi}$ is the identity matrix; otherwise, we may interpret $\boldsymbol{\Pi}$ as being the identity matrix on the subspace containing the data. In particular, the matrix is the limit of the hat matrix in the previous bound as the parameter approaches zero: $\boldsymbol{\Pi} = \lim_{\phi \to 0} \boldsymbol{H}$.

The design parameter determines the weights of the harmonic mean between the two matrices. When $\phi = 1$, the bound is the Gram matrix, demonstrating that the design does no balancing of the covariates. When $\phi = 0$, the bound is the scaled projection matrix, demonstrating that the design maximally balances the covariates in the sense that the covariance matrix of assignment vectors $\mathrm{Cov}(\boldsymbol{z})$ is maximally aligned with the covariate span. Intermediate values interpolate between the two extremes.

The projection matrix is scaled by ξ^2 in the bound. No design can improve upon this scaling term without imposing restrictions on the covariates. The scaling accounts for outliers among the units. The balancing will be sensitive to such outliers because the design can only assign integral treatments to the units. Consider an experiment in which one unit has large $\|\boldsymbol{x}_i\|$ and the norms of other units' covariate vectors are considerably smaller. It will be hard to balance this set of units because the one outlying unit will always tilt the balance in the direction of the treatment group it is assigned to. This is captured by $\xi = \max_{i \in [n]} \|\boldsymbol{x}_i\|$.

Proposition 2 allows us to strengthen the bound in Proposition 1, perhaps at the cost of being less interpretable. As a result of the matrix inequality, the largest eigenvalue of matrix on the right hand side in the proposition is an upper bound on the largest eigenvalue of the covariance matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$. We can derive largest eigenvalue of the matrix on the right hand side exactly:

$$\left(\phi \lambda_G^{-1} + (1-\phi)\xi^{-2}\right)^{-1},$$

where λ_G is the largest eigenvalue of the Gram matrix $\boldsymbol{X}^\intercal \boldsymbol{X}$. Unlike the bound on the largest eigenvalue of $\boldsymbol{H}$ we used in the proof of Proposition 1, this representation incorporates the effect of the implicit regularization imposed by the parameter.

Corollary 5. *Under the Gram–Schmidt Walk design, the imbalance of any linear function* $\boldsymbol{\theta}$ *of the covariates is bounded by*

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] \leq \Big(\phi\lambda_G^{-1} + (1-\phi)\xi^{-2}\Big)^{-1}\|\boldsymbol{\theta}\|^2.$$

This bound mirrors the matrix bound in Proposition 2, in that it is a weighted harmonic mean between λ_G and ξ^2. At the extremes, when ϕ is either one or zero, the bound is λ_G and ξ^2, respectively. Intermediate values of ϕ interpolate between the two end points.

The interpolation is monotone: the bound decreases with ϕ. This is because $\lambda_G \geq \xi^2$. As above, this indicates that the imbalance for the worst-case linear function tends to decrease as the parameter approaches zero. However, unlike the previous bound, the bound in Corollary 5 shows that the magnitude of λ_G relative to ξ determines the slope of the decrease. The eigenvalue λ_G is typically considerably larger than the norm ξ, so the imbalance tends to decrease quickly with ϕ. To see this, let $k \in [n]$ be such that $\|\boldsymbol{x}_k\| = \xi = \max_{i\in[n]}\|\boldsymbol{x}_i\|$, and observe that

$$\lambda_G = \max_{\|\boldsymbol{\theta}\|\leq 1} \sum_{i=1}^{n} \langle \boldsymbol{x}_i, \boldsymbol{\theta}\rangle^2 \geq \max_{\|\boldsymbol{\theta}\|\leq 1} \langle \boldsymbol{x}_k, \boldsymbol{\theta}\rangle^2 = \|\boldsymbol{x}_k\|^2 = \xi^2.$$

The gap introduced by the inequality is large as long as there is not a unit with a covariate vector of disproportionately large norm that is nearly orthogonal to the other units. This again highlights the effect of outliers among units. If outlying units exist, integrality of the assignments makes it hard to balance the covariates. The fewer outliers there are, the larger λ_G will be relative to ξ^2, and the more balance can be achieved.

8.5 Balancing decorrelated covariates

Some experimenters prefer to normalize and decorrelate the covariates before designing their experiments. The covariates are then transformed so their sample means are zero and the sample covariance matrix is the identity matrix. This is sometimes called a Mahalanobis or whitening transformation. The interpretation of Proposition 2 is particularly straightforward in this case because the Gram matrix of covariates is in a scaled isotropic position: $\boldsymbol{X}^\intercal \boldsymbol{X} = n\boldsymbol{I}$. The bound thus becomes

$$\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z}) \preceq \Big(\phi n^{-1} + (1-\phi)\xi^{-2}\Big)^{-1}\boldsymbol{I}.$$

The interpolation remains monotone because $\xi^2 \leq n$ when the covariates are decorrelated. Indeed, as suggested by the discussion in the previous section, if there are not extreme outliers among the units, then ξ^2 is considerably smaller than n.

Corollary 5 provides an even sharper characterization of the imbalance of linear functions of the covariates in this setting. Because all eigenvalues of the Gram matrix are the same when the covariates are decorrelated, the worst-case bound over all linear functions coincides with the point-wise bound in Corollary 3.

Corollary 6. *When the covariates are decorrelated, so that* $\boldsymbol{X}^\mathsf{T}\boldsymbol{X} = n\boldsymbol{I}$*, the upper bounds on* $\mathrm{E}\big[(\boldsymbol{\theta}^\mathsf{T}\boldsymbol{X}^\mathsf{T}\boldsymbol{z})^2\big]$ *in Corollaries 3 and 5 coincide for all linear functions* $\boldsymbol{\theta} \in \mathbb{R}^d$*:*

$$\frac{\xi^2}{1-\phi}\boldsymbol{\theta}^\mathsf{T}\boldsymbol{H}\boldsymbol{\theta} = \Big(\phi n^{-1} + (1-\phi)\xi^{-2}\Big)^{-1}\|\boldsymbol{\theta}\|^2.$$

The corollary shows that the imbalance bound is point-wise monotone when the covariates are in isotropic position. This is in contrast to above where the bound was shown to be monotonically decreasing for the worst-case linear function. In addition to point-wise monotonicity, the relative decrease in the bound is the same for all functions. This property is related to the *equal percent variance reducing* property discussed by Lock Morgan & Rubin (2012).

Point-wise monotonicity does not hold for general covariate matrices. When the covariates are not in isotropic position, there may be functions that can be balanced only by introducing imbalances for some other functions. This is the case when there are outliers among the covariates as we discussed in Section 8.2. The implication is that some functions may become less balanced as ϕ decreases. The reason is that the added imbalance, which typically is small, allows for large reductions of the imbalances of other functions.

9 The balance–robustness frontier

We have shown that the Gram–Schmidt Walk design provides control over the balance–robustness trade-off and that it provides considerably better balance than the fully randomized design. In this section, we ask whether there exist other designs that better navigate the balance–robustness trade-off by providing either more balance at a given level of robustness or more robustness at a given level of balance. We find that no design will be able to uniformly perform much better than the Gram–Schmidt Walk design, and in this sense, the design is close to optimal.

We begin by demonstrating that achieving more covariate balance than that which is guaranteed by Gram–Schmidt Walk design with $\phi = 0$ is computationally intractable. Charikar et al. (2011) prove that, given an n-by-n matrix $\boldsymbol{X}$ with ± 1 entries, it is NP-hard to determine whether

$$\min_{\boldsymbol{z}\in\{\pm 1\}^n} \|\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\|^2 \geq c\,n^2 \quad \text{or} \quad \min_{\boldsymbol{z}\in\{\pm 1\}^n} \|\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\|^2 = 0,$$

where $c > 0$ is universal, but presently unspecified, constant. We compare this hardness result to the covariate balance guarantees we prove for the Gram–Schmidt Walk design with $\phi = 0$. The covariate balance guarantees of Proposition 2 imply that in this case,

$$\mathrm{E}\big[\|\boldsymbol{X}^\intercal \boldsymbol{z}\|^2\big] \leq \xi^2 \|\boldsymbol{\Pi} \boldsymbol{z}\|^2 \leq \xi^2 \|\boldsymbol{z}\|^2 = n^2,$$

where the second inequality follows from the fact that projection matrices are contractive. The final equality follows because $\boldsymbol{X}$ have ± 1 entries, so all rows have norm $\sqrt{n}$ and $\xi^2 = n$. Thus, improving the covariate balance by even a constant factor pushes up against the boundary of computational tractability. This demonstrates that no computationally feasible design can provide a significantly better guarantee on expected covariate balance without assumptions on the structure of the covariates.

Next, we turn our attention to how the Gram–Schmidt Walk design navigates the trade-off between robustness and covariate balance. Proposition 2 and Theorem 2 imply that for any value of the design parameter $\phi \in [0, 1]$, the assignment vector $\boldsymbol{z}$ drawn from the Gram–Schmidt Walk design satisfies the robustness and covariate balance properties

$$\|\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\| \leq \left(\phi \lambda_G^{-1} + (1 - \phi)\xi^{-2}\right)^{-1} \qquad \text{and} \qquad \|\mathrm{Cov}(\boldsymbol{z})\| \leq 1/\phi,$$

where $\|\cdot\|$ denotes the spectral norm and $\lambda_G = \|\boldsymbol{X}^\intercal \boldsymbol{X}\|$. There are families of covariate matrices $\boldsymbol{X}$ for which no design can do much better. We briefly describe one such example here and defer technical proofs and additional examples to Supplement A.

The covariate matrix in this example consists of groups of units whose covariate vectors are identical within groups and orthogonal between groups. There are d groups, each of an odd size $k \geq 3$, so that there are $n = dk$ units. Let $\boldsymbol{v}_1, \ldots \boldsymbol{v}_d$ be d-dimensional orthonormal vectors. All units in the ℓth group have the covariate vector $\boldsymbol{x}_i = \boldsymbol{v}_\ell$. Thus, the n-by-d covariate matrix $\boldsymbol{X}$ is obtained by stacking k copies of each of these orthonormal vectors. For any design with $\Pr(z_i = 1) = 1/2$ for all units, we prove in Supplement A that if

$$\|\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\| \leq \left(\phi \lambda_G^{-1} + (1 - \phi)\xi^{-2}\right)^{-1} \quad \text{then} \quad \|\mathrm{Cov}(\boldsymbol{z})\| \geq \frac{1 + k(1 - \phi)}{\phi + k(1 - \phi)}.$$

For values of the design parameter that are not close to zero, this demonstrates that the Gram–Schmidt Walk design is almost tight in exploring this trade-off. To see this, consider the value of group size $k = 3$ and design parameter $\phi = 1/2$. For these values, the Gram–Schmidt Walk design achieves the level of covariate balance $\|\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\| \leq 2/(\lambda_G^{-1} + \xi^{-2})$ while ensuring the robustness guarantee $\|\mathrm{Cov}(\boldsymbol{z})\| \leq 2$. Our example shows that any design which meets this level of balance must incur $\|\mathrm{Cov}(\boldsymbol{z})\| \geq 5/4$.

This example is not meant to be representative of experiments in general. Its purpose is instead to demonstrate that the trade-off between robustness and covariate balance guar-

anteed by the Gram–Schmidt Walk cannot be much improved without assumptions on the covariates or the potential outcomes.

10 Characterizing the mean square error

The ultimate goal is to improve the precision of the treatment effect estimator. The following theorem characterizes its mean square error under the Gram–Schmidt Walk design.

Theorem 3. *The mean squared error under the Gram–Schmidt Walk design is at most the minimum of the loss function of an implicit ridge regression of the average of the potential outcome vectors* $\boldsymbol{\mu} = (\boldsymbol{a} + \boldsymbol{b})/2$ *on the covariates:*

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] \leq \frac{4L}{n} \qquad \textit{where} \qquad L = \min_{\boldsymbol{\beta} \in \mathbb{R}^d} \left[\frac{1}{\phi n} \big\| \boldsymbol{\mu} - \boldsymbol{X}\boldsymbol{\beta} \big\|^2 + \frac{\xi^2}{(1-\phi)n} \big\| \boldsymbol{\beta} \big\|^2 \right].$$

Proof. We showed in the proof of Theorem 2 that

$$\mathrm{Cov}(\boldsymbol{z}) \preceq \boldsymbol{Q} = \Big(\phi \boldsymbol{I} + (1-\phi)\xi^{-2} \boldsymbol{X}\boldsymbol{X}^\intercal \Big)^{-1}.$$

Using the expression for the mean squared error derived in Lemma 1, we write

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4}{n^2} \boldsymbol{\mu}^\intercal \, \mathrm{Cov}(\boldsymbol{z}) \boldsymbol{\mu} \leq \frac{4}{n^2} \boldsymbol{\mu}^\intercal \boldsymbol{Q} \boldsymbol{\mu}.$$

Finally, Lemma A10 in Supplement A uses the Woodbury matrix identity to show that $L = \boldsymbol{\mu}^\intercal \boldsymbol{Q} \boldsymbol{\mu}/n$. □

The theorem refines the upper bound on mean square error presented in Theorem 2. It demonstrates that the mean square error depends on the design parameter ϕ and the degree to which the covariates are predictive of the potential outcomes. The predictiveness is captured by L. This is the optimal value of the loss function of a ridge regression of the potential outcomes on the covariates using $\xi^2\phi/(1-\phi)$ as the regularization parameter. The loss is scaled by the inverse of the design parameter to highlight the weighting between the terms.

The first term of the loss L captures how well a linear function $\boldsymbol{\beta}$ predicts the potential outcomes using the covariates. This term can be made small if the potential outcome vector is close to the span of the covariates. The second term captures the magnitude of the linear function. The factor ξ^2 puts this magnitude on a neutral scale so that the optimum is not affected by a rescaling of the covariates. The design parameter ϕ determines the trade-off between the two terms, assigning more focus to either finding a function that predicts the outcomes well or one that is of small magnitude.

The bound provides a deeper understanding of the behavior of the design. As $\phi \to 0$, the design prioritizes covariate balance. This is reflected in the loss L by the convergence of the optimal function $\boldsymbol{\beta}$ to the best linear predictor. However, a design with such an extreme focus on covariate balance would not be robust. The bound also reflects this: unless the covariates are perfectly predictive of the potential outcomes, the bound grows without limit as ϕ approaches zero.

We concluded in the previous section that the design achieves more balance on covariates that are not outliers, by which we mean covariates with small leverage scores with respect to the regularized hat matrix $\boldsymbol{H}$. Theorem 3 provides some intuition for this behavior. If a non-outlying covariate is strongly predictive of the potential outcomes, we may shift the loading in $\boldsymbol{\beta}$ of that covariate to other covariates without large costs in terms of predictiveness. However, a similar shift for an outlying covariate will result in considerably worse predictions because there are no other covariates that can act as substitutes. The bound captures this with the penalty in the second term. In the first example, the shift of the covariate loadings will make $\|\boldsymbol{\beta}\|$ small without making the first term much larger. In the second example, the only way to make $\|\boldsymbol{\beta}\|$ small is to make the first term large.

Another way to see this is to interpret the second term of the loss L is as a complexity penalty. If the magnitude of a function $\boldsymbol{\beta}$ is large relative to the magnitudes of the potential outcomes and covariates, then it can exploit weak associations that a simpler function would not be able to pick up. In this sense, the design implicitly balances linear functions of the covariates that are not too complex. The implication is that it performs well when the potential outcomes can be approximated by a relative simple linear function of the covariates.

The bound also helps us understand when the design is expected to perform poorly. Theorem 2 demonstrated that the worst-case mean square error is $4M/\phi n$. Theorem 3 shows that this worst-case occurs when $L = M/\phi = \|\boldsymbol{\mu}\|^2/\phi n$. By inspecting the definition of L, we see that this is when the minimizer of the loss is $\boldsymbol{\beta} = \mathbf{0}$. In other words, the worst-case is when the potential outcome vector is in the orthogonal complement of the covariate span so that the covariates are completely unpredictive.

We can relate the performance of the design to the performance of the minimax design. We showed in Section 3.2 that the mean square error for the minimax design is $4M/n$. Using the bound in Theorem 3, the relative precision is bounded by L/M. When scaled by ϕ to produce $\phi L/M \in (0, 1]$, the quantity can interpreted as a type of goodness-of-fit measure for the ridge regression similar to the coefficient of determination, often denoted R^2. When this measure is close to zero, the potential outcomes can be well-approximated by a simple linear function. The design is guaranteed to outperform the minimax design when this measure is smaller than ϕ, which happens exactly when L is smaller than M.

11 Tail bounds and confidence statements

11.1 Sharpened tail bound

The previous sections examined the behavior of the design and the estimator in a mean square sense. In this section, we extend the investigation to tail behavior. This provides an alternative and often sharper description of the properties of the design, and it facilitates an interval estimator of the average treatment effect with known coverage properties in finite samples.

Bansal et al. (2019) used the martingale inequality of Freedman (1975) to show that the Gram–Schmidt Walk algorithm produces assignments such that $\boldsymbol{Bz}$ is a subgaussian random vector with variance parameter $\sigma^2 \leq 40$. This result allows us to investigate the behavior of the design in terms tail probabilities. The concern is that tail bounds based on $\sigma^2 = 40$ will generally be too loose to be useful in a statistical context. Unless we are interested in the extreme ends of the tails, Chebyshev's inequality based on the mean square error results in Section 10 will generally be more informative.

A key contribution of this paper is to strengthen the analysis of the tail behavior of the Gram–Schmidt Walk algorithm. We develop a new proof technique for establishing martingale concentration that allows us to obtain a tight upper bound on the subgaussian variance parameter.

Theorem 4. *Under the Gram–Schmidt Walk design, the vector $\boldsymbol{Bz}$ is subgaussian with variance parameter $\sigma^2 = 1$:*

$$\mathrm{E}\big[\exp\big(\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle\big)\big] \leq \exp\big(\|\boldsymbol{v}\|^2/2\big) \qquad \textit{for all} \qquad \boldsymbol{v} \in \mathbb{R}^{n+d}.$$

Sketch of Proof. Recall the projection matrix $\boldsymbol{P} = \boldsymbol{B}\big(\boldsymbol{B}^\intercal \boldsymbol{B}\big)^{-1}\boldsymbol{B}^\intercal$ from Theorem 1. Because projection is a contractive operator, we have

$$\exp\big(\|\boldsymbol{Pv}\|^2/2\big) \leq \exp\big(\|\boldsymbol{v}\|^2/2\big) \qquad \text{for all} \qquad \boldsymbol{v} \in \mathbb{R}^{n+d}.$$

Therefore, to prove the theorem, it suffices to show that

$$\mathrm{E}\left[\exp\Big(\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle - \|\boldsymbol{Pv}\|^2/2\Big)\right] \leq 1.$$

Following the proof of Theorem 1, we decompose the assignment vector into its fractional updates and then group them according to pivot phases,

$$\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle = \sum_{t=1}^{T} \delta_t \langle \boldsymbol{Bu}_t, \boldsymbol{v}\rangle = \sum_{i=1}^{n} \sum_{t \in S_i} \delta_t \langle \boldsymbol{Bu}_t, \boldsymbol{v}\rangle.$$

Similarly, we decompose the projection $\boldsymbol{P}$ into the mutually orthogonal projections given by each pivot phase:

$$\|\boldsymbol{P}\boldsymbol{v}\|^2 = \left\|\sum_{i=1}^{n} \boldsymbol{P}_i \boldsymbol{v}\right\|^2 = \sum_{i=1}^{n} \|\boldsymbol{P}_i \boldsymbol{v}\|^2,$$

where, as in the proof of Theorem 1, $\boldsymbol{P}_i$ denotes the projection matrix onto the subspace corresponding to pivot phase i that contains the updates $\{\boldsymbol{B}\boldsymbol{u}_t : t \in S_i\}$.

We consider the difference D_i between the two decompositions separately for each potential pivot unit i:

$$D_i = \sum_{t \in S_i} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v} \rangle - \|\boldsymbol{P}_i \boldsymbol{v}\|^2/2.$$

This allows us to write

$$\mathrm{E}\left[\exp\Big(\langle \boldsymbol{B}\boldsymbol{z}, \boldsymbol{v} \rangle - \|\boldsymbol{P}\boldsymbol{v}\|^2/2\Big)\right] = \mathrm{E}\left[\exp\left(\sum_{i=1}^{n} D_i\right)\right] = \mathrm{E}\left[\prod_{i=1}^{n} \exp(D_i)\right].$$

If a unit is never chosen as the pivot, the corresponding pivot phase is empty and $D_i = 0$. We can therefore restrict the product to the units which at some point are pivots. For notational convenience in this proof sketch, suppose that the pivot units are $1, 2, \ldots, r$ and they are chosen as pivots in this order. We then have

$$\mathrm{E}\left[\prod_{i=1}^{n} \exp(D_i)\right] = \mathrm{E}\left[\prod_{i=1}^{r} \exp(D_i)\right].$$

Consider a pivot unit i, where $1 \le i \le r$. Let Δ_i denote all random decisions made by the algorithm up to and including when i is chosen as the pivot. This includes all randomly chosen step sizes in the pivot phases $1, \ldots, i-1$, but not the step sizes in phases $i, \ldots, r$. The key part of the argument, which we prove in Supplement A, is that

$$\mathrm{E}\big[\exp(D_i) \,\big|\, \Delta_i\big] \le 1.$$

This follows from the choice of the step sizes, the fact that a unit remains a pivot until it is assigned an integral assignment, and the fact that each column of $\boldsymbol{B}$ has norm at most one.

We can now prove the inequality by backward induction. Because Δ_r includes all random decisions before unit r was selected as pivot, the quantities $D_1, \ldots, D_{r-1}$ are not random conditional on Δ_r. Using the law of iterated expectation, we can write

$$\mathrm{E}\left[\prod_{i=1}^{r} \exp(D_i)\right] = \mathrm{E}\left[\mathrm{E}\big[\exp(D_r) \,\big|\, \Delta_r\big] \prod_{i=1}^{r-1} \exp(D_i)\right] \le \mathrm{E}\left[\prod_{i=1}^{r-1} \exp(D_i)\right].$$

The proof is completed by induction over the remaining $r - 1$ pivot phases. □

The central step in the proof is bounding the conditional expectation of the exponential quantity during a pivot phase. Previous proof techniques bound this quantity through Taylor series approximations, which necessarily incur a loss in approximation and result in overly conservative subgaussian constants. In contrast, our proof analyzes the expected exponential quantity directly by carefully considering the choice of step size and another backwards induction argument. In this way, we can obtain $\sigma^2 = 1$, which is tight. This proof technique may be of independent interest for studying martingale concentration more generally.

Theorem 4 shows that linear functions of the augmented covariates are well concentrated. Because the augmented covariates contain the original covariates, this implies concentration of the imbalance of any linear function of the covariates. This concentration becomes tighter as the design parameter ϕ decreases. The proof of this is analogous to the derivation of the covariate balance results in Section 8 using Theorem 1. However, in the interest of space, our focus in the rest of the section is concentration of the estimator and the construction of confidence intervals.

11.2 Confidence intervals

The sharpened tail bound allows us to show that the Horvitz–Thompson estimator is subgaussian as well. This yields an interval estimator for the average treatment effect. The following proposition and corollary provide the details.

Proposition 3. *Under the Gram–Schmidt Walk design, the mass of the tails of the sampling distribution of the Horvitz–Thompson estimator is bounded by*

$$\Pr\big(|\widehat{\tau} - \tau| \geq \gamma\big) \leq 2\exp\left(\frac{-\gamma^2 n}{8L}\right) \qquad \textit{for all} \qquad \gamma > 0.$$

Proof. We prove the bound for the upper tail. The proof for the lower tail is identical. For any $t > 0$, we have

$$\Pr\big(\widehat{\tau} - \tau \geq \gamma\big) \leq \exp(-t\gamma)\,\mathrm{E}\big[\exp\big(t(\widehat{\tau} - \tau)\big)\big].$$

This can be shown either as a consequence of Markov's inequality or from the exponential inequality $\mathbb{1}[x \geq 0] \leq \exp(tx)$.

Lemma A1 in Supplement A shows that $\widehat{\tau} - \tau = 2n^{-1}\langle \boldsymbol{z}, \boldsymbol{\mu}\rangle$. The columns of $\boldsymbol{B}$ are linearly independent by construction, so we can define a vector $\boldsymbol{v} = 2tn^{-1}\boldsymbol{B}\big(\boldsymbol{B}^\intercal\boldsymbol{B}\big)^{-1}\boldsymbol{\mu}$. This allows us to write

$$\mathrm{E}\big[\exp\big(t(\widehat{\tau} - \tau)\big)\big] = \mathrm{E}\big[\exp\big(2tn^{-1}\langle \boldsymbol{z}, \boldsymbol{\mu}\rangle\big)\big] = \mathrm{E}\big[\exp\big(\langle \boldsymbol{B}\boldsymbol{z}, \boldsymbol{v}\rangle\big)\big].$$

Theorem 4 upper bounds the right-hand side by $\exp(\|\boldsymbol{v}\|^2/2)$. For the current choice of $\boldsymbol{v}$, the squared norm simplifies to

$$\|\boldsymbol{v}\|^2 = \frac{4t^2}{n^2}\boldsymbol{\mu}^\intercal\big(\boldsymbol{B}^\intercal\boldsymbol{B}\big)^{-1}\boldsymbol{\mu} = \frac{4t^2 L}{n},$$

where the final equality follows from Lemma A10 in Supplement A. Taken together, we obtain

$$\Pr\big(\widehat{\tau} - \tau \geq \gamma\big) \leq \exp\bigg(\frac{2t^2 L}{n} - t\gamma\bigg).$$

The proof is completed by setting $t = \gamma n/4L$. □

Corollary 7. *The random interval centered at $\widehat{\tau}$ with radius $\gamma_\alpha = \sqrt{8\log(2/\alpha)L/n}$ is a valid $(1-\alpha)$-confidence interval:*

$$\Pr\big(\widehat{\tau} - \gamma_\alpha \leq \tau \leq \widehat{\tau} + \gamma_\alpha\big) \geq 1 - \alpha.$$

The corollary illustrates the usefulness of the sharpened tail bound in Theorem 4. Confidence intervals based on the tail bound in Bansal et al. (2019) would be $\sqrt{40} \approx 6.3$ times wider than the intervals in Corollary 7.

A comparison between the intervals in Corollary 7 and conventional intervals is more intricate. One aspect is that our intervals do not rely on asymptotic approximations. This makes them particularly useful in experiments with small samples because large sample approximations may then not be appropriate. However, this comes at the cost of potentially wider intervals. For example, a common approach is to approximate the sampling distribution with a normal distribution. Using the variance bound in Theorem 3, such an approach would suggest intervals with radius $\Phi^{-1}\big(1-\alpha/2\big)\sqrt{4L/n}$ where $\Phi^{-1}\colon [0,1] \to \mathbb{R}$ is the quantile function of the standard normal deviate. Hence, for confidence levels 95% and 99%, the intervals in Corollary 7 would be about 1.39 and 1.26 times wider than those based on a normal approximation.

It remains an open question whether the sampling distribution of the Horvitz–Thompson estimator approaches a normal distribution under the Gram–Schmidt Walk design. Li et al. (2018) show that rerandomization does not yield estimators that are asymptotically normal. The Gram–Schmidt Walk design resembles rerandomization in some aspects, but it does not truncate the distribution of the design in the way rerandomization does. Based on our simulation results, we conjecture that the estimator is asymptotically normal under the design. However, until this has been shown formally, experimenters should show caution using a normal approximation even when the number of units is large.

Another aspect to consider is that intervals in Corollary 7 are implicitly based on the variance bound $4L/n$ in Theorem 3. It is common that confidence intervals for causal quantities are based on variance bounds, but the purpose of those bounds is different from

the purpose of the variance bound here. In particular, as we discuss in the next section, the conventional variance bounds address the fact that we cannot simultaneously observe all potential outcomes, so we cannot construct an unbiased estimator of the variance. The variance bound in Theorem 3 instead concerns the predictiveness of the covariates with respect to the potential outcomes.

As an illustration, consider confidence intervals based on Chebyshev's inequality. Using the variance bound in Theorem 3, this inequality would suggest intervals with radius $\sqrt{4L/\alpha n}$. For confidence levels 95% and 99%, these intervals are about 1.6 and 3.1 times wider, respectively, than the intervals in Corollary 7. However, Chebyshev's inequality holds for the variance of the estimator, so we do not need to use the variance bound in Theorem 3. Because the bound in Theorem 3 sometimes is quite loose, confidence intervals based on Chebyshev's inequality using the variance could be narrower than the intervals in Corollary 7. That is, $\text{Var}(\widehat{\tau})/\alpha$ may be smaller than $8\log(2/\alpha)L/n$ because $4L/n$ may be considerably larger than $\text{Var}(\widehat{\tau})$.

It is when the design parameter ϕ is close to zero that the variance bound in Theorem 3 tends to be loose. For this reason, the confidence intervals we describe here are primarily useful when experimenters prioritize robustness and have small samples.

11.3 An estimator of the ridge loss

The ridge loss L in Theorem 3 is not known because it depends on $\boldsymbol{\mu}$. This vector is unobserved even after the experiment is conducted. To construct the confidence intervals in Corollary 7, we need an estimator of L.

Recall that we can write the ridge loss as a quadratic form:

$$L = \frac{1}{n}\boldsymbol{\mu}^\intercal \boldsymbol{Q}\boldsymbol{\mu} \qquad \text{where} \qquad \boldsymbol{Q} = \left(\boldsymbol{B}^\intercal\boldsymbol{B}\right)^{-1} = \left(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\right)^{-1}.$$

The matrix $\boldsymbol{Q}$ is known, so the elements of $\boldsymbol{\mu}$ are the only unknown quantities. Using the construction of $\boldsymbol{\mu} = (\boldsymbol{a}+\boldsymbol{b})/2$, we can decompose the scaled ridge loss as

$$4nL = 4\boldsymbol{\mu}^\intercal\boldsymbol{Q}\boldsymbol{\mu} = \boldsymbol{a}^\intercal\boldsymbol{Q}\boldsymbol{a} + \boldsymbol{b}^\intercal\boldsymbol{Q}\boldsymbol{b} + 2\boldsymbol{a}^\intercal\boldsymbol{Q}\boldsymbol{b}.$$

We obtain an estimator of the loss if we can estimate the constituent terms of this expression. This can be achieved as long as each term is observed with a positive probability.

There are three types of terms. The first type corresponds to the diagonals of $\boldsymbol{Q}$ in $\boldsymbol{a}^\intercal\boldsymbol{Q}\boldsymbol{a}$ and $\boldsymbol{b}^\intercal\boldsymbol{Q}\boldsymbol{b}$. These terms are of the forms $q_{ii}a_i^2$ or $q_{ii}b_i^2$ where q_{ij} denotes the representative element of $\boldsymbol{Q}$. Corollary 1 ensures that these quantities are observed with probabilities of one half, so they can be estimated. The second type are the off-diagonal terms, which are of the forms $q_{ij}a_ia_j$, $q_{ij}b_ib_j$ and $q_{ij}a_ib_j$. The following lemma ensures that these quantities are observed with positive probabilities.

Lemma 4. *The second-order assignment probabilities are bounded away from zero under the Gram–Schmidt Walk design for all pairs of units and all treatments:*

$$\Pr\big((z_i, z_j) = \boldsymbol{v}\big) > \frac{1}{4n} \min\Big\{\phi, \frac{\phi^2}{1-\phi}\Big\} \quad \textit{for all} \quad i \neq j \quad \textit{and all} \quad \boldsymbol{v} \in \{\pm 1\}^2.$$

Lemma 4 is a worst-case bound uniformly over all pairs of units, ensuring that the estimator we construct is always well-defined. The second-order assignment probabilities will generally be further away from zero than what is indicated by the bound. Lemma A16 in Supplement A describes one such setting.

The third type of term comes from the diagonals of $\boldsymbol{Q}$ in $\boldsymbol{a}^\intercal \boldsymbol{Q} \boldsymbol{b}$, which are of the form $q_{ii} a_i b_i$. These quantities are never observed because a unit cannot be assigned to two different treatments simultaneously (Neyman, 1923; Holland, 1986). Following Aronow & Samii (2017), we use Young's inequality for products to construct an upper bound:

$$2\boldsymbol{a}^\intercal \operatorname{diag}(\boldsymbol{Q})\boldsymbol{b} \leq \boldsymbol{a}^\intercal \operatorname{diag}(\boldsymbol{Q})\boldsymbol{a} + \boldsymbol{b}^\intercal \operatorname{diag}(\boldsymbol{Q})\boldsymbol{b},$$

where $\operatorname{diag}(\boldsymbol{Q})$ denotes a diagonal matrix that is equal to the diagonal in $\boldsymbol{Q}$. This allows us to bound the ridge loss by a quantity that does not involve the problematic terms:

$$4nL \leq \boldsymbol{a}^\intercal \big[\boldsymbol{Q} + \operatorname{diag}(\boldsymbol{Q})\big]\boldsymbol{a} + \boldsymbol{b}^\intercal \big[\boldsymbol{Q} + \operatorname{diag}(\boldsymbol{Q})\big]\boldsymbol{b} + 2\boldsymbol{a}^\intercal \big[\boldsymbol{Q} - \operatorname{diag}(\boldsymbol{Q})\big]\boldsymbol{b}.$$

A Horvitz–Thompson-type estimator can now be used to estimate the constituent terms. In particular, construct a random matrix $\widehat{\boldsymbol{Q}}$ such that its representative element is

$$\widehat{q}_{ij} = \begin{cases} q_{ii} & \text{if } i = j, \\ q_{ij} / \big(1 + z_i z_j \operatorname{E}[z_i z_j]\big) & \text{otherwise,} \end{cases}$$

where q_{ij} denotes the representative element of $\boldsymbol{Q}$. The ridge loss can then be estimated by

$$\widehat{L} = \frac{1}{n} \boldsymbol{y}^\intercal \widehat{\boldsymbol{Q}} \boldsymbol{y}.$$

Proposition 4. *The ridge loss estimator is conservative in expectation:* $\operatorname{E}[\widehat{L}] \geq L$.

Conservativeness ensures that the estimator, in expectation, underestimates the precision of the point estimator, resulting in overly wide confidence intervals. An unbiased estimator is in general not possible because we never observe both potential outcomes of a unit simultaneously. Experimenters tend to prefer pessimistic confidence intervals over unduly optimistic ones, which Proposition 4 ensures. However, the proposition does not ensure that the estimator is larger than the true ridge loss for any specific assignment.

The matrix $\widehat{\boldsymbol{Q}}$ is in principle observed, but its construction may be computationally

challenging in experiments with large samples. The first challenge is the construction of $\boldsymbol{Q}$. A straightforward way to compute this matrix is simply to invert the matrix $\boldsymbol{B}^\intercal\boldsymbol{B}$. However, this will often require considerably more computational resources than sampling assignments from the design. In Supplement B, we provide an algorithm to compute $\boldsymbol{Q}$ that has the same time and space complexity as the design itself.

The second challenge is to derive the expectations $\mathrm{E}[z_i z_j]$. The design does not provide direct control over the cross-moments of the assignment vector, and these moments generally depend on the covariates in intricate ways. We may, however, estimate the expectations to an arbitrary precision by repeatedly sampling from the design (Fattorini, 2006; Aronow & Samii, 2017). The average of $\boldsymbol{z}\boldsymbol{z}^\intercal$ over these repeated samples provides an estimate of $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal]$. With a small or moderate number of units, we can draw enough assignments to estimate the entries of $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal]$ with high accuracy. It may, however, be computationally infeasible to draw a large number of assignment vectors when n is large.

In Supplement A, we derive an alternative estimator of L based on the empirical ridge loss for the realized assignments. This estimator does not depend on $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal]$, and it thereby avoids the need for Monte Carlo estimation. The alternative estimator may underestimate L because the empirical ridge loss will be systematically lower than the population loss. This is, however, typically not an important concern when number of units is large.

12 Computational properties

The structure of the augmented covariates allow us to construct a customized implementation of the Gram–Schmidt Walk algorithm that is considerably faster than a general implementation. Supplement B describes this implementation and proves its computational properties. The results are summarized here.

Lemma 5. *The Gram–Schmidt Walk terminates after at most n iterations.*

Proof. The step direction is selected under the condition that the coordinates of units with integral assignments are zero. As a consequence, once a unit is assigned an integral assignment, it keeps that assignment. Furthermore, the candidate step sizes are selected so that at least one fractional assignment is updated to be integral at every iteration. The implication is that the number of units with integral assignments grows by at least one per iteration. □

The lemma itself does not imply that the algorithm terminates quickly because each iteration may be slow to complete. The main bottleneck here is the computation of the step direction. This is a least squares problem, so the solution can be obtained by solving a system of linear equations. The number of equations is on the order of the number of units. However, the structure of the augmented covariates allows us to reduce the

problem to a linear system with d equations followed by a matrix–vector multiplication. This greatly reduces the run time when the number of covariates is fewer than the number of units: $d < n$. We obtain additional computational improvements by maintaining a matrix factorization for repeated linear system solves. Together, the techniques allow us to complete each iteration using $\mathcal{O}(dn)$ operations. Combined with Lemma 5, this yields the following proposition.

Proposition 5. *Assignments from the Gram–Schmidt Walk design can be sampled using $\mathcal{O}(dn^2)$ arithmetic operations and $\mathcal{O}(d^2 + n)$ additional storage.*

The proposition tells us that sampling an assignment from the design requires roughly the same computational resources as computing all pairwise inner products between all covariate vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. These products are, for example, used to derive Euclidean distances between the vectors. The run time of the Gram–Schmidt Walk design is therefore on the same order as a greedy implementation of the matched pair design.

13 Extensions

13.1 Non-uniform assignment probabilities

The Gram–Schmidt Walk design can be extended to allow arbitrary assignment probabilities. We achieve this by changing the initial fractional assignments of the algorithm. The experimenter provides a parameter vector $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n) \in (0,1)^n$ specifying the desired first-order assignment probability for each unit. The first step of the algorithm in Section 4 is then modified so that $\boldsymbol{z}_1 \leftarrow 2\boldsymbol{\pi} - \mathbf{1}$. The following corollary is a direct consequence of the martingale property of the fractional updates, in the same fashion as Corollary 1.

Corollary 8. *Under the non-uniform Gram–Schmidt Walk design,*

$$\Pr(z_i = 1) = \pi_i \qquad \textit{for all} \qquad i \in [n].$$

The properties of the original version of the design can be extended to the non-uniform version. To do so, we need to redefine the potential outcome vectors as

$$\widetilde{\boldsymbol{a}} = \left(\frac{a_1}{2\pi_1}, \ldots, \frac{a_n}{2\pi_n}\right) \qquad \text{and} \qquad \widetilde{\boldsymbol{b}} = \left(\frac{b_1}{2(1-\pi_1)}, \ldots, \frac{b_n}{2(1-\pi_n)}\right).$$

These are the original potential outcome vectors except that each coordinate is weighted by the probability that the potential outcome is observed. If $\boldsymbol{\pi} = 0.5 \times \mathbf{1}$, then $\widetilde{\boldsymbol{a}} = \boldsymbol{a}$ and $\widetilde{\boldsymbol{b}} = \boldsymbol{b}$, which replicates the uniform version of the design. The mean square error of the

Horvitz–Thompson estimator can now be expressed as

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4}{n^2} \widetilde{\boldsymbol{\mu}}^{\intercal} \operatorname{Cov}(\boldsymbol{z}) \widetilde{\boldsymbol{\mu}} \qquad \text{where} \qquad \widetilde{\boldsymbol{\mu}} = \frac{\widetilde{\boldsymbol{a}} + \widetilde{\boldsymbol{b}}}{2}.$$

The expression extends Lemma 1 to any experimental design with non-deterministic assignments. As a consequence, the spectral interpretation in Section 3.2 is valid also for non-uniform designs if one substitutes $\widetilde{\boldsymbol{\mu}}$ for $\boldsymbol{\mu}$. We show in Supplement A that Theorems 1 and 4 hold for the non-uniform version of the design, so all properties that follow from these theorems also apply to the extended version when $\widetilde{\boldsymbol{\mu}}$ is substituted for $\boldsymbol{\mu}$.

13.2 Fixed treatment group sizes

The Gram–Schmidt Walk design does not fix the size of the treatment groups. It tends to balance the group sizes if the covariates include a constant, because balancing such a covariate is the same as balancing the group sizes. However, this only stabilizes the group sizes, and the number of treated units is still random. The design can be extended to strictly enforce a desired number of treated units.

The group-balanced Gram–Schmidt Walk design is obtained by changing the construction of the step direction. In Step 3c of the algorithm in Section 4, the direction $\boldsymbol{u}_t$ was selected from the set U. This set contained all vectors satisfying the constraints for the pivot unit and the units with integral assignments. For this extension, we add the constraint that the sum of the coordinates of the step direction is zero. That is, we impose $\langle \mathbf{1}, \boldsymbol{u} \rangle = 0$ for all vectors $\boldsymbol{u} \in U$. One exception is when only one unit remains with a fractional assignment, in which case the step direction is the corresponding standard basis vector.

The modification ensures that the number of treated units $n_+ = \sum_{i=1}^n \mathbb{1}[z_i = 1]$ is as close as possible the expected number of treated units $\mathrm{E}[n_+] = \sum_{i=1}^n \pi_i$.

Proposition 6. *With probability one under the group-balanced Gram–Schmidt Walk design,*

$$\big|n_+ - \mathrm{E}[n_+]\big| < 1.$$

If $\mathrm{E}[n_+]$ *is an integer, then* $n_+ = \mathrm{E}[n_+]$ *with probability one.*

We describe an efficient implementation of this version of the Gram–Schmidt Walk design in Supplement B. The computational requirements are roughly the same as the original version, as both the time and space complexities are unchanged. However, the modification breaks certain orthogonality properties of the iterative updates, so the analysis in Sections 6 and 11 does not apply. The simulation study in the next section indicates that the behavior of the extended version is close to the original version.

The main reason for fixing the group sizes is that the estimator becomes invariant to constant shifts in the potential outcomes. Experimenters often find this a desirable property because the average treatment effect is itself invariant to such shifts. In terms of the spectral interpretation in Section 3.2, fixing the group sizes results in a covariance matrix with the vector of ones as an eigenvector for which the corresponding eigenvalue is zero. Such a covariance matrix can only be achieved by increasing the remaining eigenvalues, and thus potentially making the worst-case precision worse. This is a cost one might be willing to accept if the potential outcomes are known to be aligned with the vector of ones.

14 Numerical illustration

To illustrate and complement the theoretical results, we conduct a simulation study of the Gram–Schmidt Walk design and a set of comparison designs.

The simulations are based on data from an experiment by Groh & McKenzie (2016). The experiment investigates how insurance against macroeconomic shocks affects microenterprises in Egypt. The sample consisted of 2961 enterprises that were clients of Egypt's largest microfinance institution. The authors offered the insurance to a randomly selected subset of the enterprises, using a combination of stratification and the matched pair design. After three to seven months, they measured various outcomes and estimated the treatment effects by comparing the two treatment groups. The estimates indicate that the insurance had little impact on the enterprises.

Our aim here is not to recreate the exact empirical setting in the original experiment. This is generally not possible because we never observe all potential outcomes. The purpose is instead to use the data from Groh & McKenzie (2016) to create a plausible empirical setting. We create three samples from the original data set. Respectively, they contain the 30, 296 and 2960 first observations according to the original order of the data. All samples contain an even number of observations because some of the comparison designs require modifications to accommodate an odd number. In the interest of space, we present only the results for the sample with 296 observations in the main paper. The results from the other two samples are presented in Supplement C.

The covariates we will seek to balance are the 14 covariates that Groh and McKenzie use in their matched paired design. However, two of these covariates are almost collinear. They are indicator variables of missingness of two other covariates, and we collapse them into a single covariate using disjunction. The covariates are decorrelated before treatment assignment so that they are in scaled isotropic position: $\boldsymbol{X}^\intercal \boldsymbol{X} = n\boldsymbol{I}$.

We include the following experimental designs in the simulation study:

1. The fully randomized design, in which the treatments are assigned independently.

2. The group-balanced randomization design, in which each treatment group is ensured to contain exactly half of the units.
3. The matched pair design using the network flow algorithm by Greevy et al. (2004) to construct optimal pairs.
4. The essentially non-random design by Bertsimas et al. (2015), which we abbreviate as BJK. This design finds the most balanced assignment according to an intricate balance objective. The only randomness is a sign flip of the assignment vector. The design is specified by a parameter which determines the trade-off between the first and second moments of the covariate imbalances. We set this parameter to $\rho = 0.5$, which is the value chosen by Bertsimas et al. (2015) in their application and the default value in their implementation of the design.
5. The design by Krieger et al. (2019), abbreviated KAK, that makes local swaps in a randomly generated assignment vector until a local imbalance minimum is reached.
6. The rerandomization procedure described by Lock Morgan & Rubin (2012) and Li et al. (2018). We use four different acceptance criteria. These are, respectively, that the imbalance of an acceptable assignment is at most a fraction of 0.5, 0.2, 0.15 or 0.1 of the expected imbalance under the fully randomized design as measured by squared norm. The criteria correspond to acceptance rates of about 6.8%, 0.08%, 0.02% and 0.001% of the candidate assignments drawn from the group-balanced design.
7. The Gram–Schmidt Walk design as presented in Section 4. We set the parameter ϕ to five values ranging from focusing mostly on balance to focusing mostly on robustness: 0.01, 0.1, 0.5, 0.9 and 0.99.
8. The group-balanced version of the Gram–Schmidt Walk design presented in Section 13.2, using the same parameter values as for the unbalanced version.

Table 1 presents the results based on one million draws from each design. The first column, denoted $\lambda_{\boldsymbol{z}}$, is the largest eigenvalue of the covariance matrix $\mathrm{Cov}(\boldsymbol{z})$. This quantity captures the robustness of the design. Monte Carlo estimates of largest eigenvalues of the covariance matrices are generally less precise than corresponding estimates of particular values of the quadratic form determined by the covariance matrix. This is the reason for the large number of Monte Carlo replicates in our simulation study. The one million replicates should give sufficient precision to make the estimates informative. We note, however, that the eigenvalue is known to be 1, 1.003 and 2, respectively, for the first three designs, indicating that we slightly overestimate the values.

We see that the BJK design affords no robustness with a maximum eigenvalue of 296, which is the largest possible value. This tells us that the mean square error under this design could be 296 times greater than under the fully randomized design. The remaining designs have values in the interval $[1, 2]$. For rerandomization and the Gram–Schmidt Walk

Table 1: Robustness, balance and precision under various designs when $n = 296$

		Covariate balance			Root mean square error			
Design	$\lambda_{\boldsymbol{z}}$	$\lambda_{\boldsymbol{X}}$	$\boldsymbol{X}$	$\mathbf{1}$	A	B	C	D
Fully random	1.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.04	1.00	1.00	0.00	0.94	0.71	1.00	1.00
Matched pairs	2.05	0.42	0.18	0.00	0.95	0.54	1.08	0.37
BJK	296.00	11.01	0.85	0.00	1.15	0.14	2.72	0.26
KAK	1.63	0.08	0.01	0.00	0.93	0.55	1.28	0.05
Rerand 0.50	1.26	0.43	0.41	0.00	0.94	0.62	1.12	0.64
Rerand 0.20	1.45	0.17	0.17	0.00	0.93	0.58	1.20	0.41
Rerand 0.15	1.49	0.13	0.13	0.00	0.93	0.57	1.22	0.36
Rerand 0.10	1.54	0.09	0.09	0.00	0.93	0.57	1.24	0.29
GSW 0.99	1.03	0.98	0.98	0.97	1.00	0.99	1.00	0.99
GSW 0.90	1.08	0.81	0.79	0.78	0.99	0.92	1.03	0.89
GSW 0.50	1.30	0.33	0.29	0.29	0.95	0.71	1.14	0.54
GSW 0.10	1.50	0.08	0.05	0.05	0.94	0.58	1.22	0.23
GSW 0.01	1.58	0.03	0.02	0.02	0.93	0.57	1.26	0.14
BGSW 0.99	1.04	0.98	0.98	0.00	0.94	0.70	1.01	0.99
BGSW 0.90	1.08	0.81	0.79	0.00	0.94	0.68	1.04	0.89
BGSW 0.50	1.30	0.33	0.29	0.00	0.94	0.60	1.14	0.54
BGSW 0.10	1.50	0.08	0.05	0.00	0.93	0.56	1.22	0.23
BGSW 0.01	1.58	0.04	0.02	0.00	0.93	0.56	1.26	0.15

design, the level of balance achieved depends on the acceptance criteria and parameter values. As we expect from the discussion in Section 3.2, the largest eigenvalue increases as more balance is sought. The eigenvalue for the Gram–Schmidt Walk design is considerably smaller than $1/\phi$, which is the guaranteed level of robustness given by Theorem 2. This highlights that the guarantee is the worst-case over all possible covariates.

The second column, denoted $\lambda_{\boldsymbol{X}}$, is the maximum eigenvalue of the covariance matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$. This captures the level balance for the worst-case linear function the covariates. Unlike the metric in the first column, the maximum eigenvalue of the covariance matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ does not have a natural scale. We therefore present this metric relative to the fully randomized design in the first row. All designs that aim to balance the covariates yield smaller eigenvalues than the benchmark design. The one exception is the BJK design, which focus on a heuristic balance metric that includes both first and second moments of the covariate distributions. As a result, the design could potentially balance some non-linear functions better than the other designs, at the cost of less balance on linear functions as indicated by this column.

The third column, denoted $\boldsymbol{X}$, is the mean square norm of the vector of covariate

difference between the treatment groups: $\mathrm{E}[\|\boldsymbol{X}^\intercal \boldsymbol{z}\|^2]$. These results largely mirror the results for the maximum eigenvalue of the covariance matrix $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ in the previous column. The main difference is the BJK design performs better here than for the eigenvalue, indicating that it is better at inducing balance on the first moments of the covariates than on the worst-case linear function.

The fourth column, denoted $\mathbf{1}$, is the mean square size difference of the treatment groups:

$$\mathrm{E}\Big[\big(|Z^+| - |Z^-|\big)^2\Big] = \mathrm{E}\big[\langle \mathbf{1}, \boldsymbol{z}\rangle^2\big].$$

Most of the designs balance the treatment groups by construction, and this measure is then zero. The exceptions are the fully randomized design and the unbalanced version of the Gram–Schmidt Walk design.

The final four columns, denoted with letters A–D, are the root mean square error of the Horvitz–Thompson estimator for four different outcomes. The mean square error of the estimator depends on the potential outcomes solely through the vector $\boldsymbol{\mu} = (\boldsymbol{a} + \boldsymbol{b})/2$. Hence, the existence of a treatment effect does not matter for these results, so for all four sets of outcomes, we set $a_i = b_i$ for all units $i \in [n]$. However, the outcomes generally differ between units, so that $a_i \neq a_j$. We investigate the difference-in-means estimator and the regression estimator by Lin (2013) in Supplement C. The behavior of these estimators depend on existence of a treatment effects, and we include outcomes with heterogeneous treatment effects in that analysis.

The four outcomes are:

A. An outcome from the original study, namely whether the enterprises have hired a new worker after treatment assignment. We remove the estimated treatment effect from units that were treated in the original study. This is in an effort to remove most of any treatment effect in the original study. The purpose of this approach is, however, not to recreate the true potential outcomes. That would require a constant treatment effect assumption, which we have no reason to believe that it holds. The purpose is instead to create potential outcomes that are empirically plausible. The covariates are not particularly informative of this outcome. The coefficient of determination (R^2) is 0.065.

B. Another outcome from the original study, namely the profits of the enterprises after treatment assignment. We remove the estimated treatment effect from the originally treated units in the same way as for the previous outcome. Two of the covariates are the enterprises' profits at baseline before treatment assignment, so the covariates are highly predictive of this outcome. The coefficient of determination is 0.419.

C. An artificially generated outcome to represent a worst-case scenario. We generate the outcome based on the largest eigenvectors of five of the designs in the study:

the matched pair design, the BJK design, the KAK design, rerandomization with criterion 0.10 and the Gram–Schmidt Walk design with $\phi = 0.01$. In particular, the outcome vector is the largest eigenvector of the sum of the outer products of the eigenvectors from the five designs. These designs are strongly tilted towards seeking covariate balance, and they are subsequently some of the least robust designs in the study. By construction, the covariates are almost completely uninformative of this potential outcome. The coefficient of determination is less than 0.001.

D. An artificially generated outcome to represent a setting where the covariates are perfectly predictive of the potential outcomes. The potential outcomes for each unit is the sum of its covariates: $a_i = b_i = \langle \mathbf{1}, \boldsymbol{x}_i \rangle$. The coefficient of determination is 1.0.

The root mean square error of the first and third outcomes are similar for all designs. These are the outcomes for which the covariates are not predictive, so we expect only small improvements over the fully randomized design. The covariates are somewhat predictive of the first outcome, and we see some improvements in precision. The group-balanced design is one of the best performing designs here, indicating that the outcome is aligned with the vector of ones. The third outcome is a type of communal worst case by construction. All designs yield less precision than the fully randomized design, and designs that put more emphasis on balance tend to perform worse. The BJK design is a notable outlier here with a root mean square error almost three times as high as the fully randomized design. This is a consequence of the high risk–high reward strategy that this design implicitly adopts. Note that this root mean square error is still far from the design's worst-case precision, which is $\sqrt{296} = 17.2$ times higher than the fully randomized design.

The second and fourth outcomes demonstrate that we can achieve considerable improvement over the fully randomized design if the covariates are predictive of the potential outcomes. As a general pattern, the more the designs balance the covariates, the more precise the estimator is for these outcomes. For the second outcome, the BJK design stands out with a root mean square error of almost one tenth of the benchmark design. This is, again, a consequence of its high risk–high reward strategy. It also indicates that at least one of the non-linear functions that the BJK design implicitly targets is more predictive than the linear functions the other designs target. For the fourth outcome, the designs that target linear functions give the most precise estimates. This is expected given that the outcome is generated so that the covariates are perfectly linearly predictive.

In Supplement C, we report coverage probabilities and widths of the confidence intervals discussed in Section 11. The confidence intervals in Corollary 7, which were based on the sharpened subgaussian tail bounds, covers the true average treatment effect with a probability that exceeds the nominal level for all three sample sizes and all outcomes. This is expected given that the intervals do not use large sample approximations and are therefore valid in finite samples. Perhaps less expected is that the same holds already with 296 units for the confidence intervals based on a normal approximation. This is an indication that

the sampling distribution of the estimator may converge to a normal distribution under the Gram–Schmidt Walk design. The relative width of the confidence intervals in Corollary 7 depends on the choice of the design parameter ϕ. For small values of ϕ, the intervals are considerably wider than even the intervals based on Chebyshev's inequality, and they are then unlikely to be useful in practice. For values $\phi \geq 0.5$, the intervals are markedly narrower than those based on Chebyshev's inequality, but as expected, they are still wider than those based on a normal approximation.

15 Practical considerations and recommendations

It is beyond the scope of the paper to investigate which covariates should be balanced and how the balance–robustness trade-off should be resolved. These questions can only be answered by an experimenter's preferences and substantive knowledge about the empirical setting. The general approach should be to prioritize balance over robustness in settings where the covariates are suspected to be predictive. Experimenters may therefore want to set the parameter ϕ to a lower value when they have high-quality covariates, such as a baseline measure of the outcome variable. However, the exchange rate between balance and robustness becomes worse as ϕ decreases. For example, Theorem 3 tells us that going from $\phi = 0.1$ to $\phi = 0.01$ yields an improvement in the bound on the mean square error of about 10% in the best-case scenario where the covariates are perfectly predictive of the potential outcomes. However, the theorem also tells us that the change could make the mean square error ten times as large in the worse-case scenario where the covariates are completely unpredictive. Experimenters should therefore be careful setting the parameter too low.

Experimenters should also consider transforming the covariates to make them as linearly predictive as possible relative to their dimensions. This could include removing or down-weighting covariates which are suspected not to be predictive and including higher-order terms and interactions if they are believed to be prognostically important. Experimenters should avoid covariates that are on very different scales, unless the scaling reflect the predictiveness of the covariates, because the design will put disproportionally much effort towards balancing covariates on large scales. It may therefore sometimes be useful to normalize or decorrelate the covariates prior to the design phase. By the same token, experimenters should generally avoid including covariates not believed to be predictive because the design will attempt to balance these uninformative covariates at the expense of important covariates. In some cases, experimenters do not know which covariates are prognostically important, and they may therefore want to seek balance on a large set of them. The Gram–Schmidt Walk design can accommodate this, but the balance on each covariate will naturally be worse than in a setting where a small group of covariates is targeted.

While we hope experimenters will find the Gram–Schmidt Walk design useful in practice,

many of the insights in the paper are applicable even if they prefer to use other designs. We suspect the most useful of these insights is the spectral interpretation in Section 3. It is not always necessary to investigate the covariance matrix of the assignment vector analytically as we have done in this paper. As long as it is possible to repeatedly sample from a design before the experiment is conducted, one can use the Monte Carlo method to estimate the covariance matrix to arbitrary precision. This allows experimenters to inspect the eigensystem of the matrix before conducting an experiment, and to examine how the design will perform for various potential outcomes. For example, experimenters may investigate a design's worst-case precision by estimating the maximum eigenvalue of the covariance matrix. This could help experimenters make more informed decisions when designing their experiments because they can choose between many possible designs with known properties.

One approach we suspect experimenters will find particularly useful is hybrid designs that combine components of other designs. When carefully combined, such hybrids inherit the balancing properties of its parent designs without sacrificing too much robustness. However, the hybrids are often harder to investigate analytically, so an estimated eigensystem may give experimenters enough insights to be comfortable to use such designs in practice. We saw one example of this with the group-balanced Gram–Schmidt Walk design in Section 13. This is effectively a hybrid of the group-balanced randomization design and the ordinary Gram–Schmidt Walk design. Another hybrid that we suspect experimenters will find useful is the combination of the matched pair design and the Gram–Schmidt Walk design. The assignments are here perfectly correlated within pairs as in the matched pair design, but the sign of the assignments in each pair is decided by the Gram–Schmidt Walk design, so the assignments are weakly dependent between pairs.

16 Concluding remarks

Scientists use randomized experiments for the robustness they provide. Even a minimal amount of randomization provides robustness in the sense of unbiasedness. The motivating idea of this paper is that a more appropriate concept of robustness is the magnitude of the estimation error under unfavorable circumstances. This perspective highlights that a compromise between balance and robustness is central to the experimental design problem. At one extreme, we can resolve this trade-off cautiously by assigning treatments independently at random. This yields a design that is maximally robust. At the other extreme, we can make all assignments perfectly dependent. This yields a design that performs exceptionally well for some potential outcomes, but it will perform exceptionally poorly for other outcomes. Most experimenters are neither so risk averse nor so risk loving to prefer either of these two options. Instead, they prefer the intermediate designs that introduce weak dependencies between the assignments to achieve some balance at the cost of some

robustness.

The purpose of the Gram–Schmidt Walk design is to help experimenters navigate this compromise. The design does not perform uniformly better than existing designs. Indeed, we showed in Section 3.2 that all designs are admissible, so no uniformly optimal design exists. The main benefit of the design is instead that it provides precise control over the balance–robustness trade-off through its parameter. Our theoretical results also provide a firm understanding of the design's behavior. This yields both well-motivated confidence statements that are valid in finite samples and a near-optimality guarantee on the trade-off between robustness and balance on worst-case linear functions.

One of the chief short-comings of the Gram–Schmidt Walk design is that it solely focuses on linear functions and does not inherently balance non-linear functions of the covariates. We also caution the reader that the estimator's sampling distribution depends on more than the covariance matrix of the assignment vector. Stronger tail bounds require a characterization of higher moments, but these are difficult to analyze. This leads to one of the major open problems suggested by our analysis: does the assignment vector produced by the Gram–Schmidt Walk design satisfy an instance-optimal subgaussian inequality that can be stated in terms of the covariance matrix of the imbalances of the augmented covariates instead of the upper bound we prove on this covariance matrix? Proving such a result could facilitate narrower confidence intervals, but it would require different techniques than those we have used here.

References

Aronow, P. M. & Middleton, J. A. (2013). A class of unbiased estimators of the average treatment effect in randomized experiments. *Journal of Causal Inference*, 1(1).

Aronow, P. M. & Samii, C. (2017). Estimating average causal effects under general interference. *Annals of Applied Statistics*, 11(4), 1912–1947.

Bai, Y. (2019). Randomization under permutation invariance. SSRN Preprint 3475147.

Banaszczyk, W. (1998). Balancing vectors and Gaussian measures of n-dimensional convex bodies. *Random Structures and Algorithms*, 12(4), 351–360.

Bansal, N., Dadush, D., Garg, S., & Lovett, S. (2019). The Gram-Schmidt Walk: A cure for the Banaszczyk Blues. *Theory of Computing*, 15(21), 1–27.

Basse, G., Ding, Y., & Toulis, P. (2019). Minimax crossover designs. arXiv:1908.03531v1.

Bertsimas, D., Johnson, M., & Kallus, N. (2015). The power of optimization over randomization in designing experiments involving small samples. *Operations Research*, 63(4), 868–876.

Bhatia, R. (1997). *Matrix Analysis*. New York: Springer.

Boyd, S. & Vandenberghe, L. (2004). *Convex Optimization*. New York: Cambridge University Press.

Charikar, M., Newman, A., & Nikolov, A. (2011). Tight hardness results for minimizing discrepancy. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11 (pp. 1607–1614).: Society for Industrial and Applied Mathematics.

Chazelle, B. (2000). *The Discrepancy Method: Randomness and Complexity*. Cambridge: Cambridge University Press.

Deaton, A. & Cartwright, N. (2018). Understanding and misunderstanding randomized controlled trials. *Social Science & Medicine*, 210, 2–21.

Delevoye, A. & Sävje, F. (2020). Consistency of the Horvitz–Thompson estimator under general sampling and experimental designs. *Journal of Statistical Planning and Inference*, 207, 190–197.

Efron, B. (1971). Forcing a sequential experiment to be balanced. *Biometrika*, 58(3), 403–417.

Fattorini, L. (2006). Applying the Horvitz–Thompson criterion in complex designs: A computer-intensive perspective for estimating inclusion probabilities. *Biometrika*, 93(2), 269–278.

Fisher, R. A. (1925). *Statistical Method for Research Workers*. Edinburgh: Oliver & Boyd.

Fisher, R. A. (1926). The arrangement of field experiments. *Journal of the Ministry of Agriculture of Great Britain*, 33, 503–513.

Fisher, R. A. (1935). *The Design of Experiments*. London: Oliver & Boyd.

Freedman, D. A. (1975). On tail probabilities for martingales. *Annals of Probability*, 3(1), 100–118.

Greevy, R., Lu, B., Silber, J. H., & Rosenbaum, P. (2004). Optimal multivariate matching before randomization. *Biostatistics*, 5(2), 263–275.

Groh, M. & McKenzie, D. (2016). Macroinsurance for microenterprises: A randomized experiment in post-revolution Egypt. *Journal of Development Economics*, 118, 13–25.

Harville, D. A. (1975). Experimental randomization: Who needs it? *American Statistician*, 29(1), 27–31.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. New York: Springer, second edition.

Higgins, M. J., Sävje, F., & Sekhon, J. S. (2016). Improving massive experiments with threshold blocking. *Proceedings of the National Academy of Sciences*, 113(27), 7369–7376.

Holland, P. W. (1986). Statistics and causal inference. *Journal of the American Statistical Association*, 81(396), 945–960.

Hooper, P. M. (1989). Minimaxity of randomized optimal designs. *Annals of Statistics*, 17(3), 1315–1324.

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

Imai, K., King, G., & Nall, C. (2009). The essential role of pair matching in cluster-randomized experiments, with application to the mexican universal health insurance evaluation. *Statistical Science*, 24(1), 29–53.

Kallus, N. (2018). Optimal a priori balance in the design of controlled experiments. *Journal of the Royal Statistical Society: Series B*, 80(1), 85–112.

Kapelner, A., Krieger, A. M., Sklar, M., & Azriel, D. (2019). Optimal rerandomization via a criterion that provides insurance against failed experiments. arXiv:1905.03337v1.

Kapelner, A., Krieger, A. M., Sklar, M., Shalit, U., & Azriel, D. (2020). Harmonizing optimized designs with classic randomization in experiments. *American Statistician*, in print, 1–12.

Kasy, M. (2016). Why experimenters might not always want to randomize, and what they could do instead. *Political Analysis*, 24(03), 324–338.

Kiefer, J. (1959). Optimum experimental designs. *Journal of the Royal Statistical Society: Series B*, 21(2), 272–304.

Krieger, A. M., Azriel, D., & Kapelner, A. (2019). Nearly random designs with greatly improved balance. *Biometrika*, 106(3), 695–701.

Li, K.-C. (1983). Minimaxity for randomized designs: Some general results. *Annals of Statistics*, 11(1), 225–239.

Li, X., Ding, P., & Rubin, D. B. (2018). Asymptotic theory of rerandomization in treatment–control experiments. *Proceedings of the National Academy of Sciences*, 115(37), 9157–9162.

Lin, W. (2013). Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique. *Annals of Applied Statistics*, 7(1), 295–318.

Lock Morgan, K. & Rubin, D. B. (2012). Rerandomization to improve covariate balance in experiments. *Annals of Statistics*, 40(2), 1263–1282.

Matoušek, J. (1999). *Geometric Discrepancy*. Berlin: Springer.

Narain, R. (1951). On sampling without replacement with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, 3, 169—-175.

Neyman, J. (1923). On the application of probability theory to agricultural experiments. Essay on principles. Section 9. *Statistical Science*, 5(4), 465–472. Reprinted in 1990.

Rubin, D. B. (1980). Comment: Randomization analysis of experimental data. *Journal of the American Statistical Association*, 75(371), 591.

Spencer, J. (1985). Six standard deviations suffice. *Transactions of the American Mathematical Society*, 289(2), 679–679.

Stewart, G. W. (1998). *Matrix Algorithms: Vol. 1. Basic Decompositions*. Philadelphia: Society for Industrial and Applied Mathematics.

Strang, G. (2009). *Introduction to Linear Algebra*. Wellesley: Wellesley-Cambridge Press, fourth edition.

Student (1923). On testing varieties of cereals. *Biometrika*, 15(3-4), 271–293.

Student (1938). Comparison between balanced and random arrangements of field plots. *Biometrika*, 29(3/4), 363–378.

Taves, D. R. (1974). Minimization: A new method of assigning patients to treatment and control groups. *Clinical Pharmacology & Therapeutics*, 15(5), 443–453.

Trefethen, L. N. & Bau, D. (1997). *Numerical Linear Algebra*. Philadelphia: Society for Industrial and Applied Mathematics.

Wu, C.-F. (1981). On the robustness and efficiency of some randomized designs. *Annals of Statistics*, 9(6), 1168–1177.

Supplement A: Additional proofs and results

Contents

A1 Spectral interpretation

In this section, we derive the expression of the mean squared error of the Horvitz–Thompson estimator under a given design as a quadratic form in the covariance matrix of the assignment vector. We begin by deriving an expression for the error of the Horvitz–Thompson estimator. Once this expression of the error is obtained, the expression of the mean squared error as a quadratic form is straightforward. Finally, we derive the more general expression for designs with non-uniform treatment probabilities.

A1.1 Expressing MSE as a quadratic form (Lemma 1)

We begin by deriving an expression for the difference between the average treatment effect and the Horvitz–Thompson estimator.

Lemma A1. *For any experimental design with* $\Pr(z_i = 1) = 1/2$ *for all* $i \in [n]$*, the error of the Horvitz–Thompson estimator can be written as*

$$\widehat{\tau} - \tau = \frac{2}{n}\langle \boldsymbol{z}, \boldsymbol{\mu} \rangle \qquad \textit{where} \qquad \boldsymbol{\mu} = \frac{\boldsymbol{a} + \boldsymbol{b}}{2}.$$

Proof. Recall that the average treatment effect and Horvitz–Thompson estimator can be written as

$$\tau = \frac{1}{n}\langle \mathbf{1}, \boldsymbol{a} - \boldsymbol{b} \rangle \qquad \text{and} \qquad \widehat{\tau} = \frac{2}{n}\langle \boldsymbol{z}, \boldsymbol{y} \rangle.$$

Let $\mathring{\boldsymbol{a}} = (\mathring{a}_1, \ldots, \mathring{a}_n)$ and $\mathring{\boldsymbol{b}} = (\mathring{b}_1, \ldots, \mathring{b}_n)$ be the observed parts of $\boldsymbol{a}$ and $\boldsymbol{b}$, namely

$$\mathring{a}_i = \begin{cases} a_i & \text{if } z_i = 1, \\ 0 & \text{if } z_i = -1, \end{cases} \qquad \text{and} \qquad \mathring{b}_i = \begin{cases} 0 & \text{if } z_i = 1, \\ b_i & \text{if } z_i = -1. \end{cases}$$

Using linearity and the definitions of $\mathring{\boldsymbol{a}}$ and $\mathring{\boldsymbol{b}}$, we rewrite the average treatment effect as

$$n\tau = \langle \mathbf{1}, \boldsymbol{a} - \boldsymbol{b} \rangle = \langle \mathbf{1}, \mathring{\boldsymbol{a}} \rangle + \langle \mathbf{1}, \boldsymbol{a} - \mathring{\boldsymbol{a}} \rangle - \langle \mathbf{1}, \mathring{\boldsymbol{b}} \rangle - \langle \mathbf{1}, \boldsymbol{b} - \mathring{\boldsymbol{b}} \rangle.$$

Note that $\langle \mathbf{1}, \mathring{\boldsymbol{a}} \rangle = \langle \boldsymbol{z}, \mathring{\boldsymbol{a}} \rangle$ because $\mathring{a}_i$ is non-zero only when $z_i = 1$. By the same argument,

$$\langle \mathbf{1}, \boldsymbol{a} - \mathring{\boldsymbol{a}} \rangle = \langle -\boldsymbol{z}, \boldsymbol{a} - \mathring{\boldsymbol{a}} \rangle, \qquad \langle \mathbf{1}, \mathring{\boldsymbol{b}} \rangle = \langle -\boldsymbol{z}, \mathring{\boldsymbol{b}} \rangle \qquad \text{and} \qquad \langle \mathbf{1}, \boldsymbol{b} - \mathring{\boldsymbol{b}} \rangle = \langle \boldsymbol{z}, \boldsymbol{b} - \mathring{\boldsymbol{b}} \rangle.$$

This allows us to write the average treatment effect as

$$n\tau = \langle \boldsymbol{z}, \mathring{\boldsymbol{a}} \rangle - \langle \boldsymbol{z}, \boldsymbol{a} - \mathring{\boldsymbol{a}} \rangle + \langle \boldsymbol{z}, \mathring{\boldsymbol{b}} \rangle - \langle \boldsymbol{z}, \boldsymbol{b} - \mathring{\boldsymbol{b}} \rangle = 2\langle \boldsymbol{z}, \mathring{\boldsymbol{a}} + \mathring{\boldsymbol{b}} \rangle - \langle \boldsymbol{z}, \boldsymbol{a} + \boldsymbol{b} \rangle$$

Using the identity $\boldsymbol{y} = \mathring{\boldsymbol{a}} + \mathring{\boldsymbol{b}}$, we may rewrite the Horvitz-Thompson estimator as

$$n\widehat{\tau} = 2\langle \boldsymbol{z}, \boldsymbol{y} \rangle = 2\langle \boldsymbol{z}, \mathring{\boldsymbol{a}} + \mathring{\boldsymbol{b}} \rangle.$$

Using these expressions for the average treatment effect and the Horvitz-Thompson estimator, we can express their difference as

$$n(\widehat{\tau} - \tau) = n\widehat{\tau} - n\tau = \Big(2\langle \boldsymbol{z}, \mathring{\boldsymbol{a}} + \mathring{\boldsymbol{b}} \rangle\Big) - \Big(2\langle \boldsymbol{z}, \mathring{\boldsymbol{a}} + \mathring{\boldsymbol{b}} \rangle - \langle \boldsymbol{z}, \boldsymbol{a} + \boldsymbol{b} \rangle\Big) = \langle \boldsymbol{z}, \boldsymbol{a} + \boldsymbol{b} \rangle = 2\langle \boldsymbol{z}, \boldsymbol{\mu} \rangle,$$

where the final equality follows from the definition of $\boldsymbol{\mu} = (\boldsymbol{a} + \boldsymbol{b})/2$. □

The mean squared error follows from the expression for the error of the Horvitz–Thompson estimator in Lemma A1.

Lemma 1. *For any experimental design with* $\Pr(z_i = 1) = 1/2$ *for all* $i \in [n]$*, the mean square error of the Horvitz–Thompson estimator is*

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4}{n^2}\boldsymbol{\mu}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{\mu} \qquad \textit{where} \qquad \boldsymbol{\mu} = \frac{\boldsymbol{a} + \boldsymbol{b}}{2}.$$

Proof. Lemma A1 gives $n(\widehat{\tau} - \tau) = 2\langle \boldsymbol{z}, \boldsymbol{\mu} \rangle$. The expectation of the square of this expression is

$$n^2 \,\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = 4\boldsymbol{\mu}^\intercal \,\mathrm{E}\big[\boldsymbol{z}\boldsymbol{z}^\intercal\big]\boldsymbol{\mu},$$

because $\boldsymbol{\mu}$ is not random. The proof is completed by noting that $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal] = \operatorname{Cov}(\boldsymbol{z})$ because $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{0}$ when $\Pr(z_i = 1) = 1/2$ for all $i \in [n]$. □

A1.2 Extension to non-uniform assignment probabilities

The main body of the paper was primarily concerned with designs with uniform treatment probabilities; that is, $\Pr(z_i = 1) = 1/2$ for all units $i \in [n]$. In this section, we demonstrate how our the spectral interpretation extends to designs with non-uniform treatment probabilities. We will also see that our analysis of the Gram–Schmidt Walk design carries through for non-uniform treatment probabilities.

The primary goal of this discussion is to derive the mean squared error of the Horvitz–Thompson estimator for designs with non-uniform treatment probabilities. In general, the Horvitz–Thompson estimator is defined as

$$\widehat{\tau} = \frac{1}{n}\left[\sum_{i \in Z^+} \frac{a_i}{\Pr(z_i = 1)} - \sum_{i \in Z^-} \frac{b_i}{\Pr(z_i = -1)}\right].$$

We now derive an expression for the error of the Horvitz–Thompson estimator.

Lemma A2. *For any experimental design with assignment probabilities* $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n)$ *bounded away from zero and one, the error of the Horvitz–Thompson estimator can be written as*

$$\widehat{\tau} - \tau = \frac{2}{n}\big\langle \boldsymbol{z} - \mathrm{E}[\boldsymbol{z}], \widetilde{\boldsymbol{\mu}} \big\rangle \qquad \textit{where} \qquad \widetilde{\boldsymbol{\mu}} = \frac{\widetilde{\boldsymbol{a}} + \widetilde{\boldsymbol{b}}}{2},$$

and

$$\widetilde{\boldsymbol{a}} = \left(\frac{a_1}{2\pi_1}, \ldots, \frac{a_n}{2\pi_n}\right) \qquad \textit{and} \qquad \widetilde{\boldsymbol{b}} = \left(\frac{b_1}{2(1-\pi_1)}, \ldots, \frac{b_n}{2(1-\pi_n)}\right).$$

Proof. Let $Z^+ = \{i \in [n] : z_i = 1\}$ and $Z^- = \{i \in [n] : z_i = -1\}$ be the random partition of the units into treatment groups. We can now write the average treatment effect and the

Horvitz–Thompson estimator as

$$\tau = \frac{1}{n}\left[\sum_{i\in Z^+}(a_i - b_i) + \sum_{i\in Z^-}(a_i - b_i)\right] \qquad \text{and} \qquad \widehat{\tau} = \frac{1}{n}\left[\sum_{i\in Z^+}\frac{a_i}{\pi_i} - \sum_{i\in Z^-}\frac{b_i}{1-\pi_i}\right].$$

Moving the factor n to the left hand side for convenience, we can write the difference between the estimate and the estimtand as

$$\begin{aligned} n(\widehat{\tau} - \tau) &= \sum_{i\in Z^+}\left[\frac{1-\pi_i}{\pi_i}a_i + b_i\right] - \sum_{i\in Z^-}\left[a_i + \frac{\pi_i}{1-\pi_i}b_i\right] \\ &= \sum_{i\in Z^+} 4(1-\pi_i)\left[\frac{a_i}{4\pi_i} + \frac{b_i}{4(1-\pi_i)}\right] - \sum_{i\in Z^-} 4\pi_i\left[\frac{a_i}{4\pi_i} + \frac{b_i}{4(1-\pi_i)}\right]. \end{aligned}$$

The quantities in square brackets are the elements of $\widetilde{\boldsymbol{\mu}} = (\widetilde{\mu}_1, \ldots, \widetilde{\mu}_n)$. Note that

$$z_i - \mathrm{E}[z_i] = \begin{cases} 2(1-\pi_i) & \text{when } z_i = 1, \\ -2\pi_i & \text{when } z_i = -1, \end{cases}$$

because $\mathrm{E}[z_i] = \Pr(z_i = 1) - \Pr(z_i = -1) = \pi_i - (1-\pi_i)$. Finally, rewrite the sum as

$$n(\widehat{\tau} - \tau) = 2\sum_{i=1}^{n}\big(z_i - \mathrm{E}[z_i]\big)\widetilde{\mu}_i = 2\big\langle \boldsymbol{z} - \mathrm{E}[\boldsymbol{z}], \widetilde{\boldsymbol{\mu}}\big\rangle. \qquad \square$$

As before, we may now express the mean squared error of the Horvitz-Thompson estimator as a quadratic form using the error expression above. The only difference between the mean squared expression in Lemma 1 and the more general expression below is the definition of the average potential vectors.

Corollary A1. *For any experimental design, the mean square error of the Horvitz–Thompson estimator can be written as*

$$\mathrm{E}\big[(\widehat{\tau} - \tau)^2\big] = \frac{4}{n^2}\widetilde{\boldsymbol{\mu}}^\intercal \operatorname{Cov}(\boldsymbol{z})\widetilde{\boldsymbol{\mu}} \qquad \textit{where} \qquad \widetilde{\boldsymbol{\mu}} = \frac{\widetilde{\boldsymbol{a}} + \widetilde{\boldsymbol{b}}}{2}.$$

Proof. Use Lemma A2 to write

$$\frac{n^2}{4}(\widehat{\tau} - \tau)^2 = \widetilde{\boldsymbol{\mu}}^\intercal\Big(\boldsymbol{z}\boldsymbol{z}^\intercal - \mathrm{E}[\boldsymbol{z}]\boldsymbol{z}^\intercal - \boldsymbol{z}\,\mathrm{E}[\boldsymbol{z}]^\intercal + \mathrm{E}[\boldsymbol{z}]\,\mathrm{E}[\boldsymbol{z}]^\intercal\Big)\widetilde{\boldsymbol{\mu}}$$

The vector $\widetilde{\boldsymbol{\mu}}$ is not random, so by linearity of expectation,

$$\frac{n^2}{4}\,\mathrm{E}\big[(\widehat{\tau}-\tau)^2\big] = \widetilde{\boldsymbol{\mu}}^{\mathsf{T}}\Big(\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathsf{T}}] - \mathrm{E}[\boldsymbol{z}]\,\mathrm{E}[\boldsymbol{z}]^{\mathsf{T}}\Big)\widetilde{\boldsymbol{\mu}}.$$

The proof is completed by noting that $\mathrm{Cov}(\boldsymbol{z}) = \mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathsf{T}}] - \mathrm{E}[\boldsymbol{z}]\,\mathrm{E}[\boldsymbol{z}]^{\mathsf{T}}$. □

The main paper contains several results that are presented assuming uniform treatment probabilities: $\Pr(z_i = 1) = 1/2$ for all units $i \in [n]$. We did this only for expositional reasons, and all results can be extended to settings with non-uniform first-order assignment probabilities by replacing the definition of the average potential vector $\boldsymbol{\mu}$ with the more general definition used in the statements of Lemma A2 and Corollary A1. The one exception is the uniform bound on the second-order assignment probabilities in Section A4. We conjecture that also this result can be extended to non-uniform probabilities, but we have not proven this.

A2 Analysis of the Gram–Schmidt Walk algorithm

In this section, we restate the Gram–Schmidt Walk algorithm of Bansal et al. (2019) and present our analysis of the algorithm. We analyze the Gram–Schmidt Walk algorithm under more general conditions than what we consider in our analysis of the Gram–Schmidt Walk design. At the end of the section, we discuss how the analysis of the Gram–Schmidt Walk algorithm extends to the Gram–Schmidt Walk design.

We begin by restating the algorithm and introducing notation that will be used in the proofs. Next, we describe a formal connection to the Gram–Schmidt orthogonalization process which is also used in our proofs. We then provide proofs of the covariance bound (Theorem 1) and the subgaussian concentration (Theorem 4) of the Gram–Schmidt Walk algorithm. Finally, we discuss the extension of this analysis to the Gram–Schmidt Walk design.

A2.1 Gram–Schmidt Walk algorithm

In this section, we restate the Gram–Schmidt Walk algorithm using more detailed notation. This more detailed notation contains explicit references to the iteration index and will be used in the proofs in this supplement. Algorithm 1 below is the Gram–Schmidt Walk algorithm of Bansal et al. (2019). Randomizing the choice of pivots is not necessary for the algorithm or the analysis presented here, so we defer randomization of pivots to the discussion of the Gram–Schmidt Walk design in Section A2.5. The algorithm presented in Section 4 sets the initial point $\boldsymbol{z}_1 = \mathbf{0}$.

Algorithm 1: Gram–Schmidt Walk

Input : Vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n \in \mathbb{R}^m$ arranged as columns in the matrix $\boldsymbol{B}$ and an initial point $\boldsymbol{z}_1 \in [-1,1]^n$

Output: $\boldsymbol{z} \in \{\pm 1\}^n$

Set iteration index $t \leftarrow 1$ and alive set $\mathcal{A}_1 \leftarrow [n]$.

Set the first pivot $p_0 \leftarrow n$

while $\mathcal{A}_t \neq \emptyset$ **do**

  if $p_{t-1} \notin \mathcal{A}_t$ **then**

    Set the pivot p_t to the largest index in $\mathcal{A}_t$.

  else

    $p_t \leftarrow p_{t-1}$

  end

  Compute the step direction

$$\boldsymbol{u}_t \leftarrow \arg\min_{\boldsymbol{u} \in U} \|\boldsymbol{B}\boldsymbol{u}\|,$$

  where U is the set of all $\boldsymbol{u} \in \mathbb{R}^n$ such that $\boldsymbol{u}(p_t) = 1$ and $\boldsymbol{u}(i) = 0$ for all $i \notin \mathcal{A}_t$.

  Set $\delta_t^+ \leftarrow |\max \Delta|$ and $\delta_t^- \leftarrow |\min \Delta|$ where $\Delta = \{\delta \in \mathbb{R} : \boldsymbol{z}_t + \delta \boldsymbol{u}_t \in [-1,1]^n\}$.

  Set the step size δ_t at random according to

$$\delta_t \leftarrow \begin{cases} \delta_t^+ & \text{with probability } \delta_t^- / (\delta_t^+ + \delta_t^-), \\ -\delta_t^- & \text{with probability } \delta_t^+ / (\delta_t^+ + \delta_t^-). \end{cases}$$

  Update the fractional assignment $\boldsymbol{z}_{t+1} \leftarrow \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$

  Update set of alive units $\mathcal{A}_{t+1} \leftarrow \{i \in [n] \mid |\boldsymbol{z}_t(i)| < 1\}$

  Increment the iteration index $t \leftarrow t + 1$

end

return $\boldsymbol{z} \leftarrow$ the final iterate $\boldsymbol{z}_{T+1}$

We remark on some of the differences between the notation in Algorithm 1 here and the pseudo-code presented in the main body of the paper. First, the Gram–Schmidt Walk algorithm takes as input arbitrary vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n \in \mathbb{R}^m$. For purposes of analysis, we often assume that the ℓ_2 norms of these input vectors is at most 1. Second, in this version, which is identical to the algorithm developed by Bansal et al. (2019), we do not choose the pivots at random. In fact, the only source of randomness in Algorithm 1 is the choice of step size δ_t at each iteration. In Section A2.5, we demonstrate that selecting pivots uniformly at random from $\mathcal{A}_t$ is equivalent to randomly permuting the input order of the input vectors and running Algorithm 1. Finally, the notation presented here contains more reference to iteration indices. In particular, the notation of the pivot unit p_t, the alive set $\mathcal{A}_t$, and the choice of update steps δ_t^+, δ_t^- all feature the iteration index in the subscript. We also use the notation that $\boldsymbol{u}_t(i)$ denotes the ith coordinate of the vector $\boldsymbol{u}$ at time t.

We denote the (random) number of iterations by T. We now introduce a notational convention which improves the clarity of some further analysis. Because the number of iterations T is always at most n by Lemma 5, we may suppose that the algorithm runs for exactly n iterations and that for iterations $t > T$, we set the update direction $\boldsymbol{u}_t = \boldsymbol{0}$ and the step size $\delta_t = 0$. The same vector $\boldsymbol{z}$ is returned and the output distribution of the algorithm is unchanged. We remark that this convention is used sparingly throughout the analysis and does not change the algorithm.

The concept of pivot phases was central to the analysis in Bansal et al. (2019) and it remains a central part of the analysis presented here as well. For each unit $i \in [n]$, we define the *pivot phase* S_i to be the set of iterations for which unit i is the pivot, i.e.

$$S_i = \{t : p_t = i\}.$$

During a particular run of the algorithm, the pivot phase S_i may be empty if unit i is not chosen as a pivot unit during that run.

During the course of the algorithm, a unit $i \in [n]$ is said to be *alive* if $|\boldsymbol{z}_t(i)| < 1$ and *frozen* otherwise. This is the convention is used by Bansal et al. (2019) and it reflects that fact that once a unit is frozen, its fractional assignment becomes integral and it is no longer updated. The set $\mathcal{A}_t$ is referred to as the *alive set* because it contains all alive units at the beginning of iteration t. We refer to the vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ as the input vectors. We may slightly abuse our terminology and call an input vector $\boldsymbol{b}_i$ alive or frozen when we mean that the corresponding unit i is alive or frozen.

We say that a unit i is *decided by the algorithm* when it is either selected as the pivot (Lines 2 or 5) or frozen without being chosen as the pivot (Line 12). Throughout the proofs below, we often condition on the previous random decisions made by the algorithm. We use Δ_i to denote all the random decisions made by the algorithm up to and including when unit i was decided by the algorithm. There is, however, some care to be taken in this

definition to distinguish between units which are chosen as pivots and those which are not. If i is chosen as a pivot at the beginning of iteration t, then Δ_i includes all previous choices of step sizes $\delta_1 \dots \delta_{t-1}$. If i is frozen at the end of iteration t without being chosen as the pivot, then Δ_i includes all choices of step sizes $\delta_1 \dots \delta_t$. Other types of conditioning will be presented throughout the proofs as the needs arise.

A2.2 Connection to Gram–Schmidt orthogonalization

A key aspect in our analysis of the Gram–Schmdit Walk algorithm is a Gram–Schmidt orthogonalization applied to a random re-ordering of the input vectors. We use the randomized Gram–Schmidt orthogonalization to obtain the tight bounds on the covariance matrix and the subgaussian constant in Theorems 1 and 4, respectively. In this section, we describe this connection in detail, providing additional notation and several technical lemmas which will be used in the proofs of Theorems 1 and 4.

Before continuing, we make two remarks regarding the randomized Gram–Schmidt orthogonalization. First, we emphasize that this re-ordering and orthogonalization is only for the purposes of analysis and is not executed by the algorithm. We also remark that although Bansal et al. (2019) discuss how the Gram–Schmidt Walk algorithm was inspired by Gram–Schmidt orthogonalization, an explicit connection is not made in that paper. This is one of the technical differences in our analysis which allows us to obtain tighter bounds.

We begin this discussion by first describing the randomized re-ordering of the input vectors and then defining the Gram–Schmidt Orthogonalization processes applied to this re-ordering. Let us introduce the notation of the re-ordering. The inputs vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \dots \boldsymbol{b}_n \in \mathbb{R}^m$ will be re-ordered as

$$\boldsymbol{b}_{\boldsymbol{\sigma}(1)}, \boldsymbol{b}_{\boldsymbol{\sigma}(2)}, \dots \boldsymbol{b}_{\boldsymbol{\sigma}(n)} \ ,$$

where $\boldsymbol{\sigma}$ is a bijection mapping positions in the re-ordering to the units. Formally, $\boldsymbol{\sigma} : [n] \to [n]$ and to avoid confusion in this notation, we reserve the symbol r for a position in the re-ordering and the symbol i for a unit. In this way, we write $\boldsymbol{\sigma}(r) = i$ to mean that the rth position in the re-ordering is occupied by unit i. We may also refer to the position of a specific unit in the re-ordering using the inverse function $\boldsymbol{\sigma}^{-1}$. That is, $\boldsymbol{\sigma}^{-1}(i) = r$ means that the unit i is assigned to position r in the re-ordering.

The re-ordering we consider is random and it is defined by the random choices made in the algorithm. Recall that a unit i is decided by the algorithm when it is either selected as the pivot (Lines 2 or 5) or frozen without being chosen as the pivot (Line 12). The ordering of the units $\boldsymbol{\sigma}(1), \boldsymbol{\sigma}(2), \dots \boldsymbol{\sigma}(n)$ will be the *reverse order* in which they are decided, breaking ties arbitrarily. In this way, as the algorithm decides units at each iteration, the randomized re-ordering is determined in reverse order. For example, the first unit to be decided is the first pivot unit p_1 so that $\boldsymbol{\sigma}(n) = p_1 = n$. If a single unit $j \neq p_1$ is frozen in the first

iteration, then this is the next unit decided by the algorithm, in which case it is second to last in the re-ordering, i.e. $\boldsymbol{\sigma}(n-1) = j$. On the other hand, if only the pivot p_1 is frozen in the first iteration, the next unit decided by the algorithm is the next pivot, which is p_2. In this case, $\boldsymbol{\sigma}(n-1) = p_2$.

Next, we introduce the Gram–Schmidt orthogonalization process on this randomized re-ordering of the input vectors. The Gram–Schmidt orthogonalization process is a method to construct a sequence of orthonormal vectors which form a basis for the span of a given set of vectors. For our problem at hand, we denote this sequence of orthonormal basis vectors by

$$\boldsymbol{w}_{\boldsymbol{\sigma}(1)}, \boldsymbol{w}_{\boldsymbol{\sigma}(2)}, \dots \boldsymbol{w}_{\boldsymbol{\sigma}(n)}.$$

They are recursively defined by the Gram–Schmidt orthogonalization process

$$\boldsymbol{w}_{\boldsymbol{\sigma}(1)} = \frac{\boldsymbol{b}_{\boldsymbol{\sigma}(1)}}{\|\boldsymbol{b}_{\boldsymbol{\sigma}(1)}\|} \quad \text{and} \quad \boldsymbol{w}_{\boldsymbol{\sigma}(r)} = \frac{\boldsymbol{b}_{\boldsymbol{\sigma}(r)} - \boldsymbol{A}_r \boldsymbol{b}_{\boldsymbol{\sigma}(r)}}{\left\|\boldsymbol{b}_{\boldsymbol{\sigma}(r)} - \boldsymbol{A}_r \boldsymbol{b}_{\boldsymbol{\sigma}(r)}\right\|} \quad \text{for } r = 2, \dots n,$$

where $\boldsymbol{A}_r = \sum_{s<r} \boldsymbol{w}_{\boldsymbol{\sigma}(s)} \boldsymbol{w}_{\boldsymbol{\sigma}(s)}^{\intercal}$ is the projection onto the span of the first $r-1$ input vectors $\boldsymbol{b}_{\boldsymbol{\sigma}(1)} \dots \boldsymbol{b}_{\boldsymbol{\sigma}(r-1)}$. Because the random re-ordering of the input vectors is determined by the random choices of $\delta_1 \dots \delta_n$ in the algorithm, the random sequence $\boldsymbol{w}_{\boldsymbol{\sigma}(1)} \dots \boldsymbol{w}_{\boldsymbol{\sigma}(n)}$ is also determined by the random choices made by the algorithm. Regardless of the randomization, this sequence of vectors forms an orthonormal basis for the span of the input vectors. Moreover, while the vector $\boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ depends on the set of vectors $\left\{\boldsymbol{b}_{\boldsymbol{\sigma}(1)}, \dots, \boldsymbol{b}_{\boldsymbol{\sigma}(r-1)}\right\}$, it does not depend on their order. For further reading on the Gram–Schmidt orthogonalization process, we refer readers to Chapter 4 of Strang (2009).

The main benefit of using this Gram–Schmidt orthogonalization process is that we can cleanly analyze the behavior of the algorithm within pivot phases. In particular, it provides a way to partition the span of the input vectors into orthogonal subspaces $V_1, V_2, \dots V_n$ corresponding to each of the n units. These subspaces are defined by the algorithm's random choices within the corresponding unit's pivot phase. We begin by defining the subspaces for units that are chosen as pivots. Let i be a unit which is chosen as pivot and assume it has position $r = \boldsymbol{\sigma}^{-1}(i)$ in the reordering so that the $k+1$ vectors which are decided during this pivot phase appear in the ordering as $\boldsymbol{b}_{\boldsymbol{\sigma}(r-k)}, \boldsymbol{b}_{\boldsymbol{\sigma}(r-k+1)}, \dots \boldsymbol{b}_{\boldsymbol{\sigma}(r)}$. The subspace $V_i \subset \mathbb{R}^m$ is defined to be the span of the vectors $\boldsymbol{b}_{\boldsymbol{\sigma}(r-k)}, \boldsymbol{b}_{\boldsymbol{\sigma}(r-k+1)}, \dots \boldsymbol{b}_{\boldsymbol{\sigma}(r)}$ after they have been projected orthogonal to $\boldsymbol{b}_{\boldsymbol{\sigma}(1)}, \boldsymbol{b}_{\boldsymbol{\sigma}(2)}, \dots, \boldsymbol{b}_{\boldsymbol{\sigma}(r-k-1)}$. As the set $\{\boldsymbol{\sigma}(1), \dots, \boldsymbol{\sigma}(r-k-1)\}$ is determined at this time, the projection is well-defined. The vectors

$$\boldsymbol{w}_{\boldsymbol{\sigma}(r-k)}, \boldsymbol{w}_{\boldsymbol{\sigma}(r-k+1)}, \dots, \boldsymbol{w}_{\boldsymbol{\sigma}(r)}$$

form an orthonormal basis for the subspace V_i and the projection matrix onto this subspace

is

$$\boldsymbol{P}_i = \sum_{s=0}^{k} \boldsymbol{w}_{\boldsymbol{\sigma}(r-s)} \boldsymbol{w}_{\boldsymbol{\sigma}(r-s)}^{\mathsf{T}}.$$

If a unit i is never chosen as a pivot unit, then V_i is the zero subspace and so the projection matrix P_i is the zero matrix. We remark that these subspaces and projection matrices are the ones referenced in the proof sketches of Theorems 1 and 4.

The following lemma follows directly from the definition of the subspaces but may also be verified by orthonormality of the vector sequence produced by Gram–Schmidt orthogonalization.

Lemma A3. *The subspaces are $V_1, V_2, \ldots V_n$ are orthogonal and their union is* $\operatorname{span}\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n\}$. *Equivalently, the corresponding projection matrices $\boldsymbol{P}_1 \ldots \boldsymbol{P}_n$ satisfy*

$$\sum_{i=1}^{n} \boldsymbol{P}_i = \boldsymbol{P},$$

where $\boldsymbol{P}$ is the projection matrix onto $\operatorname{span}\{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n\}$.

Next, we will show that the fractional balance update $\boldsymbol{B}\boldsymbol{u}_t$ is contained in the subspace corresponding to the current pivot, V_{p_t}. We will show a stronger property, but in order to make these statements precise, we need additional notation which connects an iteration t with the re-ordered positions of the units that have already been decided during in the current pivot phase. We define ℓ_t and g_t to be the least and greatest re-ordering positions that were decided during the current pivot phase before Line 9 at iteration t. The first unit to be decided in any pivot phase is the pivot unit. Thus the greatest re-ordering position of any unit which was decided during the current pivot phase is $g_t = \boldsymbol{\sigma}^{-1}(p_t)$. Note that when we arrive at Line 9, $\mathcal{A}_t \setminus p_t$ is the set of units which have not yet been decided. Thus, these are the units which will appear earliest in the re-ordering (although their ordering is not yet determined) and so we have that $\ell_t = |\mathcal{A}_t \setminus p_t| + 1 = |\mathcal{A}_t|$. In the first iteration of a pivot phase, we have $\ell_t = g_t$ because only the pivot has been decided before Line 9 at this iteration.

Using this notation, at Line 9 of iteration t, the input vectors whose units have been decided during the current pivot phase are

$$\boldsymbol{b}_{\boldsymbol{\sigma}(\ell_t)}, \boldsymbol{b}_{\boldsymbol{\sigma}(\ell_t+1)}, \ldots \boldsymbol{b}_{\boldsymbol{\sigma}(g_t)}.$$

The next lemma demonstrates that the fractional update $\boldsymbol{B}\boldsymbol{u}_t$ is the projection of the pivot onto the subspace spanned by $\boldsymbol{w}_{\boldsymbol{\sigma}(\ell_t)}, \boldsymbol{w}_{\boldsymbol{\sigma}(\ell_t+1)}, \ldots \boldsymbol{w}_{\boldsymbol{\sigma}(g_t)}$.

Lemma A4. *At each iteration t, we can write $\boldsymbol{B}\boldsymbol{u}_t$ in the orthonormal basis $\boldsymbol{w}_{\boldsymbol{\sigma}(1)} \ldots \boldsymbol{w}_{\boldsymbol{\sigma}(n)}$*

as

$$\boldsymbol{B}\boldsymbol{u}_t = \sum_{r=\ell_t}^{g_t} \big\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} \big\rangle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}.$$

Proof. Recall that the step direction $\boldsymbol{u}_t$ is determined by a least squares problem. That is, the undecided coordinates of the step direction, $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)$, are the minimizers of the least squares program

$$\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = \operatorname*{arg\,min}_{u_i : i \in \mathcal{A}_t \setminus p_t} \left\| \boldsymbol{b}_{p_t} + \sum_{i \in \mathcal{A}_t \setminus p_t} u_i \boldsymbol{b}_i \right\|^2.$$

Because the step direction is the minimizer, it must satisfy the normal equations

$$\boldsymbol{B}\boldsymbol{u}_t = \boldsymbol{b}_{p_t} - \boldsymbol{A}_t \boldsymbol{b}_{p_t},$$

where $\boldsymbol{A}_t$ is the projection matrix onto the span of the alive vectors which are not the pivot. That is, $\boldsymbol{b}_i$ for i in $\mathcal{A}_t \setminus p_t = \{\boldsymbol{\sigma}(1), \ldots, \boldsymbol{\sigma}(\ell_t) - 1\}$. By the construction of the re-ordering and the Gram–Schmidt orthogonalization, we have that $\boldsymbol{A}_t = \sum_{s<\ell_t} \boldsymbol{w}_{\boldsymbol{\sigma}(s)} \boldsymbol{w}_{\boldsymbol{\sigma}(s)}^{\mathsf{T}}$. Writing the fractional balance update $\boldsymbol{B}\boldsymbol{u}_t$ in the orthonormal basis, we have that

$$\begin{aligned}
\boldsymbol{B}\boldsymbol{u}_t &= \sum_{r=1}^{n} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{B}\boldsymbol{u}_t \rangle \boldsymbol{w}_{\boldsymbol{\sigma}(r)} && \text{(orthonormal basis)} \\
&= \sum_{r=1}^{n} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} - \boldsymbol{A}_t \boldsymbol{b}_{p_t} \rangle \boldsymbol{w}_{\boldsymbol{\sigma}(r)} && \text{(normal equations)} \\
&= \sum_{r=1}^{n} \Big[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} \rangle - \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{A}_t \boldsymbol{b}_{p_t} \rangle \Big] \boldsymbol{w}_{\boldsymbol{\sigma}(r)} && \text{(linearity)} \\
&= \sum_{r=1}^{n} \Big[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} \rangle - \langle \boldsymbol{A}_t \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} \rangle \Big] \boldsymbol{w}_{\boldsymbol{\sigma}(r)}. && \text{(projection matrix, } \boldsymbol{A}_t^{\mathsf{T}} = \boldsymbol{A}_t)
\end{aligned}$$

We now examine each term in this sum. If $r < \ell_t$ then $\boldsymbol{A}_t \boldsymbol{w}_{\boldsymbol{\sigma}(r)} = \boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ because $\boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ is a vector in the subspace associated with the projection $\boldsymbol{A}_t$. Thus, the two terms in the bracket are the same, so the terms corresponding to $r < \ell_t$ are zero and do not contribute to the sum. If $r \geq \ell_t$, then by the construction of the re-ordering and Gram–Schmidt orthogonalization, $\boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ is orthogonal to the subspace corresponding to $\boldsymbol{A}_t$ and so $\boldsymbol{A}_t \boldsymbol{w}_{\boldsymbol{\sigma}(r)} = 0$. This means that for $\ell_t \leq r \leq g_t$, the second term in the brackets is zero, and only the first term in brackets contributes to the sum. On the other hand, if $r > g_t$, then by the re-ordering and Gram–Schmidt orthogonalization, $\boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ is orthogonal to $\boldsymbol{b}_{\boldsymbol{\sigma}(g_t)} = \boldsymbol{b}_{p_t}$. In this case, both terms in the brackets are zero and the terms corresponding to $r > g_t$ contribute nothing to

the sum. Thus, we have shown that

$$\boldsymbol{B}\boldsymbol{u}_t = \sum_{r=\ell_t}^{g_t} \big\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_{p_t} \big\rangle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}. \qquad \square$$

A2.3 Covariance bound (Theorem 1)

This section contains a proof of an extended version of the covariance bound in Theorem 1. We begin by deriving a form of the covariance matrix of the assignment vector in terms of the update quantities in the algorithm.

Lemma A5. *The covariance matrix of the assignment vector is given by*

$$\mathrm{Cov}(\boldsymbol{z}) = \mathrm{E}\left[\sum_{t=1}^{T} \delta_t^2 \boldsymbol{u}_t \boldsymbol{u}_t^\intercal\right].$$

Proof. First, observe that

$$\mathrm{Cov}(\boldsymbol{z}) = \mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal] - \mathrm{E}[\boldsymbol{z}]\,\mathrm{E}[\boldsymbol{z}]^\intercal = \mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal] - \boldsymbol{z}_1\boldsymbol{z}_1^\intercal$$

where the second equality uses $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{z}_1$, which in a consequence of the martingale property (Lemma 2). By the update rule $\boldsymbol{z}_{t+1} \leftarrow \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$,

$$\boldsymbol{z}_{t+1}\boldsymbol{z}_{t+1}^\intercal = (\boldsymbol{z}_t + \delta_t\boldsymbol{u}_t)(\boldsymbol{z}_t + \delta_t\boldsymbol{u}_t)^\intercal = \boldsymbol{z}_t\boldsymbol{z}_t^\intercal + \delta_t\big(\boldsymbol{u}_t\boldsymbol{z}_t^\intercal + \boldsymbol{z}_t\boldsymbol{u}_t^\intercal\big) + \delta_t^2\boldsymbol{u}_t\boldsymbol{u}_t^\intercal.$$

Iteratively applying this over all iterations $t \in \{1, 2, \ldots\}$ and using that the returned vector is $\boldsymbol{z} = \boldsymbol{z}_{T+1}$, we have that

$$\boldsymbol{z}\boldsymbol{z}^\intercal = \boldsymbol{z}_{T+1}\boldsymbol{z}_{T+1}^\intercal = \boldsymbol{z}_1\boldsymbol{z}_1^\intercal + \sum_{t=1}^{T} \delta_t\big(\boldsymbol{u}_t\boldsymbol{z}_t^\intercal + \boldsymbol{z}_t\boldsymbol{u}_t^\intercal\big) + \sum_{t=1}^{T} \delta_t^2\boldsymbol{u}_t\boldsymbol{u}_t^\intercal.$$

Substituting this expression of $\boldsymbol{z}\boldsymbol{z}^\intercal$ into $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^\intercal]$ in the earlier covariance calculation, we obtain that

$$\mathrm{Cov}(\boldsymbol{z}) = \mathrm{E}\left[\sum_{t=1}^{T} \delta_t^2\boldsymbol{u}_t\boldsymbol{u}_t^\intercal\right] + \mathrm{E}\left[\sum_{t=1}^{T} \delta_t\big(\boldsymbol{u}_t\boldsymbol{z}_t^\intercal + \boldsymbol{z}_t\boldsymbol{u}_t^\intercal\big)\right] \tag{A1}$$

We will now show that the last term is zero because the step size δ_t is zero in expectation. By linearity of expectation and using the convention that the algorithm runs for n iterations

with $\delta_t = 0$ and $\boldsymbol{u}_t = \mathbf{0}$ for $t > T$,

$$\mathrm{E}\left[\sum_{t=1}^{T} \delta_t\big(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal\big)\right] = \sum_{t=1}^{n} \mathrm{E}\big[\delta_t\big(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal\big)\big]$$

For a fixed iteration t, consider the individual term $\mathrm{E}[\delta_t(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal)]$ in the sum above. Observe that if we condition on all previous random decisions made by the algorithm before step size δ_t is chosen (i.e. choices of step sizes $\delta_1 \ldots \delta_{t-1}$), then the step direction $\boldsymbol{u}_t$ and fractional assignment $\boldsymbol{z}_t$ are both determined, so that $\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal$ is a deterministic quantity. In this way, δ_t is conditionally independent of $\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal$ conditioned on all previous random decisions made by the algorithm. Using the fact that the expected step size δ_t is zero, we have that

$$\mathrm{E}[\delta_t\big(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal\big) \mid \delta_1 \ldots \delta_{t-1}] = \big(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal\big) \cdot \mathrm{E}[\delta_t \mid \delta_1 \ldots \delta_{t-1}] = 0$$

for all iterations t. By the law of total expectation, $\mathrm{E}[\delta_t(\boldsymbol{u}_t \boldsymbol{z}_t^\intercal + \boldsymbol{z}_t \boldsymbol{u}_t^\intercal)] = 0$ and so that the second term in (A1) is zero. □

Next, we prove a lemma stating that the expected sum of the squared step sizes in the remainder of a pivot phase is not too large in expectation. To do this, we introduce notation that connects a position in the re-ordering to the subsequent iterations in a pivot phase. For each position r in the re-ordering, we define

$$L_r = \{t : \ell_t \leq r \leq g_t\}.$$

The set L_r allows us to discuss what happens in the remaining iterations of a pivot phase after the unit in position r has been decided. For example, if a unit i is chosen as the pivot and assigned to position r, then L_r is the entire pivot phase S_i. If a non-pivot unit i is frozen and assigned to position r, then L_r are the remaining iterations in the pivot phase. Note that L_r may be empty if a non-pivot unit is frozen along with pivot at the last iteration of the pivot phase. We are now ready to state a lemma on the expected sum of the squared step sizes throughout the remainder of a pivot phase.

Lemma A6. *For each $r \in [n]$, conditional on the random decisions made up until unit $\boldsymbol{\sigma}(r)$ is decided, the expected sum of squared step sizes in the remainder of its pivot phase is at most one. That is, for each unit $i \in [n]$ with re-ordering position $r = \boldsymbol{\sigma}^{-1}(i)$,*

$$\mathrm{E}\left[\sum_{t \in L_r} \delta_t^2 \,\middle|\, \Delta_{\boldsymbol{\sigma}(r)}\right] \leq 1.$$

Proof. Because only one pivot phase is being considered, we drop the iteration subscripts

here and write the pivot as p. Recall that $\Delta_{\boldsymbol{\sigma}(r)}$ denotes all the random decisions made by the algorithm up to and including when unit i was decided by the algorithm. If L_r is empty, then the statement is trivially true. Otherwise, L_r is a (random) contiguous set of iterations $t_0, t_0+1, \ldots t_0+k$, where t_0+k is the last iteration in the pivot phase. Because the pivot phase terminates when the pivot p is frozen, $|\boldsymbol{z}_{t_0+k}(p)| = 1$. It follows that

$$\begin{aligned}
1 - \boldsymbol{z}_{t_0}(p)^2 &= \boldsymbol{z}_{t_0+k}(p)^2 - \boldsymbol{z}_{t_0}(p)^2 && (|\boldsymbol{z}_{t_0+k}(p)| = 1) \\
&= \sum_{s=0}^{k-1} \left[\boldsymbol{z}_{t_0+s+1}(p)^2 - \boldsymbol{z}_{t_0+s}(p)^2\right] && \text{(telescoping sum)} \\
&= \sum_{s=0}^{k-1} \left[\left(\boldsymbol{z}_{t_0+s}(p) + \delta_{t_0+s}\boldsymbol{u}_{t_0+s}(p)\right)^2 - \boldsymbol{z}_{t_0+s}(p)^2\right] && \text{(update rule)} \\
&= \sum_{s=0}^{k-1} \left[\delta_{t_0+s}^2 \boldsymbol{u}_{t_0+s}(p)^2 + 2\delta_{t_0+s}\boldsymbol{u}_{t_0+s}(p)\boldsymbol{z}_{t_0+s}(p)\right] && \text{(cancelling terms)} \\
&= \sum_{t\in L_r} \left[\delta_t^2 \boldsymbol{u}_t(p)^2 + 2\delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p)\right] \ .
\end{aligned}$$

Taking conditional expectations of both sides and using linearity of expectation, we have that

$$1 - \boldsymbol{z}_{t_0}(p)^2 = \mathrm{E}\left[\sum_{t\in L_r} \delta_t^2 \;\middle|\; \Delta_{\boldsymbol{\sigma}(r)}\right] + 2\,\mathrm{E}\left[\sum_{t\in L_r} \delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \;\middle|\; \Delta_{\boldsymbol{\sigma}(r)}\right], \tag{A2}$$

because the left hand side is a deterministic quantity under this conditioning. We now seek to show that the second term on the right hand side is zero. To this end, observe that we may extend the sum from iterations $t \in L_r$ to all remaining iterations because $\boldsymbol{u}_t(p) = 0$ for iterations t after the current pivot phase, i.e.,

$$\mathrm{E}\left[\sum_{t\in L_r} \delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \;\middle|\; \Delta_{\boldsymbol{\sigma}(r)}\right] = \mathrm{E}\left[\sum_{t\geq t_0} \delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \;\middle|\; \Delta_{\boldsymbol{\sigma}(r)}\right] = \sum_{t\geq t_0} \mathrm{E}\Big[\delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \;\Big|\; \Delta_{\boldsymbol{\sigma}(r)}\Big].$$

We now show that each term $\mathrm{E}[\delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \mid \Delta_{\boldsymbol{\sigma}(r)}]$ is zero for each t. Suppose that we further condition on all previous random decisions made by the algorithm before step size δ_t is chosen. In this case, the quantity $\boldsymbol{u}_t(p)\boldsymbol{z}_t(p)$ is completely determined and so δ_t is independent of $\boldsymbol{u}_t(p)\boldsymbol{z}_t(p)$. Moreover, the step size has mean zero, as shown in the proof of Lemma 2. Thus, for $t \geq t_0$,

$$\mathrm{E}[\delta_t \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \mid \delta_1 \ldots \delta_{t-1}] = \boldsymbol{u}_t(p)\boldsymbol{z}_t(p) \cdot \mathrm{E}[\delta_t \mid \delta_1 \ldots \delta_{t-1}] = 0$$

By the law of total expectation, it follows that the term $\mathrm{E}[\delta_t \boldsymbol{z}_t(p) \mid \Delta_{\boldsymbol{\sigma}(r)}]$ is zero for $t \geq t_0$.

Thus, the second term in (A2) is zero and so we have that

$$\mathrm{E}\left[\sum_{t\in L_r}\delta_t^2 \;\middle|\; \Delta_{\boldsymbol{\sigma}(r)}\right] = 1 - \boldsymbol{z}_{t_0}(p)^2 \le 1,$$

where the inequality follows from $\boldsymbol{z}_{t_0}(p) \in (-1, 1)$. □

At this point, we are ready to prove the covariance bound.

Theorem 1*. *If all input vectors $\boldsymbol{b}_1 \dots \boldsymbol{b}_n$ have ℓ_2 norm at most one, then the covariance matrix of the vector of imbalances $\boldsymbol{Bz}$ is bounded in the Loewner order by the orthogonal projection onto the subspace spanned by the columns of $\boldsymbol{B}$:*

$$\mathrm{Cov}(\boldsymbol{Bz}) \preceq \boldsymbol{P} = \boldsymbol{B}\big(\boldsymbol{B}^\intercal\boldsymbol{B}\big)^\dagger\boldsymbol{B}^\intercal,$$

where we recall that $\boldsymbol{A}^\dagger$ denotes the pseudoinverse of the matrix $\boldsymbol{A}$.

Proof. To prove the matrix inequality in the statement of the theorem, we seek to show that

$$\boldsymbol{v}^\intercal \mathrm{Cov}(\boldsymbol{Bz})\boldsymbol{v} \le \boldsymbol{v}^\intercal\boldsymbol{P}\boldsymbol{v} \qquad \text{for all } \boldsymbol{v}\in\mathbb{R}^m$$

Using Lemma A5 for the form of $\mathrm{Cov}(\boldsymbol{z})$ and linearity of expectation, we have that

$$\boldsymbol{v}^\intercal \mathrm{Cov}(\boldsymbol{Bz})\boldsymbol{v} = \boldsymbol{v}^\intercal\boldsymbol{B}\,\mathrm{Cov}(\boldsymbol{z})\boldsymbol{B}^\intercal\boldsymbol{v} = \boldsymbol{v}^\intercal\boldsymbol{B}\,\mathrm{E}\left[\sum_{t=1}^{T}\delta_t^2\boldsymbol{u}_t\boldsymbol{u}_t^\intercal\right]\boldsymbol{B}^\intercal\boldsymbol{v} = \mathrm{E}\left[\sum_{t=1}^{T}\delta_t^2\langle\boldsymbol{Bu}_t,\boldsymbol{v}\rangle^2\right].$$

Thus, we seek to show that for all $\boldsymbol{v}\in\mathbb{R}^m$,

$$\mathrm{E}\left[\sum_{t=1}^{T}\delta_t^2\langle\boldsymbol{Bu}_t,\boldsymbol{v}\rangle^2\right] \le \boldsymbol{v}^\intercal\boldsymbol{P}\boldsymbol{v}.$$

Next, we compute an upper bound on the quadratic forms in the sum. For each iteration t,

$$\begin{aligned}
\langle\boldsymbol{Bu}_t,\boldsymbol{v}\rangle^2 &= \left\langle\sum_{r=\ell_t}^{g_t}\big\langle\boldsymbol{w}_{\boldsymbol{\sigma}(i)},\boldsymbol{b}_{p_t}\big\rangle\boldsymbol{w}_{\boldsymbol{\sigma}(i)},\boldsymbol{v}\right\rangle^2 && \text{(Lemma A4)}\\
&= \left(\sum_{r=\ell_t}^{g_t}\big\langle\boldsymbol{w}_{\boldsymbol{\sigma}(r)},\boldsymbol{b}_{p_t}\big\rangle\langle\boldsymbol{w}_{\boldsymbol{\sigma}(r)},\boldsymbol{v}\rangle\right)^2 && \text{(linearity)}\\
&\le \left(\sum_{r=\ell_t}^{g_t}\big\langle\boldsymbol{w}_{\boldsymbol{\sigma}(r)},\boldsymbol{b}_{p_t}\big\rangle^2\right)\left(\sum_{r=\ell_t}^{g_t}\langle\boldsymbol{w}_{\boldsymbol{\sigma}(r)},\boldsymbol{v}\rangle^2\right) && \text{(Cauchy–Schwarz)}
\end{aligned}$$

$$\leq \|\boldsymbol{b}_{p_t}\|^2 \cdot \left(\sum_{r=\ell_t}^{g_t} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2\right) \qquad (\boldsymbol{w}_{\boldsymbol{\sigma}(r)} \text{ are orthonormal})$$

$$\leq \left(\sum_{r=\ell_t}^{g_t} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2\right). \qquad (\text{by assumption, } \|\boldsymbol{b}_{p_t}\|^2 \leq 1)$$

Using this upper bound, we obtain an upper bound for the expected quantity of interest,

$$\begin{aligned}
\mathrm{E}\left[\sum_{t=1}^{T} \delta_t^2 \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right] &\leq \mathrm{E}\left[\sum_{t=1}^{T} \delta_t^2 \left(\sum_{r=\ell_t}^{g_t} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2\right)\right] && \text{(from above)} \\
&= \mathrm{E}\left[\sum_{r=1}^{n} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2 \sum_{t\in L_r} \delta_t^2\right] && \text{(rearranging terms)} \\
&= \sum_{r=1}^{n} \mathrm{E}\left[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2 \sum_{t\in L_r} \delta_t^2\right] && \text{(linearity of expectation)}
\end{aligned}$$

We examine each of the terms in this sum. Fix a position r in the random re-ordering. Suppose that we further condition on $\Delta_{\boldsymbol{\sigma}(r)}$, which contains all random decisions made by the algorithm up to and including when unit $\boldsymbol{\sigma}(r)$ was decided by the algorithm. Under this conditioning, the vector $\boldsymbol{w}_{\boldsymbol{\sigma}(r)}$ is completely determined and so the quantity $\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2$ is also completely determined. In this way, the random term $\sum_{t\in L_r} \delta_t^2$ is conditionally independent of $\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2$ given $\Delta_{\boldsymbol{\sigma}(r)}$. Thus, we have that

$$\mathrm{E}\left[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2 \sum_{t\in L_r} \delta_t^2 \middle| \Delta_{\boldsymbol{\sigma}(r)}\right] = \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2 \cdot \mathrm{E}\left[\sum_{t\in L_r} \delta_t^2 \middle| \Delta_{\boldsymbol{\sigma}(r)}\right] \leq \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2,$$

where the equality is due to conditional independence and the inequality follows from Lemma A6. Using iterated expectation, it follows that

$$\mathrm{E}\left[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2 \sum_{t\in L_r} \delta_t^2\right] \leq \mathrm{E}\Big[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2\Big].$$

Substituting this bound and using linearity of expectation yields

$$\mathrm{E}\left[\sum_{t=1}^{T} \delta_t^2 \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle^2\right] \leq \sum_{r=1}^{n} \mathrm{E}\Big[\langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v}\rangle^2\Big] = \boldsymbol{v}^\mathsf{T} \mathrm{E}\left[\sum_{r=1}^{n} \boldsymbol{w}_{\boldsymbol{\sigma}(r)} \boldsymbol{w}_{\boldsymbol{\sigma}(r)}^\mathsf{T}\right] \boldsymbol{v} = \boldsymbol{v}^\mathsf{T} \boldsymbol{P} \boldsymbol{v} \ ,$$

where the last equality follows from the fact that the vectors $\boldsymbol{w}_{\boldsymbol{\sigma}(1)}, \boldsymbol{w}_{\boldsymbol{\sigma}(2)}, \ldots, \boldsymbol{w}_{\boldsymbol{\sigma}(n)}$ form

an orthonormal basis for the span of input vectors, thus $\sum_{r=1}^{n} \boldsymbol{w}_{\boldsymbol{\sigma}(r)} \boldsymbol{w}_{\boldsymbol{\sigma}(r)}^{\intercal} = \boldsymbol{P}$ holds deterministically, regardless of the randomized re-ordering. □

A2.4 Subgaussian bound (Theorem 4)

In this section, we prove an extended version of the subgaussian concentration inequality of Theorem 4. We begin by presenting the main technical inequality (Lemma A7) which is stated in terms of operator monotonicity and proved using basic calculus. Next, we present Lemma A8, which analyzes the behavior of the Gram–Schmidt Walk algorithm in one pivot phase using a backwards induction style argument. Finally, we prove the subgaussian concentration inequality by showing how we may repeatedly apply Lemma A8.

The main technical inequality is stated in terms of operator monotonicity, which we briefly describe here. Let $\mathcal{D}$ be a set of n-by-n symmetric matrices. A real-valued matrix function $f : \mathcal{D} \to \mathbb{R}$ is said to be *operator monotone increasing* if

$$\boldsymbol{A}, \boldsymbol{B} \in \mathcal{D} \text{ with } \boldsymbol{A} \preceq \boldsymbol{B} \Rightarrow f(\boldsymbol{A}) \leq f(\boldsymbol{B}).$$

Intuitively, a real-valued matrix function f is monotone increasing if "larger" matrices (as determined by the Loewner order) are assigned larger values. We say that f is *operator monotone decreasing* if $\boldsymbol{A} \preceq \boldsymbol{B}$ implies instead that $f(\boldsymbol{A}) \geq f(\boldsymbol{B})$. Although there is a well developed theory of operator monotonicity, we use only very basic facts here which are mostly self contained. For more information on operator monotonicity, we refer readers to Chapter 5 of Bhatia (1997).

Lemma A7. *For all $x \in [-1, 1]$ the function*

$$f_x \begin{pmatrix} \alpha & \eta \\ \eta & \beta \end{pmatrix} = \exp\left(-\frac{1}{2}\alpha\beta\right)\left[\frac{1+x}{2}\exp((1-x)\eta) + \frac{1-x}{2}\exp(-(1+x)\eta)\right]$$

is operator monotone decreasing over the set of 2-by-2 positive semidefinite matrices.

Proof. Operator monotonicity of a function $g : \mathcal{D} \to \mathbb{R}$ is preserved under composition with any monotone increasing $h : \mathbb{R} \to \mathbb{R}$. Using this and observing that f_x takes positive values for $x \in [-1, 1]$, we have that f_x is operator monotone decreasing if and only if $\log f_x$ is operator monotone decreasing. Moreover, a differentiable function $g : \mathcal{D} \to \mathbb{R}$ is operator monotone decreasing if and only if $-\nabla g(\boldsymbol{A})$ is positive semidefinite for all $\boldsymbol{A} \in \mathcal{D}$. The function f_x under consideration is differentiable and thus, to prove the lemma, it suffices to show that

$$-\nabla \log f_x \begin{pmatrix} \alpha & \eta \\ \eta & \beta \end{pmatrix}$$

is positive semidefinite when the 2-by-2 input matrix is positive semidefinite, i.e., $\alpha, \beta \geq 0$ and $\alpha\beta \geq \eta^2$.

We begin by defining the shorthand

$$\psi_x(\eta) = \log\left[\frac{1+x}{2}\exp((1-x)\eta) + \frac{1-x}{2}\exp(-(1+x)\eta)\right]$$

for the log of the bracketed term in the definition of f_x. Using this, we may write the function $\log f_x$ as

$$\log f_x \begin{pmatrix} \alpha & \eta \\ \eta & \beta \end{pmatrix} = \psi_x(\eta) - \frac{1}{2}\alpha\beta.$$

From the above expression, it is clear that $\partial_\alpha \log f_x = -\beta/2$, $\partial_\beta \log f_x = -\alpha/2$, and $\partial_\eta \log f_x = \partial_\eta \psi_x$. Thus, the matrix gradient may be computed:

$$-2\nabla \log f_x = \begin{pmatrix} \beta & -\partial_\eta \psi_x(\eta) \\ -\partial_\eta \psi_x(\eta) & \alpha \end{pmatrix}.$$

Recall that when computing the matrix gradient, we scale the off diagonals by $1/2$, as they appear twice in the trace inner product. We seek to show that the matrix above is positive semidefinite when the input matrix is positive semidefinite. Because the matrix above is 2-by-2, proving that it is positive semidefinite is equivalent to showing the three inequalities $\alpha, \beta \geq 0$ and $\alpha\beta \geq (\partial_\eta \psi_x(\eta))^2$. Because the input matrix is positive semidefinite, we already have that $\alpha, \beta \geq 0$. To show the final inequality, we show in the next part of the proof that $\eta^2 \geq (\partial_\eta \psi_x(\eta))^2$. Because the input matrix already satisifes $\alpha\beta \geq \eta^2$, this will imply the final inequality.

So for the final part of the proof, we focus on showing the inequality

$$(\partial_\eta \psi_x(\eta))^2 \leq \eta^2 \quad \text{for all } x \in [-1, 1].$$

To this end, we use an enveloping argument to show that $|\partial_\eta \psi_x(\eta)| \leq |\eta|$ for all $x \in [-1, 1]$. We begin by computing the first and second derivatives of $\psi_x(\eta)$. First, we rewrite the function $\psi_x(\eta)$ as

$$\begin{aligned}
\psi_x(\eta) &= \log\left[\frac{1+x}{2}\exp(1-x)\eta + \frac{1-x}{2}\exp -(1+x)\eta\right] \\
&= \log\left[\frac{1}{2}\left(e^{\eta - x\eta} + xe^{\eta - x\eta} + e^{-\eta - x\eta} - xe^{-\eta - x\eta}\right)\right] \\
&= \log\left[\frac{e^{-x\eta}}{2}(e^{\eta} + xe^{\eta} + e^{-\eta} - xe^{-\eta})\right]
\end{aligned}$$

$$= \log\left[\frac{1}{2}(e^{\eta} + xe^{\eta} + e^{-\eta} - xe^{-\eta})\right] - x\eta$$
$$= \log[\cosh(\eta) + x\sinh(\eta)] - x\eta.$$

Next, we compute the derivative $\partial_\eta \psi_x(\eta)$ by using chain rule and derivatives of log and hyperbolic trigonometric functions:

$$\partial_\eta \psi_x(\eta) = \frac{\sinh(\eta) + x\cosh(\eta)}{\cosh(\eta) + x\sinh(\eta)} - x.$$

Finally, we compute the second derivative of $\psi_x(\eta)$ using the above result, the quotient rule, and derivatives for the hyperbolic functions:

$$\partial_\eta^2 \psi_x(\eta) = 1 - \left(\frac{\sinh(\eta) + x\cosh(\eta)}{\cosh(\eta) + x\sinh(\eta)}\right)^2 = 1 - (\partial_\eta \psi_x(\eta) + x)^2.$$

We now establish the basis of our enveloping argument. That is, we show that the second derivative of $\psi_x(\eta)$ is bounded above and below by

$$0 \leq \partial_\eta^2 \psi_x(\eta) \leq 1 \quad \text{for all} \quad \eta \in \mathbb{R} \quad \text{and} \quad x \in [-1, 1].$$

The upper bound is immediate from the earlier expression, as $\partial_\eta^2 \psi_x(\eta) = 1 - (\partial_\eta \psi_x(\eta) + x)^2 \leq 1$. The lower bound is a consequence of $x \in [-1, 1]$. To see this, observe that

$$\begin{aligned}
\partial_\eta^2 \psi_x(\eta) = 1 - \left(\frac{\sinh(\eta) + x\cosh(\eta)}{\cosh(\eta) + x\sinh(\eta)}\right)^2 &\geq 0 \\
&\Leftrightarrow (\cosh(\eta) + x\sinh(\eta))^2 \geq (\sinh(\eta) + x\cosh(\eta))^2 \\
&\Leftrightarrow \cosh^2(\eta) + x^2\sinh^2(\eta) \geq \sinh^2(\eta) + x^2\cosh^2(\eta) \\
&\Leftrightarrow \cosh^2(\eta) - \sinh^2(\eta) \geq x^2(\cosh^2(\eta) - \sinh^2(\eta)) \\
&\Leftrightarrow 1 \geq x^2
\end{aligned}$$

Now, we make our enveloping argument. First, we observe that $\partial_\eta \psi_x(0) = 0$. Next, for $\eta > 0$, we can bound the value of $\partial_\eta \psi_x(\eta)$ from above and below by

$$\partial_\eta \psi_x(\eta) = \partial_\eta \psi_x(0) + \int_{y=0}^{\eta} \partial_\eta^2 \psi_x(y) dy \leq 0 + \int_{y=0}^{\eta} 1 dy = \eta$$
$$\partial_\eta \psi_x(\eta) = \partial_\eta \psi_x(0) + \int_{y=0}^{\eta} \partial_\eta^2 \psi_x(y) dy \geq 0 + \int_{y=0}^{\eta} 0 dy = 0.$$

Written together, these inequalities state that $0 \leq \partial_\eta \psi_x(\eta) \leq \eta$ for values $\eta \geq 0$. A similar

enveloping argument shows that $-\eta \leq \partial_\eta \psi_x(\eta) \leq 0$ for values $\eta \leq 0$. Putting these two together, we have that $|\partial_\eta \psi_x(\eta)| \leq |\eta|$ for all $\eta \in \mathbb{R}$ and $x \in [-1, 1]$, as desired. □

Lemma A8. *Let p be a unit that is chosen as the pivot and let Δ_p denote all random decisions made by the algorithm up until the beginning of pivot phase p. If $\|\boldsymbol{b}_p\| \leq 1$, then for all $\boldsymbol{v} \in \mathbb{R}^m$,*

$$\mathrm{E}\left[\exp\left(\sum_{t \in S_p} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle - \frac{1}{2}\|\boldsymbol{P}_p \boldsymbol{b}_p\|^2 \cdot \|\boldsymbol{P}_p \boldsymbol{v}\|^2\right) \,\middle|\, \Delta_p\right] \leq 1,$$

where S_p is the set of iterations for which p is the pivot.

Proof. Let t_p be the iteration at which p is first chosen to be the pivot. This iteration t_p is a deterministic quantity conditioned on Δ_p.

We begin by describing a convention which we adopt for the purposes of this analysis. Recall that the number of iterations in a pivot phase is generally a random quantity; however, the number of iterations in a pivot phase is at most n by Lemma 5. In fact, because $t_p - 1$ iterations have already occurred, the number of iterations in the pivot phase S_p is at most $n - t_p + 1$. For the purposes of this proof, we adopt a convention which deterministically fixes the number of iterations within the pivot phase to be $n - t_p + 1$. We adopt this convention because fixing the number of iterations in a pivot phase to be a deterministic quantity simplifies our backwards induction style argument. Once the pivot is frozen at iteration t, all remaining iterations of the pivot phase $s > t$ have step size zero, i.e. $\delta_s = 0$. In this way, the fractional assignment is not updated in the remainder of the pivot phase after the pivot is frozen and thus this convention does not change the behavior of the algorithm. We emphasize again that this convention is for purposes of the current analysis and does not change the algorithm itself.

Using this convention and writing the iterations in the pivot phase as $S_p = \{t_p \dots n\}$, we seek to show that

$$\mathrm{E}\left[\exp\left(\sum_{t=t_p}^{n} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle - \frac{1}{2}\|\boldsymbol{P}_p \boldsymbol{b}_p\|^2 \cdot \|\boldsymbol{P}_p \boldsymbol{v}\|^2\right) \,\middle|\, \Delta_p\right] \leq 1. \tag{A3}$$

All expectations in the remainder of the proof are conditioned on Δ_p and so we drop this notation.

We now rewrite the terms in the exponent by using the sequence of orthonormal basis vectors produced by the Gram–Schmidt orthogonalization process, as described in Section A2.2. Suppose that the pivot unit has position $r = \boldsymbol{\sigma}^{-1}(p)$ in the reordering so that the $k + 1$ vectors which are decided during this pivot phase appear in the ordering as

$$\boldsymbol{b}_{\boldsymbol{\sigma}(r-k)}, \boldsymbol{b}_{\boldsymbol{\sigma}(r-k+1)}, \dots \boldsymbol{b}_{\boldsymbol{\sigma}(r)},$$

where the pivot vector is the last in this re-ordering, i.e., $\boldsymbol{\sigma}(r) = p$, and so $\boldsymbol{b}_{\boldsymbol{\sigma}(r)} = \boldsymbol{b}_p$. The corresponding basis vectors produced by the Gram–Schmidt orthogonalization are

$$\boldsymbol{w}_{\boldsymbol{\sigma}(r-k)}, \boldsymbol{w}_{\boldsymbol{\sigma}(r-k+1)}, \ldots \boldsymbol{w}_{\boldsymbol{\sigma}(r)}.$$

We now define a way to partition these reordering positions according to the iterations when they were decided. For each iteration $t = t_p, \ldots n$ in this pivot phase, we define Q_t to be the reordering positions of the units that are frozen during the fractional assignment update in Line 12 during iteration t. By our convention, it may happen that $\delta_t = 0$ and in this case, $Q_t = \emptyset$. We also define $Q_{t_p-1} = \{g_p\} = \{\boldsymbol{\sigma}^{-1}(p)\}$, which is the re-ordering index of the pivot. We remark that this reordering position is deterministic given the conditioning Δ_p and the subscript $t_p - 1$ is chosen for notational convenience. Note that the reordering positions are determined in the order $Q_{t_p-1}, Q_{t_p}, \ldots Q_n$ and this forms a partition of the reordering positions decided in this pivot phase.

Lemma A4 shows that for each iteration t,

$$\boldsymbol{B}\boldsymbol{u}_t = \sum_{s=t_p-1}^{t-1} \sum_{r \in Q_s} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_p \rangle \boldsymbol{w}_{\boldsymbol{\sigma}(r)} \quad \text{and so} \quad \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v} \rangle = \sum_{s=t_p-1}^{t-1} \sum_{r \in Q_s} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_p \rangle \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v} \rangle.$$

Recall that the projection matrix $\boldsymbol{P}_p$ is defined as

$$\boldsymbol{P}_p = \sum_{s=t_p-1}^{n} \sum_{r \in Q_s} \boldsymbol{w}_{\boldsymbol{\sigma}(r)} \boldsymbol{w}_{\boldsymbol{\sigma}(r)}^{\intercal}$$

and thus we have that

$$\|\boldsymbol{P}_p \boldsymbol{b}_p\|^2 = \sum_{s=t_p-1}^{n} \sum_{r \in Q_s} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_p \rangle^2 \quad \text{and} \quad \|\boldsymbol{P}_p \boldsymbol{v}\|^2 = \sum_{s=t_p-1}^{n} \sum_{r \in Q_s} \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v} \rangle^2$$

For notational convenience, for each reordering position r, let $\alpha_r = \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{b}_p \rangle$ and $\beta_r = \langle \boldsymbol{w}_{\boldsymbol{\sigma}(r)}, \boldsymbol{v} \rangle$.

Substituting these terms into (A3), we have that the desired inequality may be written as

$$\mathrm{E}\left[\exp\left(\sum_{t=t_p}^{n} \delta_t \sum_{s=t_p-1}^{t-1} \sum_{r \in Q_s} \alpha_r \beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{n} \sum_{r \in Q_s} \alpha_r^2\Big) \cdot \Big(\sum_{s=t_p-1}^{n} \sum_{r \in Q_s} \beta_r^2\Big)\right)\right] \leq 1.$$

We will prove this inequality using a backwards induction style argument. We use the main technical inequality of Lemma A7 to show that, conditioned on the first $n-1$ iterations, the expectation above is maximized when $\alpha_r = \beta_r = 0$ for all $r \in Q_n$. In some sense, this is identifying the worst-case values that $\{(\alpha_r, \beta_r) : r \in Q_n\}$ may take. We then

continue backwards and show that given the values of $\{(\alpha_r, \beta_r) : r \in Q_t\}$ for $t < R$, the values of $\{(\alpha_r, \beta_r) : r \in \cup_{s=R}^{n} Q_s\}$ which maximize the expectation are $\alpha_r = \beta_r = 0$.

We now proceed more formally. For each $R = 0, 1, \ldots n$, we define the quantity

$$g(R) = \mathrm{E}\left[\exp\left(\Big(\sum_{t=t_p}^{n} \delta_t \sum_{s=t_p-1}^{\min\{R,t-1\}} \sum_{r\in Q_s} \alpha_r\beta_r\Big) - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)\right)\right]$$

Note that $g(R)$ is similar to the expectation we are interested in bounding, except that $\alpha_r = \beta_r = 0$ for all $r \in \cup_{s>R} Q_s$. Note that $g(n)$ is exactly the expectation that we seek to upper bound by 1. We prove this upper bound by establishing the following chain of inequalities

$$g(n) \le g(n-1) \le \cdots \le g(t_p) \le 1.$$

We prove this chain of inequalities in three steps. The first step is to establish that $g(n) \le g(n-1)$. This inequality is the simplest one to establish because it follows directly from the definition of $g(R)$. In particular, observe that the term $\sum_{t=t_p}^{n} \delta_t \sum_{s=t_p-1}^{\min\{R,t-1\}} \sum_{r\in Q_s} \alpha_r\beta_r$ is the same for $R = n$ and $R = n-1$, while the term $\frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)$ is larger for $R = n$ than for $R = n-1$. Thus, $g(n) \le g(n-1)$.

We now show the second chunk of inequalities: $g(R) \le g(R-1)$ for $t_p < R \le n-1$. Before continuing, we show how to use the main technical inequality (Lemma A7) to prove that for all R in this range,

$$\begin{aligned}
&\mathrm{E}\left[\exp\left(\Big(\sum_{t=R+1}^{n} \delta_t \sum_{s=t_p-1}^{R} \sum_{r\in Q_s} \alpha_r\beta_r\Big) - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)\right)\,\middle|\,\Delta_R\right] \quad \text{(A4)}\\
&\le \mathrm{E}\left[\exp\left(\Big(\sum_{t=R+1}^{n} \delta_t \sum_{s=t_p-1}^{R-1} \sum_{r\in Q_s} \alpha_r\beta_r\Big) - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\beta_r^2\Big)\right)\,\middle|\,\Delta_R\right],
\end{aligned}$$

where Δ_R denotes the step sizes, $\delta_{t_p}, \delta_{t_p+1}, \ldots \delta_R$, in addition to the previous randomness in the algorithm denoted by Δ_p. Under this conditioning, the values of $\{(\alpha_r, \beta_r) : r \in \cup_{s=t_p-1}^{R} Q_s\}$ are decided and the only random quantity in the expression above is $\sum_{t=R+1}^{n} \delta_t$. We claim that this random variable is precisely

$$\sum_{t=R+1}^{n} \delta_t = \begin{cases} 1 - \boldsymbol{z}_{R+1}(p) & \text{with probability } (1 + \boldsymbol{z}_{R+1}(p))/2 \\ -(1 + \boldsymbol{z}_{R+1}(p)) & \text{with probability } (1 - \boldsymbol{z}_{R+1}(p))/2 \end{cases}$$

To see this, observe that because the step direction satisfies $\boldsymbol{u}_t(p) = 1$ in the pivot phase p

and the update procedure is $\boldsymbol{z}_{t+1} \leftarrow \boldsymbol{z}_t + \delta_t \boldsymbol{u}_t$,

$$\boldsymbol{z}_n(p) = \sum_{t=R+1}^{n} \delta_t \boldsymbol{u}_t(p) + \boldsymbol{z}_{R+1}(p) = \sum_{t=R+1}^{n} \delta_t + \boldsymbol{z}_{R+1}(p) \quad \text{and thus} \quad \sum_{t=R+1}^{n} \delta_t = \boldsymbol{z}_n(p) - \boldsymbol{z}_{R+1}(p).$$

Because $\boldsymbol{z}_n(p)$ takes values ± 1, we have that the sum $\sum_{t=R+1}^{n} \delta_t$ only takes two values. Moreover, because all step sizes have mean zero, we have that $\mathrm{E}[\sum_{t=R+1}^{n} \delta_t] = 0$. This determines the probabilities of each of the two values.

Because we know exactly the distribution of the random sum $\sum_{t=R+1}^{n} \delta_t$, we may derive the expectation in the left hand side of (A4) exactly as

$$\begin{aligned} &\frac{1 + \boldsymbol{z}_{R+1}(p)}{2} \exp\Bigg((1 - \boldsymbol{z}_{R+1}(p)) \sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \alpha_r \beta_r - \frac{1}{2} \Big(\sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \alpha_r^2 \Big) \cdot \Big(\sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \beta_r^2 \Big) \Bigg) \qquad \text{(A5)} \\ &+ \frac{1 - \boldsymbol{z}_{R+1}(p)}{2} \exp\Bigg(-(1 + \boldsymbol{z}_{R+1}(p)) \sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \alpha_r \beta_r - \frac{1}{2} \Big(\sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \alpha_r^2 \Big) \cdot \Big(\sum_{s=t_p-1}^{R} \sum_{r \in Q_s} \beta_r^2 \Big) \Bigg) \end{aligned}$$

We now demonstrate how this expectation may be recognized as the matrix function appearing in Lemma A7. Let $\boldsymbol{A}$ and $\boldsymbol{A}_R$ be the 2-by-2 matrices given by

$$\boldsymbol{A} = \sum_{s=t_p-1}^{R-1} \sum_{r \in Q_s} \begin{pmatrix} \alpha_r^2 & \alpha_r \beta_r \\ \alpha_r \beta_r & \beta_r^2 \end{pmatrix}, \quad \boldsymbol{A}_R = \sum_{r \in Q_R} \begin{pmatrix} \alpha_r^2 & \alpha_r \beta_r \\ \alpha_r \beta_r & \beta_r^2 \end{pmatrix}.$$

These matrices are the sum of 2-by-2 positive semidefinite matrices and so they are themselves positive semidefinite. Recall that the matrix function in Lemma A7 is defined for $x \in [-1, 1]$ as

$$\begin{aligned} f_x \begin{pmatrix} \alpha & \eta \\ \eta & \beta \end{pmatrix} &= e^{-\frac{1}{2}\alpha\beta} \left[\frac{1+x}{2} \exp((1-x)\eta) + \frac{1-x}{2} \exp(-(1+x)\eta) \right] \\ &= \frac{1+x}{2} \exp\Big((1-x)\eta - \frac{1}{2}\alpha\beta \Big) + \frac{1-x}{2} \exp\Big(-(1+x)\eta - \frac{1}{2}\alpha\beta \Big). \end{aligned}$$

Observe that the expectation in (A5) is equal to $f_{\boldsymbol{z}_R(p)}(\boldsymbol{A} + \boldsymbol{A}_R)$. By Lemma A7, the function is operator monotone decreasing over positive semidefinite matrices so that

$$f_{\boldsymbol{z}_R(p)}(\boldsymbol{A} + \boldsymbol{A}_R) \leq f_{\boldsymbol{z}_R(p)}(\boldsymbol{A}).$$

The proof of inequality (A4) is completed by observing that $f_{\boldsymbol{z}_R(p)}(\boldsymbol{A})$ is equal to the expectation on the right hand side of (A4).

Now we are ready to show that $g(R) \leq g(R-1)$ for $t_p < R \leq n-1$. For notational

convenience, we define

$$X_R = \exp\left(\sum_{t=t_p}^{R} \delta_t \sum_{s=t_p-1}^{t-1} \alpha_r \beta_r\right).$$

By rearranging terms, applying iterated expectations, and using the inequality (A4), we have that

$$\begin{aligned}
&g(R) \\
&= \mathrm{E}\left[\exp\left(\sum_{t=t_p}^{n} \delta_t \sum_{s=t_p-1}^{\min\{R,t-1\}} \sum_{r\in Q_s} \alpha_r\beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)\right)\right] \\
&= \mathrm{E}\left[X_R \cdot \exp\left(\sum_{t=R+1}^{n} \delta_t \sum_{s=t_p-1}^{R} \sum_{r\in Q_s} \alpha_r\beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)\right)\right] \\
&= \mathrm{E}\left[X_R \cdot \mathrm{E}\left[\exp\left(\sum_{t=R+1}^{n} \delta_t \sum_{s=t_p-1}^{R} \sum_{r\in Q_s} \alpha_r\beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R}\sum_{r\in Q_s}\beta_r^2\Big)\right)\,\middle|\,\Delta_R\right]\right] \\
&\le \mathrm{E}\left[X_R \cdot \mathrm{E}\left[\exp\left(\sum_{t=R+1}^{n} \delta_t \sum_{s=t_p-1}^{R-1} \sum_{r\in Q_s} \alpha_r\beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\beta_r^2\Big)\right)\,\middle|\,\Delta_R\right]\right] \\
&= \mathrm{E}\left[\exp\left(\sum_{t=t_p}^{n} \delta_t \sum_{s=t_p-1}^{\min\{R-1,t-1\}} \sum_{r\in Q_s} \alpha_r\beta_r - \frac{1}{2}\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\alpha_r^2\Big)\cdot\Big(\sum_{s=t_p-1}^{R-1}\sum_{r\in Q_s}\beta_r^2\Big)\right)\right] \\
&= g(R-1)
\end{aligned}$$

This establishes the chain of inequalities

$$g(n) \le g(n-1) \le \cdots \le g(t_p).$$

Establishing that $g(t_p) \le 1$ may be done via a similar application of the operator monotonicity result of Lemma A7. In particular,

$$\begin{aligned}
g(t_p) &= \mathrm{E}\left[\exp\left(\Big(\sum_{t=t_p}^{n}\delta_t\Big)\langle \boldsymbol{w}_p, \boldsymbol{b}_p\rangle\langle \boldsymbol{w}_p, \boldsymbol{v}\rangle - \frac{1}{2}\langle \boldsymbol{w}_p, \boldsymbol{b}_p\rangle^2\langle \boldsymbol{w}_p, \boldsymbol{v}\rangle^2\right)\right] \\
&= f_{\boldsymbol{z}_{t_p}(p)}\left(\begin{bmatrix}\langle \boldsymbol{w}_p, \boldsymbol{b}_p\rangle^2 & \langle \boldsymbol{w}_p, \boldsymbol{b}_p\rangle \\ \langle \boldsymbol{w}_p, \boldsymbol{b}_p\rangle & \langle \boldsymbol{w}_p, \boldsymbol{v}\rangle^2\end{bmatrix}\right) \\
&\le f_{\boldsymbol{z}_{t_p}(p)}(\mathbf{0}) = 1. \qquad \square
\end{aligned}$$

We now present the proof of the subgaussian concentration result.

Theorem 4*. *If the input vectors $\boldsymbol{b}_1 \dots \boldsymbol{b}_n$ all have ℓ_2 norm at most 1, then the Gram–Schmidt Walk algorithm returns an assignment vector $\boldsymbol{z}$ so that the vector of imbalances $\boldsymbol{Bz}$ is subgaussian with variance parameter $\sigma^2 = 1$:*

$$\mathrm{E}\Big[\exp\Big(\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle - \langle \mathrm{E}[\boldsymbol{Bz}], \boldsymbol{v}\rangle\Big)\Big] \leq \exp\big(\|\boldsymbol{v}\|^2/2\big) \qquad \textit{for all} \qquad \boldsymbol{v} \in \mathbb{R}^{n+d}.$$

Proof. We prove the stronger inequality

$$\mathrm{E}\Big[\exp\Big(\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle - \langle \mathrm{E}[\boldsymbol{Bz}], \boldsymbol{v}\rangle\Big)\Big] \leq \mathrm{E}\Big[\exp\Big(\frac{1}{2}\sum_{i=1}^{n} \|\boldsymbol{P}_i \boldsymbol{b}_i\|^2 \|\boldsymbol{P}_i \boldsymbol{v}\|^2\Big)\Big] \quad \text{for all} \quad \boldsymbol{v} \in \mathbb{R}^m. \tag{A6}$$

To see that inequality (A6) is stronger, we use the contractive property of projection matrices and the assumption that all input vectors have ℓ_2 norm at most 1 to show

$$\sum_{i=1}^{n} \|\boldsymbol{P}_i \boldsymbol{b}_i\|^2 \|\boldsymbol{P}_i \boldsymbol{v}\|^2 \leq \sum_{i=1}^{n} \|\boldsymbol{b}_i\|^2 \|\boldsymbol{P}_i \boldsymbol{v}\|^2 \leq \sum_{i=1}^{n} \|\boldsymbol{P}_i \boldsymbol{v}\|^2 = \|\boldsymbol{P}\boldsymbol{v}\|^2 \leq \|\boldsymbol{v}\|^2.$$

This shows that inequality (A6) implies the inequality in the statement of the theorem.

We now rearrange and substitute terms in (A6) to obtain a form that we will work with during the remainder of the proof. By dividing both sides of (A6) by the right hand side, we obtain an equivalent expression of the inequality:

$$\mathrm{E}\Big[\exp\Big(\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle - \langle \mathrm{E}[\boldsymbol{Bz}], \boldsymbol{v}\rangle - \frac{1}{2}\sum_{i=1}^{n} \|\boldsymbol{P}_i \boldsymbol{b}_i\|^2 \|\boldsymbol{P}_i \boldsymbol{v}\|^2\Big)\Big] \leq 1 \quad \text{for all} \quad \boldsymbol{v} \in \mathbb{R}^m.$$

At this point, we drop the "for all $\boldsymbol{v} \in \mathbb{R}^m$" qualifier and assume that an arbitrary $\boldsymbol{v} \in \mathbb{R}^m$ is given. We re-write the quantity $\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle - \langle \mathrm{E}[\boldsymbol{Bz}], \boldsymbol{v}\rangle$ in terms of the fractional updates in the algorithm:

$$\langle \boldsymbol{Bz}, \boldsymbol{v}\rangle = \Big\langle \boldsymbol{B}\Big(\sum_{t=1}^{T} \delta_t \boldsymbol{u}_t + \boldsymbol{z}_1\Big), \boldsymbol{v}\Big\rangle = \sum_{t=1}^{T} \delta_t \langle \boldsymbol{Bu}_t, \boldsymbol{v}\rangle + \langle \boldsymbol{Bz}_1, \boldsymbol{v}\rangle = \sum_{i=1}^{n} \sum_{t \in S_i} \delta_t \langle \boldsymbol{Bu}_t, \boldsymbol{v}\rangle + \langle \boldsymbol{Bz}_1, \boldsymbol{v}\rangle.$$

Note that by the martingale property of the fractional updates (Lemma 2), $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{z}_1$. Thus,

$$\langle \mathrm{E}[\boldsymbol{Bz}], \boldsymbol{v}\rangle = \langle \boldsymbol{B}\,\mathrm{E}[\boldsymbol{z}], \boldsymbol{v}\rangle = \langle \boldsymbol{Bz}_1, \boldsymbol{v}\rangle$$

and so the difference is given by

$$\langle \boldsymbol{B}\boldsymbol{z}, \boldsymbol{v}\rangle - \langle \mathrm{E}[\boldsymbol{B}\boldsymbol{z}], \boldsymbol{v}\rangle = \sum_{i=1}^{n} \sum_{t \in S_i} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle.$$

Using this expression for the difference, we may write the desired inequality, which features a sum over units in the exponent, as follows:

$$\mathrm{E}\left[\exp\left(\sum_{i=1}^{n}\Big(\sum_{t \in S_i} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle - \frac{1}{2}\|\boldsymbol{P}_i \boldsymbol{b}_i\|^2 \|\boldsymbol{P}_i \boldsymbol{v}\|^2\Big)\right)\right] \leq 1.$$

A unit $i \in [n]$ which is not chosen as the pivot does not contribute to this sum because the corresponding pivot phase S_i is empty and the projection matrix $\boldsymbol{P}_i$ is the zero. Thus, we may write the sum over units which are chosen as the pivot. We denote the sequence of pivot units as $p_1, p_2, \ldots p_k$ where the subscripts denote the order in which the pivots are chosen by the algorithm. We seek to show that

$$\mathrm{E}\left[\exp\left(\sum_{j=1}^{k}\Big(\sum_{t \in S_{p_j}} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle - \frac{1}{2}\|\boldsymbol{P}_{p_j} \boldsymbol{b}_{p_j}\|^2 \|\boldsymbol{P}_{p_j} \boldsymbol{v}\|^2\Big)\right)\right] \leq 1.$$

To this end, we define the sequence of random variables $X_1, X_2, \ldots X_k$ by

$$X_j = \sum_{t \in S_{p_j}} \delta_t \langle \boldsymbol{B}\boldsymbol{u}_t, \boldsymbol{v}\rangle - \frac{1}{2}\|\boldsymbol{P}_{p_j} \boldsymbol{b}_{p_j}\|^2 \|\boldsymbol{P}_{p_j} \boldsymbol{v}\|^2,$$

where each X_j corresponds to the jth pivot that was chosen by the algorithm.[2] We show that $\mathrm{E}[\exp(\sum_{j=1}^{k} X_j)] \leq 1$ by proving the chain of inequalities

$$\mathrm{E}\Big[\exp\Big(\sum_{j=1}^{k} X_j\Big)\Big] \leq \mathrm{E}\Big[\exp\Big(\sum_{j=1}^{k-1} X_j\Big)\Big] \leq \cdots \leq \mathrm{E}[\exp(X_1)] \leq \mathrm{E}[\exp(0)] = 1.$$

Consider some $1 \leq \ell \leq k$. Let Δ_ℓ be all random decisions made by the algorithm up until

[2] In the proof sketch in the main paper, we used terms D_i which did not incorporate the projection $\|\boldsymbol{P}_{p_j} \boldsymbol{b}_{p_j}\|^2$, so $X_i \geq D_i$. By incorporating the projection terms in this full proof, we more clearly see the stronger inequality (A6) that is being proven. This highlights that the subgaussian bound will be loose when $\|\boldsymbol{P}_{p_j} \boldsymbol{b}_{p_j}\|^2 \leq 1$ is a loose inequality.

the beginning of pivot phase ℓ. Then observe that

$$\begin{aligned}
\mathrm{E}\Big[\exp\Big(\sum_{j=1}^{\ell} X_j\Big)\Big] &= \mathrm{E}\Big[\exp\Big(\sum_{j=1}^{\ell-1} X_j\Big) \cdot \exp(X_\ell)\Big] && \text{(property of exponential)} \\
&= \mathrm{E}\Big[\exp\Big(\sum_{j=1}^{\ell-1} X_j\Big) \cdot \mathrm{E}[\exp(X_\ell) \mid \Delta_\ell]\Big] && \text{(iterated expectations)} \\
&\leq \mathrm{E}\Big[\exp\Big(\sum_{j=1}^{\ell-1} X_j\Big)\Big], && \text{(by Lemma A8)}
\end{aligned}$$

which completes the induction. □

A2.5 Extending the analysis to the Gram–Schmidt Walk design

In this section, we demonstrate that our analysis of the Gram–Schmidt Walk algorithm extends to the Gram–Schmidt Walk design. The main difference between the Gram–Schmdit Walk algorithm and the Gram–Schmidt Walk design are the construction of input vectors and the randomized pivoting rule. The randomized pivoting rule in the design is inconsequential to the theorems proved in this section. The purpose of the randomized pivoting rule is to allow us to prove that the second-order assignment probabilities are bounded away from zero, which we need for our estimator of the ridge loss discussed in Section A4.

We remark that the Gram–Schmidt Walk design presented in Section 4 may be implemented as follows:

1. Construct the $(n+d)$-dimensional augmented covariate vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ as

$$\boldsymbol{b}_i = \begin{bmatrix} \sqrt{\phi}\boldsymbol{e}_i \\ \xi^{-1}\sqrt{1-\phi}\boldsymbol{x}_i \end{bmatrix},$$

   where $\boldsymbol{e}_i$ is the n-dimensional ith standard basis vector and $\xi = \max_{i\in[n]}\|\boldsymbol{x}_i\|$.

2. Permute the order of the input vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ with a uniformly random permutation.

3. Run the Gram–Schmidt Walk (Algorithm 1) with permuted input vectors and initial fractional assignment $\boldsymbol{z}_1 = \boldsymbol{0}$ to produce assignment vector $\boldsymbol{z}$.

The key idea behind the equivalence of these descriptions is that the method of uniformly permuting input vectors then deterministically choosing largest indexed alive unit as pivot (as presented here) produces the same distribution as choosing pivots uniformly from the set of alive units (as presented in Section 4). To see this equivalence, begin by considering

the first iteration: the largest index in a uniformly permuted list of units is uniform over all units. This means that the first pivot chosen by the two methods has the same distribution. Moreover, the construction of step direction and step size does not depend on the index of the units. In this way, a similar argument shows that these methods of selecting the pivot are equivalent: the largest index in a uniformly permuted list of alive units is uniform over all alive units. Thus, the two random pivot sampling schemes are equivalent.

Due to this equivalence, we may analyze the Gram–Schmidt Walk design by applying the analysis in this section. Because the covariance bound (Theorem 1*) and the subgaussian concentration (Theorem 4*) hold for all orderings of the input vectors, they hold for any distribution over the orderings of the input vectors. In particular, they hold for the uniform distribution over orderings of the input vectors and so they apply to the Gram–Schmidt Walk design.

Finally, we remark that the augmented covariate vectors constructed in the Gram–Schmidt Walk design satisfy the condition that each of their ℓ_2 norms is at most one. This norm condition is a scaling requirement in order to make the covariance and subgaussian bounds in Theorem 1 and Theorem 4, respectively. To see that the norm condition holds, observe that

$$\|\boldsymbol{b}_i\|^2 = \left\|\sqrt{\phi}\boldsymbol{e}_i\right\|^2 + \left\|\xi^{-1}\sqrt{1-\phi}\boldsymbol{x}_i\right\|^2 = \phi + (1-\phi)\left(\xi^{-1}\|\boldsymbol{x}_i\|\right)^2 \leq \phi + (1-\phi) = 1,$$

where the inequality follows from the definition $\xi = \max_{i\in[n]}\|\boldsymbol{x}_i\|$.

Taken together, this shows that Theorems 1 and 4 in the main paper follow from Theorems 1* and 4* in this supplement.

A3 Consequences of the augmented covariates

In this section, we prove several technical lemmas related to the augmented covariate vectors which are referenced in the main body of the paper. These technical lemmas are chiefly concerned with the matrix

$$\boldsymbol{Q} = (\boldsymbol{B}^\intercal\boldsymbol{B})^{-1} = (\phi\boldsymbol{I} + (1-\phi)\boldsymbol{X}\boldsymbol{X}^\intercal)^{-1}.$$

First, we prove an eigenvalue upper bound on the matrix $\boldsymbol{Q}$ which determines the worst-case mean squared error of the Horvitz–Thompson estimator under the Gram–Schmidt Walk design. Then, we demonstrate the relationship between the quadratic form in the matrix $\boldsymbol{Q}$ and the loss of ridge regression. Next, we derive an alternative expression of the matrix bound on the covariate imbalance. Finally, we discuss examples of covariate matrices for which the balance-robustness trade-off cannot be significantly better than that guaranteed by Gram–Schmidt Walk.

A3.1 A bound on worst case MSE (Lemma A9)

Lemma A9. *For all values of the design parameter $\phi \in (0, 1]$, the largest eigenvalue of the matrix $\boldsymbol{Q} = \left(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\right)^{-1}$ is at most $1/\phi$.*

Proof. For all values $\phi \in (0, 1]$, the matrix $(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal)$ is positive definite and thus invertible. Recall that for a positive definite matrix $\boldsymbol{A}$, the largest eigenvalue of $\boldsymbol{A}^{-1}$ is the inverse of the smallest eigenvalue of $\boldsymbol{A}$, i.e. $\lambda_{\max}(\boldsymbol{A}^{-1}) = 1/\lambda_{\min}(\boldsymbol{A})$. Thus, we seek to lower bound the smallest eigenvalue of $\left(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\right)$. To this end, observe that

$$\lambda_{\min}(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal) = \phi + (1-\phi)\xi^{-2}\lambda_{\min}(\boldsymbol{X}\boldsymbol{X}^\intercal) \geq \phi,$$

where the final inequality follows because $\boldsymbol{X}\boldsymbol{X}^\intercal$ is positive semidefinite. □

We remark that when $d < n$, the inequality in Lemma A9 holds with equality. This is because when $d < n$, $\boldsymbol{X}\boldsymbol{X}^\intercal$ is not full rank, and so $\lambda_{\min}(\boldsymbol{X}\boldsymbol{X}^\intercal) = 0$.

A3.2 Connection to ridge regression loss (Lemma A10)

We now present the relationship between the quadratic form in matrix $\boldsymbol{Q}$ and the loss of ridge regression. This lemma is used to establish an upper bound on the mean squared error of the Horvitz-Thompson estimator under the Gram–Schmidt Walk design in Theorem 3.

Lemma A10. *Let $\boldsymbol{X}$ be an arbitrary n-by-d matrix with maximum row norm $\xi = \max_{i\in[n]}\|\boldsymbol{x}_i\|$. For all $\phi \in (0, 1)$ and $\boldsymbol{\mu} \in \mathbb{R}^n$,*

$$nL = \boldsymbol{\mu}^\intercal \boldsymbol{Q}\boldsymbol{\mu} = \boldsymbol{\mu}^\intercal\left(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}^\intercal\boldsymbol{X}\right)^{-1}\boldsymbol{\mu} = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi}\|\boldsymbol{\mu} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \frac{\xi^2}{1-\phi}\|\boldsymbol{\beta}\|^2\right].$$

Proof. Let $\boldsymbol{\beta}^*$ be the optimal linear function in the minimization term above. Note that multiplying the objective function by $\phi > 0$ does not change the minimizer $\boldsymbol{\beta}^*$, and so

$$\boldsymbol{\beta}^* = \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi}\|\boldsymbol{\mu} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \frac{\xi^2}{1-\phi}\|\boldsymbol{\beta}\|^2\right] = \operatorname*{arg\,min}_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\|\boldsymbol{\mu} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \frac{\xi^2\phi}{1-\phi}\|\boldsymbol{\beta}\|^2\right],$$

which has closed-form solution (see, e.g., Hastie et al., 2009, p. 64):

$$\boldsymbol{\beta}^* = \left(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}\right)^{-1}\boldsymbol{X}^\intercal\boldsymbol{\mu} = \boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\boldsymbol{\mu},$$

where we have defined $\boldsymbol{R} = \boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}$. We next consider each of the terms in the

objective function when we substitute the optimal $\boldsymbol{\beta}^*$. The second term becomes

$$\frac{\xi^2}{1-\phi}\|\boldsymbol{\beta}^*\|^2 = \frac{\xi^2}{1-\phi}\left\|\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\boldsymbol{\mu}\right\|^2 = \frac{\xi^2}{1-\phi}\boldsymbol{\mu}^\intercal\boldsymbol{X}\boldsymbol{R}^{-2}\boldsymbol{X}^\intercal\boldsymbol{\mu}.$$

The first term becomes

$$\begin{aligned}
\frac{1}{\phi}\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}^*\|^2 &= \frac{1}{\phi}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\boldsymbol{\mu}\right\|^2 = \frac{1}{\phi}\left\|\left(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\right)\boldsymbol{\mu}\right\|^2 = \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\right)^2\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-2\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal+\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\right)\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\boldsymbol{X}\boldsymbol{R}^{-1}\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-2}\boldsymbol{X}^\intercal\boldsymbol{X}\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu}\ ,
\end{aligned}$$

where the last line follows from the fact that $\boldsymbol{R}^{-1}$ and $\boldsymbol{X}^\intercal\boldsymbol{X}$ commute. To see that the matrices $\boldsymbol{R}^{-1}$ and $\boldsymbol{X}\boldsymbol{X}^\intercal$ commute, first observe that $\boldsymbol{R} = \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}+\boldsymbol{X}^\intercal\boldsymbol{X}$ has the same eigenvectors as $\boldsymbol{X}^\intercal\boldsymbol{X}$. It follows that $\boldsymbol{R}^{-1}$ also has the same eigenvectors as $\boldsymbol{X}^\intercal\boldsymbol{X}$. Thus, the two matrices $\boldsymbol{R}^{-1}$ and $\boldsymbol{X}^\intercal\boldsymbol{X}$ are simultaneously diagonalizable and therefore commute.

Substituting these separate calculations into the objective function, we obtain the optimal value

$$\begin{aligned}
\frac{1}{\phi}\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}^*\|^2+\frac{\xi^2}{1-\phi}\|\boldsymbol{\beta}^*\|^2 &= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-2}\boldsymbol{X}^\intercal\boldsymbol{X}\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu}+\frac{\xi^2}{1-\phi}\boldsymbol{\mu}^\intercal\boldsymbol{X}\boldsymbol{R}^{-2}\boldsymbol{X}^\intercal\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-2}\boldsymbol{X}^\intercal\boldsymbol{X}-\frac{\phi\xi^2}{1-\phi}\boldsymbol{R}^{-2}\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-2}\left(\boldsymbol{X}^\intercal\boldsymbol{X}+\frac{\phi\xi^2}{1-\phi}\boldsymbol{I}\right)\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\left[2\boldsymbol{R}^{-1}-\boldsymbol{R}^{-2}\boldsymbol{R}\right]\boldsymbol{X}^\intercal\right)\boldsymbol{\mu} \\
&= \frac{1}{\phi}\boldsymbol{\mu}^\intercal\left(\boldsymbol{I}-\boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\right)\boldsymbol{\mu}
\end{aligned}$$

To complete the proof, we apply the Woodbury identity which asserts that for appropriately sized matrices $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{C}$, $(\boldsymbol{I}+\boldsymbol{U}\boldsymbol{C}\boldsymbol{V})^{-1} = \boldsymbol{I}-\boldsymbol{U}\left(\boldsymbol{C}^{-1}+\boldsymbol{V}\boldsymbol{U}\right)^{-1}\boldsymbol{V}$, given that the inverses exist. Applying the Woodbury identity with $\boldsymbol{U}=\boldsymbol{X}$, $\boldsymbol{V}=\boldsymbol{X}^\intercal$, and $\boldsymbol{C}=\frac{1-\phi}{\xi^2\phi}\boldsymbol{I}$,

we obtain

$$\frac{1}{\phi}\big(\boldsymbol{I} - \boldsymbol{X}\boldsymbol{R}^{-1}\boldsymbol{X}^\intercal\big) = \frac{1}{\phi}\left(\boldsymbol{I} - \boldsymbol{X}\left(\frac{\xi^2\phi}{1-\phi}\boldsymbol{I} + \boldsymbol{X}^\intercal\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\intercal\right)$$
$$= \frac{1}{\phi}\left(\boldsymbol{I} + \frac{\xi^{-2}(1-\phi)}{\phi}\boldsymbol{X}^\intercal\boldsymbol{X}\right)^{-1} = \big(\phi\boldsymbol{I} + \xi^{-2}(1-\phi)\boldsymbol{X}^\intercal\boldsymbol{X}\big)^{-1}. \qquad \square$$

A3.3 Alternative matrix bound on covariate balance (Proposition 2)

We now present an identity which allows for an alternative form of the matrix upper bound on the covariate balance. This alternative form expresses the upper bound as the weighted harmonic mean of two matrices, which allows for different insights on covariate balance, as discussed in Section 8.

Proposition 2. *Under the Gram–Schmidt Walk design, the covariance matrix of $\boldsymbol{X}^\intercal\boldsymbol{z}$ is bounded in the Loewner order by*

$$\mathrm{Cov}(\boldsymbol{X}^\intercal\boldsymbol{z}) \preceq \Big(\phi(\boldsymbol{X}^\intercal\boldsymbol{X})^\dagger + (1-\phi)(\xi^2\boldsymbol{\Pi})^\dagger\Big)^\dagger,$$

where $\boldsymbol{\Pi}$ is the orthogonal projection onto the rows of the covariate matrix $\boldsymbol{X}$ and $\boldsymbol{A}^\dagger$ denotes the pseudo-inverse of $\boldsymbol{A}$.

Proof. After rearranging terms, Lemma 3 shows that

$$\mathrm{Cov}(\boldsymbol{X}^\intercal\boldsymbol{z}) \preceq \boldsymbol{X}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\big)^{-1}\boldsymbol{X}.$$

To prove the current proposition, we will show that we may re-write this matrix upper bound as

$$\boldsymbol{X}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\big)^{-1}\boldsymbol{X} = \Big(\phi(\boldsymbol{X}^\intercal\boldsymbol{X})^\dagger + (1-\phi)(\xi^2\boldsymbol{\Pi})^\dagger\Big)^\dagger$$

We do so by reasoning about the singular value decomposition of the covariate matrix $\boldsymbol{X}$. To this end, let $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal$ be the singular value decomposition. We only consider the case where $d \leq n$, as the case where $d > n$ follows in a similar manner. If $d \leq n$, then $\boldsymbol{U}$ is a n-by-n orthogonal matrix, $\boldsymbol{\Sigma}$ is an n-by-n diagonal matrix with non-negative diagonal entries, and $\boldsymbol{V}$ is a d-by-n matrix with orthogonal rows. Using the singular value decomposition and orthogonality properties of $\boldsymbol{U}$, we have that

$$\boldsymbol{X}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\big)^{-1}\boldsymbol{X}$$

$$
\begin{aligned}
&= \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\big)^{-1}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal && \text{(SVD)}\\
&= \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\big(\phi\boldsymbol{U}\boldsymbol{U}^\intercal + (1-\phi)\xi^{-2}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\big)^{-1}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal && (\boldsymbol{U}\boldsymbol{U}^\intercal = \boldsymbol{I})\\
&= \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\big(\boldsymbol{U}\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\big)\boldsymbol{U}^\intercal\big)^{-1}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal && \text{(distributing } \boldsymbol{U})\\
&= \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\boldsymbol{U}\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\big)^{-1}\boldsymbol{U}^\intercal\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal && \text{(inverse and } \boldsymbol{U}^{-1} = \boldsymbol{U}^\intercal)\\
&= \boldsymbol{V}\boldsymbol{\Sigma}\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\big)^{-1}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal && (\boldsymbol{U}\boldsymbol{U}^\intercal = \boldsymbol{I})
\end{aligned}
$$

We can compute the pseudo-inverse of this matrix as

$$
\begin{aligned}
\Big(\boldsymbol{X}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\big)^{-1}\boldsymbol{X}\Big)^\dagger &= \Big(\boldsymbol{V}\boldsymbol{\Sigma}\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\big)^{-1}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\Big)^\dagger\\
&= \boldsymbol{V}\boldsymbol{\Sigma}^\dagger\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\big)\boldsymbol{\Sigma}^\dagger\boldsymbol{V}^\intercal\\
&= \phi\boldsymbol{V}(\boldsymbol{\Sigma}^\dagger)^2\boldsymbol{V}^\intercal + (1-\phi)\xi^{-2}\boldsymbol{V}\boldsymbol{\Sigma}^\dagger\boldsymbol{\Sigma}\boldsymbol{V}^\intercal\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\dagger\boldsymbol{V}^\intercal\\
&= \phi\boldsymbol{V}(\boldsymbol{\Sigma}^\dagger)^2\boldsymbol{V}^\intercal + (1-\phi)\xi^{-2}(\boldsymbol{V}\boldsymbol{\Sigma}^\dagger\boldsymbol{\Sigma}\boldsymbol{V}^\intercal)^2,
\end{aligned}
$$

where the third equality follows from distributing the outer matrices. We analyze each term separately, beginning with the left term. Note that

$$
\boldsymbol{X}^\intercal\boldsymbol{X} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^\intercal\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\intercal = \boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^\intercal
$$

and so by the orthogonality of rows of $\boldsymbol{V}$, one can check that

$$
(\boldsymbol{X}^\intercal\boldsymbol{X})^\dagger = \boldsymbol{V}(\boldsymbol{\Sigma}^2)^\dagger\boldsymbol{V}^\intercal = \boldsymbol{V}(\boldsymbol{\Sigma}^\dagger)^2\boldsymbol{V}^\intercal.
$$

The matrix in the second term is equal to the orthogonal projection matrix onto the row span of $\boldsymbol{X}$. To see this, observe that $\boldsymbol{V}\boldsymbol{\Sigma}^\dagger\boldsymbol{\Sigma}\boldsymbol{V}^\intercal$ is the sum of the outer products of the right singular vectors corresponding to positive singular values. Because these vectors form an orthonormal basis for the row span of $\boldsymbol{X}$, the sum of their outer products is the projection matrix $\boldsymbol{\Pi}$. As $\boldsymbol{\Pi}^2 = \boldsymbol{\Pi} = \boldsymbol{\Pi}^\dagger$,

$$
(1-\phi)\xi^{-2}(\boldsymbol{V}\boldsymbol{\Sigma}^\dagger\boldsymbol{\Sigma}\boldsymbol{V}^\intercal)^2 = (1-\phi)\xi^{-2}\boldsymbol{\Pi}^2 = (1-\phi)\xi^{-2}\boldsymbol{\Pi}^\dagger = (1-\phi)(\xi^2\boldsymbol{\Pi})^\dagger.
$$

Putting these two terms together, we arrive at

$$
\Big(\boldsymbol{X}^\intercal\big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\big)^{-1}\boldsymbol{X}\Big)^\dagger = \phi(\boldsymbol{X}^\intercal\boldsymbol{X})^\dagger + (1-\phi)(\xi^2\boldsymbol{\Pi})^\dagger.
$$

The proof is completed by taking the pseudoinverse of both sides. □

A3.4 Examples of the balance-robustness frontier

In this section, we provide a family of covariate matrices for which no design will be able to uniformly perform much better than the Gram–Schmidt Walk design in terms of the balance-robustness tradeoff.

This family of covariate matrices is constructed by creating groups of units where the covariate vectors are identical within groups and orthogonal between groups. There will be d groups, each of an odd size $k \geq 3$, so that there are $n = dk$ units. Let $\boldsymbol{v}_1, \dots \boldsymbol{v}_d$ be d-dimensional orthonormal vectors. In our example, all units in the ℓth group will have the covariate vector $\boldsymbol{v}_\ell$. Thus, the n-by-d covariate matrix $\boldsymbol{X}$ is obtained by stacking k copies of each of these orthonormal vectors.

Our goal is to show that if a design balances these covariates, then it does so necessarily at the expense of robustness. The main idea is that if the spectral norm of $\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})$ is small, then there is a d-dimensional subspace of $\mathbb{R}^n$ where the quadratic form in $\mathrm{Cov}(\boldsymbol{z})$ is small. Together with a trace argument, this implies a lower bound on the spectral norm of $\mathrm{Cov}(\boldsymbol{z})$.

Proposition A1. *Let $\boldsymbol{X}$ be the n-by-d covariate matrix described above. For any design satisfying $\Pr(z_i = 1) = 1/2$ for all units and $\|\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\| \leq c$,*

$$\|\mathrm{Cov}(\boldsymbol{z})\| \geq 1 + \frac{1}{k} - \frac{c-1}{k(k-1)}.$$

Proof. The condition $\|\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\| \leq c$ implies that for any $\boldsymbol{\theta} \in \mathbb{R}^d$,

$$(\boldsymbol{X}\boldsymbol{\theta})^\intercal \mathrm{Cov}(\boldsymbol{z})(\boldsymbol{X}\boldsymbol{\theta}) = \boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \mathrm{Cov}(\boldsymbol{z})\boldsymbol{X}\boldsymbol{\theta} = \boldsymbol{\theta}^\intercal \mathrm{Cov}(\boldsymbol{X}\boldsymbol{z})\boldsymbol{\theta} \leq c\|\boldsymbol{\theta}\|^2. \tag{A7}$$

We are interested in the d vectors $\boldsymbol{w}_i = \frac{1}{\sqrt{k}}\boldsymbol{X}\boldsymbol{e}_i$ for $i = 1, 2, \dots d$, which correspond to $\boldsymbol{\theta}_i = \frac{1}{\sqrt{k}}\boldsymbol{e}_i$. We claim that the vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, \dots \boldsymbol{w}_d$ are orthonormal, which follows from the orthogonality of the covariate vectors. To see this, observe that

$$\langle \boldsymbol{w}_i, \boldsymbol{w}_j \rangle = \left(\frac{1}{\sqrt{k}}\boldsymbol{X}\boldsymbol{e}_i\right)^\intercal \left(\frac{1}{\sqrt{k}}\boldsymbol{X}\boldsymbol{e}_j\right) = \frac{1}{k}\boldsymbol{e}_i^\intercal(\boldsymbol{X}^\intercal\boldsymbol{X})\boldsymbol{e}_j = \frac{1}{k}\boldsymbol{e}_i^\intercal(k \cdot \boldsymbol{I})\boldsymbol{e}_j = \mathbb{1}[i = j].$$

Furthermore, we have that $\|\boldsymbol{\theta}_i\|^2 = \frac{1}{k}$ and so by (A7), we have that

$$\boldsymbol{w}_i^\intercal \mathrm{Cov}(\boldsymbol{z})\boldsymbol{w}_i \leq \frac{c}{k} \quad \text{for } i = 1, 2, \dots d, \tag{A8}$$

Inequality (A8) demonstrates that the d-dimensional subspace $\mathrm{span}\{\boldsymbol{w}_1, \boldsymbol{w}_2, \dots \boldsymbol{w}_d\}$ has a quadratic form in $\mathrm{Cov}(\boldsymbol{z})$ which is at most $\frac{c}{k}$. The Courant–Fischer theorem implies that $\lambda_d \leq \frac{c}{k}$, where the eigenvalues of $\mathrm{Cov}(\boldsymbol{z})$ are ordered as $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. By

assumption, the design satisfies the property $\Pr(z_i = 1) = 1/2$ for all units $i \in [n]$ which means that the diagonal entries of $\mathrm{Cov}(\boldsymbol{z})$ are 1, which in turn implies that the trace is n. Combining these facts yields the inequality

$$n = \mathrm{tr}(\mathrm{Cov}(\boldsymbol{z})) = \sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{d} \lambda_i + \sum_{i=d+1}^{n} \lambda_i \leq \frac{dc}{k} + \sum_{i=d+1}^{n} \lambda_i.$$

Rearranging this equation yields $\sum_{i=d+1}^{n} \lambda_i \geq n - \frac{dc}{k}$. Applying this to an averaging argument yields the desired lower bound on the largest eigenvalue of $\mathrm{Cov}(\boldsymbol{z})$:

$$\lambda_n \geq \frac{1}{n-d} \sum_{i=d+1}^{n} \lambda_i \geq \frac{1}{n-d}\left[n - \frac{dc}{k}\right] = 1 + \frac{1}{k} - \frac{c-1}{k(k-1)}. \qquad \square$$

We remark that Proposition A1 can only be applied for values of $c \geq 1$, because any design satisfying $\Pr(z_i = 1) = 1/2$ for all units must also satisfy $\|\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})\| \geq 1$. To see this, let an assignment vector $\boldsymbol{z} \in \{\pm 1\}^n$ be given and observe that

$$\|\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\|^2 = \Big\|\sum_{i=1}^{n} z_i \boldsymbol{x}_i\Big\|^2 = \Big\|\sum_{j=1}^{d}\Big(\sum_{i:\boldsymbol{x}_i = \boldsymbol{v}_j} z_i\Big)\boldsymbol{v}_j\Big\|^2 = \sum_{j=1}^{d}\Big(\sum_{i:\boldsymbol{x}_i = \boldsymbol{v}_j} z_i\Big)^2 \|\boldsymbol{v}_j\|^2 \geq d,$$

where the last equality followed from orthogonality of the vectors $\boldsymbol{v}_1, \ldots \boldsymbol{v}_d$ and the inequality followed from $\|\boldsymbol{v}_i\| = 1$ and also that the sum of the ± 1 assignment vector coordinates within a group must be odd because k is odd. The condition $\Pr(z_i = 1) = 1/2$ implies that the average assignment vector is zero. Using this and the linearity and cyclic properties of the trace, we have

$$\mathrm{tr}(\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})) = \mathrm{tr}(\mathrm{E}[\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\boldsymbol{z}^\mathsf{T}\boldsymbol{X}]) = \mathrm{E}[\mathrm{tr}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\boldsymbol{z}^\mathsf{T}\boldsymbol{X})] = \mathrm{E}[\mathrm{tr}(\boldsymbol{z}^\mathsf{T}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}\boldsymbol{z})] = \mathrm{E}[\|\boldsymbol{X}^\mathsf{T}\boldsymbol{z}\|^2] \geq d$$

Thus, a trace argument shows that the largest eigenvalue of $\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$ is at least one. Ordering the eigenvalues of $\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})$ as $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_d$, we have

$$\|\mathrm{Cov}(\boldsymbol{X}^\mathsf{T}\boldsymbol{z})\| = \lambda_d \geq \frac{1}{d}\sum_{i=1}^{d} \lambda_i \geq \frac{1}{d} \cdot d = 1.$$

Finally, we remark on how to rearrange the terms in Proposition A1 to obtain the trade-off inequality stated in Section 9. One can verify that $\boldsymbol{X}^\mathsf{T}\boldsymbol{X} = k \cdot \boldsymbol{I}$ and so its largest eigenvalue is k. Likewise, the maximum row norm is 1. Thus, for any value of $\phi \in (0, 1)$,

$$\left(\phi\lambda_G^{-1} + (1-\phi)\xi^{-2}\right)^{-1} = (\phi/k + (1-\phi))^{-1}$$

Setting $c = (\phi/k + (1-\phi))^{-1}$ in Proposition A1 yields the lower bound

$$\|\mathrm{Cov}(\boldsymbol{z})\| \geq 1 + \frac{1}{k} - \frac{c-1}{k(k-1)} = \frac{1+k(1-\phi)}{\phi+k(1-\phi)}.$$

A4 Estimating the ridge regression loss

In this section, we discuss estimators for the ridge loss quantity, which appears in our variance upper bound and our confidence intervals. We begin by deriving and analyzing the Horvitz–Thompson estimator for the ridge loss proposed in Section 11.3 in the setting where individual treatment probabilities are uniform across treatments for all units, i.e., $\Pr(z_i = 1) = 1/2$ for all $i \in [n]$. Our analysis shows that this estimator is conservative in expectation. We then prove two different lower bounds on the second-order assignment probabilities which are used in the proof of conservativeness. Next, we derive the more general form of the Horvitz–Thompson estimator for the ridge loss when the individual treatment probabilities are arbitrary. Finally, we discuss an alternative estimator for the ridge loss which does not require estimation of the second-order assignment probabilities.

A4.1 Derivation of the ridge loss estimator (Proposition 4)

In this section, we derive the form for the Horvitz–Thompson estimator for the ridge loss discussed in Section 11.3 and we prove that it is conservative in expectation. We focus here on the setting where first-order assignment probabilities are uniform across treatments for all units: $\Pr(z_i = 1) = 1/2$ for all units $i \in [n]$. In this case, recall that the ridge loss is defined as

$$L = \frac{1}{n}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{Q}\boldsymbol{\mu} \qquad \text{where} \qquad \boldsymbol{Q} = \left(\boldsymbol{B}^\mathsf{T}\boldsymbol{B}\right)^{-1} = \left(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}\right)^{-1} \text{ and } \boldsymbol{\mu} = \frac{\boldsymbol{a}+\boldsymbol{b}}{2}.$$

The ridge loss defined above contains terms like $a_i b_i$, which are never observed. In Section 11.3, we derived an estimable upper bound on the ridge loss which only depended on observable quantities:

$$4nL \leq \boldsymbol{a}^\mathsf{T}\big[\boldsymbol{Q} + \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{a} + \boldsymbol{b}^\mathsf{T}\big[\boldsymbol{Q} + \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{b} + 2\boldsymbol{a}^\mathsf{T}\big[\boldsymbol{Q} - \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{b}.$$

Note that a factor of 4 appears in the left hand side for clarity of the right hand side. The Horvitz–Thompson estimator we derive will estimate this upper bound. Using q_{ij} to denote the entry in the ith row and jth column of $\boldsymbol{Q}$, we may write the individual terms in this

upper bound as

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(1+\mathbb{1}[i=j])a_i a_j q_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n}(1+\mathbb{1}[i=j])b_i b_j q_{ij} + 2\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbb{1}[i\neq j]a_i b_j q_{ij}.$$

A Horvitz–Thompson estimator $\widehat{L}$ of this upper bound is written as

$$\begin{aligned}
4n\widehat{L} &= \sum_{i\in Z^+}\sum_{j\in Z^+}(1+\mathbb{1}[i=j])\frac{a_i a_j q_{ij}}{\Pr(z_i=1,z_j=1)} \\
&\qquad + \sum_{i\in Z^-}\sum_{j\in Z^-}(1+\mathbb{1}[i=j])\frac{b_i b_j q_{ij}}{\Pr(z_i=-1,z_j=-1)} \\
&\qquad + 2\sum_{i\in Z^+}\sum_{j\in Z^-}\frac{a_i b_j q_{ij}}{\Pr(z_i=1,z_j=-1)} \\
&= \sum_{i=1}^{n}\left[\frac{2y_i^2}{\Pr(z_i)}q_{ii} + \sum_{j\neq i}\frac{y_i y_j}{\Pr(z_i,z_j)}q_{ij}\right].
\end{aligned}$$

The last expression is written in terms of the observed outcomes $y_1,\dots y_n$, and we use $\Pr(z_i)$ and $\Pr(z_i,z_j)$ to denote the probability of the observed individual and pair of assignments. That is, $\Pr(z_i)$ is the probability that unit i is assigned to treatment z_i, and $\Pr(z_i,z_j)$ is the probability that units i and j simultaneously are assigned to treatments z_i and z_j, respectively. Bringing the factor of $4n$ to the right hand side and writing it in matrix notation, we have that $\widehat{L} = \boldsymbol{y}^\intercal\widehat{\boldsymbol{Q}}\boldsymbol{y}/n$, where $\widehat{\boldsymbol{Q}}$ is a random matrix depending on $\boldsymbol{z}$ whose entries are

$$\widehat{q}_{ij} = \begin{cases} \frac{q_{ii}}{2\Pr(z_i)} & \text{if } i=j, \\ \frac{q_{ij}}{4\Pr(z_i,z_j)} & \text{otherwise.} \end{cases}$$

Recall that we are considering the case where all units are assigned to either treatment with equal probabilities: $\Pr(z_i=1)=1/2$ for all $i\in[n]$. In this case, the entries of the matrix are

$$\widehat{q}_{ij} = \begin{cases} q_{ii} & \text{if } i=j, \\ \frac{q_{ij}}{1+z_i z_j\,\mathrm{E}[z_i z_j]} & \text{otherwise,} \end{cases}$$

because $4\Pr(z_i=v_i,z_i=v_j) = 1+v_i v_j\,\mathrm{E}[z_i z_j]$ as shown by the following lemma.

Lemma A11. *For any two random variables z_i and z_j with support $\{\pm1\}$, and for any realizations $v_i, v_j\in\{\pm1\}$,*

$$\Pr(z_i=v_1,z_j=v_2) = \frac{1}{4}\big(1+v_1\,\mathrm{E}[z_i]+v_2\,\mathrm{E}[z_j]+v_1 v_2\,\mathrm{E}[z_i z_j]\big).$$

Proof. Because the variables z_i and z_j only take values ± 1, the indicator function may be expressed as

$$\mathbb{1}[z_i = v_1, z_j = v_2] = \frac{1}{4}(z_i v_1 + 1)(z_j v_2 + 1) = \frac{1}{4}(1 + v_1 z_i + v_2 z_j + v_1 v_2 z_i z_j).$$

The lemma follows when we take the expectation of this expression:

$$\Pr(z_i = v_1, z_j = v_2) = \mathrm{E}\big[\mathbb{1}[z_i = v_1, z_j = v_2]\big] = \frac{1}{4}\big(1 + v_1 \,\mathrm{E}[z_i] + v_2 \,\mathrm{E}[z_j] + v_1 v_2 \,\mathrm{E}[z_i z_j]\big). \quad \square$$

We now prove that the Horvitz-Thompson estimator for the ridge loss is conservative in expectation when the individual treatment probabilities are uniform across treatments for all units. The proof uses Lemma 4, which shows that all second order treatment probabilities are nonzero; however, the proof of this fact contains many technical details so we defer it to the next section.

Proposition 4. *The ridge loss estimator is conservative in expectation:* $\mathrm{E}[\widehat{L}] \geq L$.

Proof. Recall that the Horvtiz-Thompson estimator of the ridge loss is constructed as

$$\widehat{L} = \frac{1}{4n}\Bigg[\sum_{i \in Z^+}\sum_{j \in Z^+}(1 + \mathbb{1}[i = j])\frac{a_i a_j q_{ij}}{\Pr(z_i = 1, z_j = 1)} + \sum_{i \in Z^-}\sum_{j \in Z^-}(1 + \mathbb{1}[i = j])\frac{b_i b_j q_{ij}}{\Pr(z_i = -1, z_j = -1)} + 2\sum_{i \in Z^+}\sum_{j \in Z^-}\frac{a_i b_j q_{ij}}{\Pr(z_i = 1, z_j = -1)}\Bigg].$$

By Lemma 4, all second order assignment probabilities are nonzero. Not only does this imply that the terms in the estimator are well defined, it also implies that the expectation of the estimator is equal to the bound of L. In particular, consider the term

$$\frac{a_i a_j q_{ij}}{\Pr(z_i = 1, z_j = 1)}$$

which appears in the sum when $z_i = z_j = 1$. This happens with probability $\Pr(z_i = 1, z_j = 1) > 0$. Thus, in expectation, this term contributes $a_i a_j q_{ij}$ to the sum. A similar argument shows that the expectation of the entire estimator is equal to the bound of L. Thus,

$$\mathrm{E}[\widehat{L}] = \frac{1}{4n}\Big[\boldsymbol{a}^\intercal\big[\boldsymbol{Q} + \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{a} + \boldsymbol{b}^\intercal\big[\boldsymbol{Q} + \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{b} + 2\boldsymbol{a}^\intercal\big[\boldsymbol{Q} - \mathrm{diag}(\boldsymbol{Q})\big]\boldsymbol{b}\Big] \geq L. \qquad \square$$

A4.2 Second-order assignment probabilities (Lemma 4)

In order to analyze the expectation of the Horvitz–Thompson estimator for the ridge regression loss, we require that all second order assignment probabilities are nonzero. In this section, we prove that this is indeed a property of the Gram–Schmidt Walk design. At the end of the section, we demonstrate a different bound on the second order assignment probabilities which does not depend on the sample size.

In order to show that the second order assignment probabilities are nonzero, we analyze the fractional assignments at the end of the first iteration. The main point of our argument is that for any pair of units $i, j \in [n]$ and assignments $v_i, v_j \in \{\pm 1\}$, there exists a choice of first pivot p_1 and first step size δ_1 so that, conditioned upon this choice, the probability of setting $z_i = v_i$ and $z_i = v_j$ in later iterations is nonzero. Our proof technique requires that each unit has equal probability of being assigned either treatment, i.e., $\Pr(z_i = 1) = 1/2$ for all $i \in [n]$. Recall that this occurs by setting the initial fractional assignment vector as $\boldsymbol{z}_1 = \mathbf{0}$.

We begin by presenting a basic lemma which bounds the joint probability of two binary random variables in terms of their marginal probabilities.

Lemma A12. *For any discrete random variables X and Y,*

$$\Pr(X = x, Y = y) \geq \Pr(X = x) - \Pr(Y \neq y).$$

Proof. Observe that by probability axioms,

$$\Pr(X = x, Y = y) = \Pr(X = x) - \Pr(X = x, Y \neq y) \geq \Pr(X = x) - \Pr(Y \neq y). \qquad \square$$

Next, we derive a unit's marginal probability of assignment conditional on the outcome of the first iteration.

Lemma A13. *The conditional probability that unit i is assigned to treatment $v_i \in \{\pm 1\}$ given the random decisions of the algorithm in the first iteration is*

$$\Pr(z_i = v_i \mid p_1, \delta_1) = \frac{1}{2}\Big(1 + v_i \boldsymbol{z}_2(i)\Big),$$

where we recall that $\boldsymbol{z}_2$ depends on p_1 and δ_1.

Proof. For any ± 1 random variable X and realization $v \in \{\pm 1\}$, we have that $\Pr(X = v) = \frac{1}{2}(1 - v \operatorname{E}[X])$. Using this expression and the martingale property of the fractional assignments (Lemma 2), we have that

$$\Pr(z_i = v_i \mid p_1, \delta_1) = \frac{1}{2}\Big(1 + v_i \operatorname{E}[\boldsymbol{z}_T(i) \mid p_1, \delta_1]\Big) = \frac{1}{2}\Big(1 + v_i \boldsymbol{z}_2(i)\Big). \qquad \square$$

To reason about the fractional assignment $\boldsymbol{z}_2$, we have to reason about the step direction vector $\boldsymbol{u}_1$. We now demonstrate how to derive a matrix which contains all possible realizations of $\boldsymbol{u}_1$ as its columns, up to scaling.

The step direction $\boldsymbol{u}_1$ is completely determined by the choice of pivot p_1. Because we are only considering the first iteration, we drop the subscript 1 for now and, instead, write $\boldsymbol{u}_p$ to denote the step direction when the unit p is chosen as the first pivot. We claim that the step direction is given by

$$\boldsymbol{u}_p = \frac{\boldsymbol{Q}(:,p)}{\boldsymbol{Q}(p,p)} \quad \text{where} \quad \boldsymbol{Q} = \left(\boldsymbol{B}^\intercal \boldsymbol{B}\right)^{-1} = \left(\phi \boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\right)^{-1}$$

and $\boldsymbol{Q}(:,i)$ denotes the ith column of $\boldsymbol{Q}$ and $\boldsymbol{Q}(i,j)$ denotes the entry in the ith row and jth column of $\boldsymbol{Q}$. To see this, recall that the first step direction is obtained by setting the pivot coordinate $\boldsymbol{u}_p(p) = 1$ and choosing the remaining coordinates as minimizers of the least squares problem

$$\boldsymbol{u}_p([n] \setminus p) = \operatorname*{arg\,min}_{u_i : i \neq p} \left\| \boldsymbol{b}_p + \sum_{i \neq p} u_i \boldsymbol{b}_i \right\|^2.$$

When the vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ are linearly independent, the solution is unique and the matrix $(\boldsymbol{B}^\intercal \boldsymbol{B})^{-1}$ exists. Recall that the augmented covariate vectors used in the Gram–Schmidt Walk design are linearly independent by construction for design parameters $\phi > 0$. By first-order optimality conditions, the entire vector $\boldsymbol{u}_p$ should satisfy the property that the vector

$$\boldsymbol{B}\boldsymbol{u}_p = \boldsymbol{b}_p + \sum_{i \neq p} u_i \boldsymbol{b}_i$$

is orthogonal to all $\boldsymbol{b}_i$ with $i \neq p$. That is,

$$0 = \left\langle \boldsymbol{b}_i, \boldsymbol{B}\boldsymbol{u}_p \right\rangle = \left\langle \boldsymbol{B}\boldsymbol{e}_i, \boldsymbol{B}\boldsymbol{u}_p \right\rangle = \left\langle \boldsymbol{B}^\intercal \boldsymbol{B}\boldsymbol{e}_i, \boldsymbol{u}_p \right\rangle \qquad \text{for all } i \neq p.$$

The columns of $\boldsymbol{Q} = (\boldsymbol{B}^\intercal \boldsymbol{B})^{-1}$ satisfy this orthogonality property, as

$$\langle \boldsymbol{B}^\intercal \boldsymbol{B}\boldsymbol{e}_i, (\boldsymbol{B}^\intercal \boldsymbol{B})^{-1}\boldsymbol{e}_p \rangle = \boldsymbol{e}_i^\intercal \boldsymbol{B}^\intercal \boldsymbol{B}(\boldsymbol{B}^\intercal \boldsymbol{B})^{-1}\boldsymbol{e}_p = \boldsymbol{e}_i^\intercal \boldsymbol{I}\boldsymbol{e}_p = \mathbb{1}[i = p].$$

Thus, by dividing the pth column $\boldsymbol{Q}(:,p)$ by the $\boldsymbol{Q}(p,p)$ diagonal entry, the pth coordinate becomes one and we obtain the direction $\boldsymbol{u}_p$.

In order to understand the step direction in the first iteration, we will prove properties of the matrix $\boldsymbol{Q}$. Before doing so, we introduce the following technical lemma.

Lemma A14. *Let $\boldsymbol{A}$ be an n-by-n positive semidefinite matrix with diagonal entries at*

most 1. *For any* $\gamma > 0$, *the matrix* $\boldsymbol{M} = (\boldsymbol{A} + \gamma \boldsymbol{I})^{-1}$ *satisfies*

$$\boldsymbol{M}(i,j)^2 \leq (1+\gamma)^{-2} \boldsymbol{M}(i,i)\boldsymbol{M}(j,j) \qquad \text{for all } i \neq j \in [n].$$

Proof. Let $S = \{i, j\}$ be a pair of indices and define $R = [n] \setminus S$ to be the remaining indices. We are interested in the principal submatrix $\boldsymbol{M}(S, S)$. By using the expression for the inverse of a block matrix, we may express this principal submatrix as

$$\begin{aligned}
\boldsymbol{M}(S,S) &= \Big(\boldsymbol{A} + \gamma \boldsymbol{I}\Big)^{-1}(S,S) && \text{(definition of } \boldsymbol{M}) \\
&= \Big(\boldsymbol{A}(S,S) + \gamma \boldsymbol{I}_S - \boldsymbol{A}(S,R)\Big(\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R\Big)^{-1}\boldsymbol{A}(R,S)\Big)^{-1} && \text{(block matrix inverse)} \\
&= \Big(\boldsymbol{A}(S,S) - \boldsymbol{A}(S,R)\Big(\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R\Big)^{-1}\boldsymbol{A}(R,S) + \gamma \boldsymbol{I}_S\Big)^{-1} && \text{(rearranging terms)} \\
&= \Big(\boldsymbol{B}_S + \gamma \boldsymbol{I}_S\Big)^{-1}, && \text{(defining } \boldsymbol{B}_S)
\end{aligned}$$

where the matrices $\boldsymbol{I}_S$ and $\boldsymbol{I}_R$ are identity matrices of the appropriate sizes.

We claim that $\boldsymbol{B}_S$ is positive semidefinite with diagonal entries at most one. The positive semidefinite property follows because $\boldsymbol{B}_S$ is the Schur complement of $\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R$ onto the block S. The matrix $\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R$ is positive semidefinite so that the matrix $\boldsymbol{A}(S,R)(\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R)^{-1}\boldsymbol{A}(R,S)$ is positive semidefinite and thus has non-negative diagonals. The diagonal entries of $\boldsymbol{A}(S,S)$ are at most one by assumption and because the diagonal entries of $\boldsymbol{A}(S,R)(\boldsymbol{A}(R,R) + \gamma \boldsymbol{I}_R)^{-1}\boldsymbol{A}(R,S)$ are non-negative, the diagonal entries of $\boldsymbol{B}_S$ are at most one.

Thus, the 2-by-2 matrix $\boldsymbol{M}(S,S)^{-1}$ may be expressed as

$$\boldsymbol{M}(S,S)^{-1} = \boldsymbol{B}_S + \gamma \boldsymbol{I} = \begin{pmatrix} \alpha & \eta \\ \eta & \beta \end{pmatrix} + \gamma \boldsymbol{I} = \begin{pmatrix} \alpha + \gamma & \eta \\ \eta & \beta + \gamma \end{pmatrix},$$

where the inequalities $\eta^2 \leq \alpha\beta$ and $\alpha, \beta \leq 1$ follow because $\boldsymbol{B}$ is positive semidefinite with diagonals at most 1. For $\gamma > 0$ this matrix is invertible, so

$$\boldsymbol{M}(S,S) = \frac{1}{\det \boldsymbol{M}(S,S)^{-1}} \begin{pmatrix} \beta + \gamma & -\eta \\ -\eta & \alpha + \gamma \end{pmatrix}.$$

If $\eta = 0$ then $\boldsymbol{M}(i,j) = 0$ so the desired inequality holds. Otherwise, $\eta^2 > 0$ and using

the properties of $\boldsymbol{B}_S$, we have that

$$\frac{\boldsymbol{M}(i,i)\boldsymbol{M}(j,j)}{\boldsymbol{M}(i,j)^2} = \frac{(\beta+\gamma)(\alpha+\gamma)}{\eta^2} \geq \frac{(\beta+\gamma)(\alpha+\gamma)}{\alpha\beta} = \Big(1+\frac{\gamma}{\beta}\Big)\Big(1+\frac{\gamma}{\alpha}\Big) \geq (1+\gamma)^2.$$

Rearranging terms yields the desired inequality. □

We now derive properties of the matrix $\boldsymbol{Q}$ which allow us to further reason about the step direction in the first iteration.

Lemma A15. *The n-by-n matrix $\boldsymbol{Q} = (\boldsymbol{B}^\intercal\boldsymbol{B})^{-1} = (\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal)^{-1}$ satisfies the following properties for all pairs of units $i \neq j \in [n]$:*

1. *Diagonal entries are lower bounded by $\boldsymbol{Q}(i,i) \geq 1$.*
2. *Off-diagonal entry upper bounded by $|\boldsymbol{Q}(i,j)| \leq \frac{1-\phi}{\phi}$.*
3. *All 2-by-2 principal submatrices admit the bound $\boldsymbol{Q}(i,j)^2 \leq (1-\phi)^2\boldsymbol{Q}(i,i)\boldsymbol{Q}(j,j)$.*

Proof. To begin proving the statements of the theorem, we derive the entries of the matrix $\boldsymbol{Q}$. By rearranging terms and using the Woodbury identity,

$$\begin{aligned}
\boldsymbol{Q} &= \Big(\phi\boldsymbol{I}_n + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\intercal\Big)^{-1} \\
&= \phi^{-1}\left[\boldsymbol{I}_n + \frac{(1-\phi)}{\phi\xi^2}\boldsymbol{X}\boldsymbol{X}^\intercal\right]^{-1} && \text{(rearranging terms)} \\
&= \phi^{-1}\left[\boldsymbol{I}_n - \frac{1-\phi}{\phi\xi^2}\boldsymbol{X}\Big(\boldsymbol{I}_d + \frac{1-\phi}{\phi\xi^2}\boldsymbol{X}^\intercal\boldsymbol{X}\Big)^{-1}\boldsymbol{X}^\intercal\right] && \text{(Woodbury identity)} \\
&= \phi^{-1}\left[\boldsymbol{I}_n - \boldsymbol{X}\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{X}^\intercal\right]. && \text{(rearranging terms)}
\end{aligned}$$

So the entries of the matrix $\boldsymbol{Q}$ may be computed directly as

$$\boldsymbol{Q}(i,j) = \boldsymbol{e}_i^\intercal\boldsymbol{Q}\boldsymbol{e}_j = \phi^{-1}\left(\mathbb{1}[i=j] - \boldsymbol{x}_i^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_j\right).$$

We will now bound a relevant quadratic form. Note that for any unit i, we have the following matrix bound: $\boldsymbol{X}^\intercal\boldsymbol{X} = \sum_{j=1}^n \boldsymbol{x}_j\boldsymbol{x}_j^\intercal \succeq \boldsymbol{x}_i\boldsymbol{x}_i^\intercal$. This implies the matrix inequality

$$\left(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\right)^{-1} \preceq \left(\boldsymbol{x}_i\boldsymbol{x}_i^\intercal + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\right)^{-1} \qquad \text{for all } i \in [n].$$

Set $\alpha = \phi\xi^2/(1-\phi)$. Using the matrix bound above and the Sherman–Morrison formula, we may bound the quadratic form as

$$\begin{aligned}
\boldsymbol{x}_i^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \alpha\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_i &\leq \boldsymbol{x}_i^\intercal\Big(\boldsymbol{x}_i\boldsymbol{x}_i^\intercal + \alpha\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_i && \text{(matrix bound above)}\\
&= \boldsymbol{x}_i^\intercal\left(\alpha^{-1}\boldsymbol{I}_d - \frac{\alpha^{-2}\boldsymbol{x}_i\boldsymbol{x}_i^\intercal}{1+\alpha^{-1}\|\boldsymbol{x}_i\|^2}\right)\boldsymbol{x}_i && \text{(Sherman–Morrison)}\\
&= \left(\alpha^{-1}\|\boldsymbol{x}_i\|^2 - \frac{\alpha^{-2}\|\boldsymbol{x}_i\|^4}{1+\alpha^{-1}\|\boldsymbol{x}_i\|^2}\right) && \text{(distributing terms)}\\
&= \frac{\|\boldsymbol{x}_i\|^2}{\alpha + \|\boldsymbol{x}_i\|^2} && \text{(rearranging terms)}\\
&= \frac{\|\boldsymbol{x}_i\|^2}{\frac{\phi\xi^2}{1-\phi} + \|\boldsymbol{x}_i\|^2} = \frac{\|\boldsymbol{x}_i\|^2/\xi^2}{\frac{\phi}{1-\phi} + \|\boldsymbol{x}_i\|^2/\xi^2} && \text{(substituting } \alpha)\\
&\leq \frac{1}{\frac{\phi}{1-\phi}+1} = 1-\phi,
\end{aligned}$$

where the second inequality follows from the facts that $\|\boldsymbol{x}_i\| \leq \max_{k\in[n]}\|\boldsymbol{x}_k\| = \xi$ and that for all $a > 0$, the function $f_a(y) = \frac{y^2}{a+y^2}$ is increasing for $y \geq 0$.

We now demonstrate the lower bound on diagonal entries of the matrix $\boldsymbol{Q}$. Using the closed form expression for the entries derived above and the bound on the quadratic form, we have

$$\boldsymbol{Q}(i,i) = \phi^{-1}\left(1 - \boldsymbol{x}_i^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_i\right) \geq \phi^{-1}(1-(1-\phi)) = \phi^{-1}\phi = 1.$$

Next, we demonstrate the upper bound on the magnitude of the off-diagonal entries. Using the closed form expression for these entries derived above, the Cauchy-Schwartz inequality, and the above bound on the quadratic form, we have

$$\begin{aligned}
\boldsymbol{Q}(i,j)^2 &= \phi^{-2}\Big(\boldsymbol{x}_i^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_j\Big)^2\\
&= \phi^{-2}\Big\langle\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1/2}\boldsymbol{x}_i, \Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1/2}\boldsymbol{x}_j\Big\rangle^2\\
&\leq \phi^{-2}\Big\|\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1/2}\boldsymbol{x}_i\Big\|^2\Big\|\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1/2}\boldsymbol{x}_j\Big\|^2 && \text{(Cauchy-Schwartz)}\\
&= \phi^{-2}\left(\boldsymbol{x}_i^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_i\right)\left(\boldsymbol{x}_j^\intercal\Big(\boldsymbol{X}^\intercal\boldsymbol{X} + \frac{\phi\xi^2}{1-\phi}\boldsymbol{I}_d\Big)^{-1}\boldsymbol{x}_j\right)
\end{aligned}$$

$$\leq \phi^{-2}(1-\phi)^2 = \Big(\frac{1-\phi}{\phi}\Big)^2, \qquad \text{(bound above)}$$

which establishes the upper bound on the off diagonal entries, $|\boldsymbol{Q}(i,j)| \leq (1-\phi)/\phi$.

Finally, we demonstrate the bound on 2-by-2 principal submatrices. Define

$$\boldsymbol{M} = \Big(\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T} + \frac{\phi}{1-\phi}\boldsymbol{I}\Big)^{-1}.$$

By rearranging terms, we have

$$\boldsymbol{Q} = \Big(\phi\boldsymbol{I} + (1-\phi)\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}\Big)^{-1} = (1-\phi)^{-1}\Big(\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T} + \frac{\phi}{1-\phi}\boldsymbol{I}\Big)^{-1} = (1-\phi)^{-1}\boldsymbol{M}.$$

As $\xi = \max_{i\in[n]}\|\boldsymbol{x}_i\|$ and the diagonal entries of $\boldsymbol{X}\boldsymbol{X}^\mathsf{T}$ are $\|\boldsymbol{x}_i\|^2$, the matrix $\xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}$ is positive semidefinite with diagonal entries at most 1. Note that the entries of $\boldsymbol{Q}$ are the same as the entries of $\boldsymbol{M}$, up to a common factor. Thus, we may apply Lemma A14 with $\boldsymbol{A} = \xi^{-2}\boldsymbol{X}\boldsymbol{X}^\mathsf{T}$ and $\gamma = \frac{\phi}{1-\phi}$ to obtain the third inequality in the statement of the proposition:

$$\boldsymbol{Q}(i,j)^2 \leq \Big(1 + \frac{\phi}{1-\phi}\Big)^{-2}\boldsymbol{Q}(i,i)\boldsymbol{Q}(j,j) = (1-\phi)^2\boldsymbol{Q}(i,i)\boldsymbol{Q}(j,j). \qquad \square$$

We now have the tools to prove the proposition of interest, namely that all pairwise second order assignment probabilities are nonzero.

Lemma 4. *The second-order assignment probabilities are bounded away from zero under the Gram–Schmidt Walk design for all pairs of units and all treatments:*

$$\Pr\big((z_i, z_j) = \boldsymbol{v}\big) > \frac{1}{4n}\min\Big\{\phi, \frac{\phi^2}{1-\phi}\Big\} \quad \textit{for all} \quad i \neq j \quad \textit{and all} \quad \boldsymbol{v} \in \{\pm 1\}^2.$$

Proof. Let $i, j \in [n]$ be two arbitrary but distinct units such that $\boldsymbol{Q}(i,i) \geq \boldsymbol{Q}(j,j)$, which is without loss of generality because of symmetry. We begin by lower bounding the second-order assignment probability conditioned on the random decisions made in the first iteration, namely the first pivot p_1 and the step size δ_1:

$$\begin{aligned}
&\Pr(z_i = v_i, z_j = v_j \mid p_1, \delta_1) \\
&\quad \geq \Pr(z_i = v_i \mid p_1, \delta_1) - \Pr(z_j \neq v_j \mid p_1, \delta_1) && \text{(Lemma A12)} \\
&\quad = \frac{1}{2}\Big(1 + v_i\,\mathrm{E}[\boldsymbol{z}_2(i) \mid p_1, \delta_1]\Big) - \frac{1}{2}\Big(1 - v_i\,\mathrm{E}[\boldsymbol{z}_2(i) \mid p_1, \delta_1]\Big) && \text{(Lemma A13)}
\end{aligned}$$

$$= \frac{1}{2}\Big(v_i \,\mathrm{E}[\boldsymbol{z}_2(i) \mid p_1, \delta_1] + v_j \,\mathrm{E}[\boldsymbol{z}_2(j) \mid p_1, \delta_1]\Big) \qquad \text{(rearranging terms)}$$

$$= \frac{1}{2}\Big(v_i \delta_1 \boldsymbol{u}_1(i) + v_j \delta_1 \boldsymbol{u}_1(j)\Big) \qquad \text{(update rules, } \boldsymbol{z}_1 = \mathbf{0}\text{)}$$

$$= \frac{1}{2}\delta_1\Big(v_i \boldsymbol{u}_1(i) + v_j \boldsymbol{u}_1(j)\Big). \qquad \text{(rearranging terms)}$$

We continue by conditioning on the event that the first pivot is unit i, so that $p_1 = i$. Once the pivot is determined, the first step direction $\boldsymbol{u}_1$ has been determined. We claim that when i is chosen as the pivot, the step direction $\boldsymbol{u}_1$ satisfies the following properties:

1. $\boldsymbol{u}_1(i) = 1$
2. $\max_{k \in [n]} |\boldsymbol{u}_1(k)| \leq \max\{1, \frac{1-\phi}{\phi}\}$
3. $|\boldsymbol{u}_1(j)| \leq 1 - \phi$

The first property follows directly from $p_1 = i$. The second property follows by considering two types of coordinates of $\boldsymbol{u}_1$. As we already noted, the pivot coordinate is $\boldsymbol{u}_1(i) = 1$. We bound the magnitude of non-pivot coordinates $k \neq i$ by combining statements (1) and (2) of Lemma A15,

$$\big|\boldsymbol{u}_1(k)\big| = \big|\langle \boldsymbol{u}_1, \boldsymbol{e}_k \rangle\big| = \left|\left\langle \frac{\boldsymbol{Q}(:,i)}{\boldsymbol{Q}(i,i)}, \boldsymbol{e}_k \right\rangle\right| = \left|\frac{\boldsymbol{Q}(k,i)}{\boldsymbol{Q}(i,i)}\right| = \frac{|\boldsymbol{Q}(k,i)|}{\boldsymbol{Q}(i,i)} \leq |\boldsymbol{Q}(k,i)| \leq \frac{1-\phi}{\phi}.$$

Combining these two yields that $|\boldsymbol{u}_1(k)| \leq \max\{1, \frac{1-\phi}{\phi}\}$ for all $k \in [n]$. The third property follows by the assumption that $\boldsymbol{Q}(i,i) \geq \boldsymbol{Q}(j,j)$ and the third part of Lemma A15. Namely, that

$$\boldsymbol{u}_1(j)^2 = \frac{\boldsymbol{Q}(i,j)^2}{\boldsymbol{Q}(i,i)^2} \leq \frac{\boldsymbol{Q}(i,j)^2}{\boldsymbol{Q}(i,i)\boldsymbol{Q}(j,j)} \leq (1-\phi)^2,$$

which demonstrates that $|\boldsymbol{u}_1(j)| \leq 1 - \phi$, as desired.

Because the initial fractional assignment is $\boldsymbol{z}_1(i) = \mathbf{0}$, the first step size δ_1 is randomly chosen as

$$\delta_1 = \begin{cases} \delta_1^+ = \Big(\max_{k \in [n]} |\boldsymbol{u}_1(k)|\Big)^{-1} & \text{with probability } 1/2 \\ \delta_1^- = \Big(\max_{k \in [n]} |\boldsymbol{u}_1(k)|\Big)^{-1} & \text{with probability } 1/2 \end{cases}$$

Suppose that we further condition on the choice of step size so that $\delta_1 v_i \geq 0$. We refer to this choice of step size as $\delta_1^{v_i}$. Conditioning on this choice of step size and using the properties of the step direction $\boldsymbol{u}_1$ yields

$$2\Pr(z_i = v_i, z_j = v_j \mid p_1, \delta_1) = \delta_1\Big(v_i \boldsymbol{u}_1(i) + v_j \boldsymbol{u}_1(j)\Big) \qquad \text{(from above)}$$

$$= \delta_1\Big(v_i + v_j \boldsymbol{u}_1(j)\Big) \qquad \text{(property 1 of } \boldsymbol{u}_1\text{)}$$

$$
\begin{aligned}
&= \Big(\max_{k\in[n]}|\boldsymbol{u}_1(k)|\Big)^{-1}\Big(1 + v_i v_j \boldsymbol{u}_1(j)\Big) && \text{(choice of } \delta_1\text{)}\\
&\geq \Big(\max\Big\{1, \frac{1-\phi}{\phi}\Big\}\Big)^{-1}\Big(1 + v_i v_j \boldsymbol{u}_1(j)\Big) && \text{(property 2 of } \boldsymbol{u}_1\text{)}\\
&= \min\Big\{1, \frac{\phi}{1-\phi}\Big\}\Big(1 + v_i v_j \boldsymbol{u}_1(j)\Big)\\
&\geq \min\Big\{1, \frac{\phi}{1-\phi}\Big\}\Big(1 - |\boldsymbol{u}_1(j)|\Big) && (v_i v_j \in \{\pm 1\})\\
&\geq \min\Big\{1, \frac{\phi}{1-\phi}\Big\}\cdot \phi && \text{(property 3 of } \boldsymbol{u}_1\text{)}\\
&= \min\Big\{\phi, \frac{\phi^2}{1-\phi}\Big\}
\end{aligned}
$$

Recall that the first pivot is chosen uniformly at random from the set of all n units, so that the probability unit i is chosen as pivot is $1/n$. In addition, the step size considered above is chosen with probability $1/2$. Thus, the probability of choosing the pivot to be i and the step size to be $\delta_1^{v_i}$ is $1/2n$. Using this and the above inequalities, we have that

$$
\begin{aligned}
\Pr(z_i = v_i, z_j = v_j) &\geq \Pr(p_1 = i, \delta_1 = \delta_1^{v_i}) \cdot \Pr(z_i = v_i, z_j = v_j \mid p_1 = i, \delta_1 = \delta_1^{v_i})\\
&\geq \frac{1}{2n} \cdot \frac{1}{2}\delta_1\Big(v_i \boldsymbol{u}_1(i) + v_j \boldsymbol{u}_1(j)\Big)\\
&\geq \frac{1}{4n}\min\Big\{\phi, \frac{\phi^2}{1-\phi}\Big\}.
\end{aligned}
$$

□

The lower bound in Lemma 4 holds for all pairs of treatment assignments and any covariate matrix. In this sense, Lemma 4 is a worst-case bound, and we conjecture that it is tight. However, we have observed that most of the second-order assignment probabilities are considerably closer to $1/4$ that what the bound in Lemma 4 suggests. Note that $1/4$ is the value of all second order assignment probabilities when the individual assignments are independent. We provide some theoretical justification for this observation in Lemma A16, which bounds the absolute difference between $1/4$ and all second order assignment probabilities. In particular, for design parameters in the range $\phi \in [0.8, 1]$, Lemma A16 provides a lower bound on all second order assignment probabilities which is independent of the sample size n. We remark that the fact that the lower bound becomes vacuous for $\phi < 0.8$ is a consequence of the proof technique in Lemma A16, and it is not a reflection of a property of the design itself.

Lemma A16. *The second-order assignment probabilities under the Gram–Schmidt Walk*

design satisfy

$$\left|\Pr\big((z_i, z_j) = \boldsymbol{v}\big) - 1/4\right| \leq \frac{1-\phi}{\phi} \quad \textit{for all} \quad i \neq j \quad \textit{and all} \quad \boldsymbol{v} \in \{\pm 1\}^2.$$

Proof. Let $i, j \in [n]$ be two arbitrary but distinct units. Consider a vector $\boldsymbol{\mu} = (\mu_1, \dots \mu_n)$ such that $\mu_k = 0$ for all $k \notin \{i, j\}$ and

$$\mu_i = \sqrt{1/2} \qquad \text{and} \qquad \mu_j = \begin{cases} \sqrt{1/2} & \text{if } \operatorname{Cov}(z_i, z_j) \geq 0, \\ -\sqrt{1/2} & \text{if } \operatorname{Cov}(z_i, z_j) < 0. \end{cases}$$

Observe that this implies that $\|\boldsymbol{\mu}\| = 1$.

The value of the quadratic form in $\operatorname{Cov}(\boldsymbol{z})$ evaluated at vector $\boldsymbol{\mu}$ is

$$\boldsymbol{\mu}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{\mu} = \mu_i^2 + \mu_j^2 + 2\mu_i\mu_j \operatorname{Cov}(z_i, z_j) = 1 + \left|\operatorname{Cov}(z_i, z_j)\right|,$$

because $2\mu_i\mu_j \operatorname{Cov}(z_i, z_j) = \left|\operatorname{Cov}(z_i, z_j)\right|$.

From Theorem 2, the largest eigenvalue of $\operatorname{Cov}(\boldsymbol{z})$ is at most $1/\phi$, so by the Courant–Fischer theorem,

$$1 + \left|\operatorname{Cov}(z_i, z_j)\right| = \boldsymbol{\mu}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{\mu} \leq \|\boldsymbol{\mu}\|^2 \cdot \max_{\|\boldsymbol{v}\|=1} \frac{\boldsymbol{v}^\intercal \operatorname{Cov}(\boldsymbol{z})\boldsymbol{v}}{\boldsymbol{v}^\intercal \boldsymbol{v}} \leq \|\boldsymbol{\mu}\|^2/\phi = 1/\phi.$$

Rearranging this inequality yields

$$\left|\operatorname{Cov}(z_i, z_j)\right| \leq \frac{1-\phi}{\phi}.$$

Recall that each unit is assigned to either treatment with equal probability so that $\operatorname{E}[z_i] = \operatorname{E}[z_j] = 0$, which implies that $\operatorname{Cov}(z_i, z_j) = \operatorname{E}[z_i z_j]$. By Lemma A11 for any treatment assignments $\boldsymbol{v} \in \{\pm 1\}^2$,

$$\left|\Pr\big((z_i, z_j) = \boldsymbol{v}\big) - 1/4\right| = \left|\operatorname{E}[z_i z_j]\right| \leq \frac{1-\phi}{\phi}. \qquad \square$$

A4.3 Alternative ridge loss estimator

Unlike the estimator of L presented in the main paper, the estimator presented in this subsection does not require that the second-order assignment probabilities are known, and it may therefore be used when it is infeasible to run a Monte Carlo to estimate these probabilities.

Recall the ridge loss in Theorem 3:

$$L = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi n}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}\right\|^2\right],$$

where $\boldsymbol{\mu} = (\boldsymbol{a}+\boldsymbol{b})/2$ is the average potential outcome vector. The alternative estimator of L is constructed in two steps. First, we construct an upper bound for L that only depends on the marginal distributions of the potential outcomes, then we provide an estimator of the bound.

Lemma A17. *For all covariate matrices $\boldsymbol{X}$ and all potential outcomes $\boldsymbol{a}$ and $\boldsymbol{b}$,*

$$L = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi n}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}\right\|^2\right] \leq \frac{L_a+L_b}{2}$$

where

$$L_a = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi n}\left\|\boldsymbol{a}-\boldsymbol{X}\boldsymbol{\beta}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}\right\|^2\right] \quad \textit{and} \quad L_b = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi n}\left\|\boldsymbol{b}-\boldsymbol{X}\boldsymbol{\beta}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}\right\|^2\right].$$

Proof. Let $\boldsymbol{\beta}_a^*$ and $\boldsymbol{\beta}_b^*$ be the minimizers of L_a and L_b. Define $\boldsymbol{\beta}_w = (\boldsymbol{\beta}_a^* + \boldsymbol{\beta}_b^*)/2$. By construction,

$$L = \min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left[\frac{1}{\phi n}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}\right\|^2\right] \leq \frac{1}{\phi n}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}_w\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}_w\right\|^2.$$

Recall that $\boldsymbol{\mu} = (\boldsymbol{a}+\boldsymbol{b})/2$, so

$$\frac{1}{\phi n}\left\|\boldsymbol{\mu}-\boldsymbol{X}\boldsymbol{\beta}_w\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\boldsymbol{\beta}_w\right\|^2 = \frac{1}{\phi n}\left\|\frac{(\boldsymbol{a}-\boldsymbol{X}\boldsymbol{\beta}_a^*)+(\boldsymbol{b}-\boldsymbol{X}\boldsymbol{\beta}_b^*)}{2}\right\|^2+\frac{\xi^2}{(1-\phi)n}\left\|\frac{\boldsymbol{\beta}_a^*+\boldsymbol{\beta}_b^*}{2}\right\|^2.$$

By applying the triangle inequality followed by Young's inequality for products, we obtain that for any two vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$,

$$\left\|\frac{\boldsymbol{v}_1+\boldsymbol{v}_2}{2}\right\|^2 \leq \left(\frac{\|\boldsymbol{v}_1\|+\|\boldsymbol{v}_2\|}{2}\right)^2 \leq \frac{\|\boldsymbol{v}_1\|^2+\|\boldsymbol{v}_2\|^2}{2}.$$

It follows that

$$\left\|\frac{(\boldsymbol{a}-\boldsymbol{X}\boldsymbol{\beta}_a^*)+(\boldsymbol{b}-\boldsymbol{X}\boldsymbol{\beta}_b^*)}{2}\right\|^2 \leq \frac{\|\boldsymbol{a}-\boldsymbol{X}\boldsymbol{\beta}_a^*\|^2+\|\boldsymbol{b}-\boldsymbol{X}\boldsymbol{\beta}_b^*\|^2}{2}$$

and

$$\left\| \frac{\boldsymbol{\beta}_a^* + \boldsymbol{\beta}_b^*}{2} \right\|^2 \leq \frac{\|\boldsymbol{\beta}_a^*\|^2 + \|\boldsymbol{\beta}_b^*\|^2}{2}. \qquad \square$$

It remains to construct estimators for L_a and L_b. Because of symmetry, the estimators will be identical, so we will focus on an estimator for L_a here. Suppose momentarily that $\boldsymbol{\beta}_a^*$ is known. The loss L_a is still unobserved because we generally do not observe all potential outcomes in $\boldsymbol{a}$ even after assignment. The quantity can, however, be estimated using a Horvitz–Thompson-type estimator:

$$\widehat{L}_a = \frac{2}{\phi n} \sum_{i \in Z^+} \left(y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta}_a^* \rangle \right)^2 + \frac{\xi^2}{(1-\phi)n} \left\| \boldsymbol{\beta}_a^* \right\|^2.$$

The concern with this approach is, of course, that $\boldsymbol{\beta}_a^*$ is not known, and it must generally be estimated itself.

One way forward is to use external data to estimate the function. Let $\boldsymbol{\beta}_e$ be a function that is either fixed or random but independent of treatment assignment. With probability one, we have

$$L_a = \min_{\boldsymbol{\beta} \in \mathbb{R}^d} \left[\frac{1}{\phi n} \left\| \boldsymbol{a} - \boldsymbol{X}\boldsymbol{\beta} \right\|^2 + \frac{\xi^2}{(1-\phi)n} \left\| \boldsymbol{\beta} \right\|^2 \right] \leq \frac{1}{\phi n} \left\| \boldsymbol{a} - \boldsymbol{X}\boldsymbol{\beta}_e \right\|^2 + \frac{\xi^2}{(1-\phi)n} \left\| \boldsymbol{\beta}_e \right\|^2.$$

Hence using the estimator $\widehat{L}_a$ with $\boldsymbol{\beta}_e$ substituted for $\boldsymbol{\beta}_a^*$ will yield a conservative estimator of L_a.

The concern now is that $\boldsymbol{\beta}_e$ may far from $\boldsymbol{\beta}_a^*$ if the external data are not representative of the potential outcomes in the experiment. The estimator will in that case be exceedingly conservative. The most natural way to estimate $\boldsymbol{\beta}_a^*$ may be to use the experimental data itself. Consider the estimator

$$\widehat{L}_{reg,a} = \min_{\boldsymbol{\beta} \in \mathbb{R}^d} \left[\frac{2}{\phi n} \sum_{i \in Z^+} \left(y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle \right)^2 + \frac{\xi^2}{(1-\phi)n} \left\| \boldsymbol{\beta} \right\|^2 \right].$$

The concern with this estimator is that it is generally easier to predict the potential outcomes in Z^+ than it is in $[n]$, so $\widehat{L}_{reg,a}$ will be systematically lower than L_a. Such overfitting will be mild when d is small relative to n or when ϕ is close to one. It may be reasonable to ignore the bias of $\widehat{L}_{reg,a}$ in those cases. The bound in Lemma A17 is generally not tight, so it provides some margin of error with respect to underestimation of L_a and L_b.

A4.4 Ridge loss estimator under non-uniform assignment probabilities

The ridge loss estimator discussed in the main paper presumes that the first-order assignment probabilities are uniform. In this section, we describe the modifications needed for non-uniform assignment probabilities. The modified estimator can thus be used with the non-uniform version of the Gram–Schmidt Walk design presented in Section 13.1 in the main paper.

For each unit $i \in [n]$, we use the shorthand that $\pi_i = \Pr(z_i = 1)$. As in Section A1.2, we construct vectors which contain the potential outcomes normalized by the assignment probabilities:

$$\widetilde{\boldsymbol{a}} = \left(\frac{a_1}{2\pi_1}, \ldots, \frac{a_n}{2\pi_n}\right), \qquad \widetilde{\boldsymbol{b}} = \left(\frac{b_1}{2(1-\pi_1)}, \ldots, \frac{b_n}{2(1-\pi_n)}\right) \qquad \text{and} \qquad \widetilde{\boldsymbol{\mu}} = \frac{\widetilde{\boldsymbol{a}} + \widetilde{\boldsymbol{b}}}{2}.$$

The ridge regression loss is now $L = \widetilde{\boldsymbol{\mu}}^\mathsf{T} \boldsymbol{Q} \widetilde{\boldsymbol{\mu}}$. As before, this contains unobservable quantities (e.g., $a_i b_i$). We use the same type of bound discussed in Section 11.3 in the main paper to obtain an observable upper bound:

$$4nL \leq \widetilde{\boldsymbol{a}}^\mathsf{T} \big[\boldsymbol{Q} + \operatorname{diag}(\boldsymbol{Q})\big] \widetilde{\boldsymbol{a}} + \widetilde{\boldsymbol{b}}^\mathsf{T} \big[\boldsymbol{Q} + \operatorname{diag}(\boldsymbol{Q})\big] \widetilde{\boldsymbol{b}} + 2\widetilde{\boldsymbol{a}}^\mathsf{T} \big[\boldsymbol{Q} - \operatorname{diag}(\boldsymbol{Q})\big] \widetilde{\boldsymbol{b}}.$$

We can write the bound as

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(1 + \mathbb{1}[i=j])\widetilde{a}_i\widetilde{a}_j q_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n}(1 + \mathbb{1}[i=j])\widetilde{b}_i\widetilde{b}_j q_{ij} + 2\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbb{1}[i \neq j]\widetilde{a}_i\widetilde{b}_j q_{ij},$$

where $\widetilde{a}_1, \ldots, \widetilde{a}_n$ and $\widetilde{b}_1, \ldots, \widetilde{b}_n$ are the elements of $\widetilde{\boldsymbol{a}}$ and $\widetilde{\boldsymbol{b}}$.

This suggests the following estimator of L in the non-uniform case:

$$\begin{aligned}\widetilde{L} = \frac{1}{n}\sum_{i\in Z^+}\sum_{j\in Z^+}(1 + \mathbb{1}[i=j])\frac{\widetilde{y}_i\widetilde{y}_j q_{ij}}{4\Pr(z_i = 1, z_j = 1)} \\ + \frac{1}{n}\sum_{i\in Z^-}\sum_{j\in Z^-}(1 + \mathbb{1}[i=j])\frac{\widetilde{y}_i\widetilde{y}_j q_{ij}}{4\Pr(z_i = -1, z_j = -1)} \\ + \frac{2}{n}\sum_{i\in Z^+}\sum_{j\in Z^-}\mathbb{1}[i \neq j]\frac{\widetilde{y}_i\widetilde{y}_j q_{ij}}{4\Pr(z_i = 1, z_j = -1)},\end{aligned}$$

where $\widetilde{\boldsymbol{y}} = (\widetilde{y}_1, \ldots, \widetilde{y}_n)$ is an observed vector with elements

$$\widetilde{y}_i = \begin{cases} \widetilde{a}_i & \text{if } z_i = 1, \\ \widetilde{b}_i & \text{if } z_i = -1. \end{cases}$$

Using Lemma A11 and the same approach as in Section A4.1, we can write the estimator as

$$\widetilde{L} = \frac{1}{n} \widetilde{\boldsymbol{y}}^{\intercal} \widetilde{\boldsymbol{Q}} \widetilde{\boldsymbol{y}},$$

where the representative element of the random matrix $\widetilde{\boldsymbol{Q}}$ is

$$\widetilde{q}_{ij} = \begin{cases} q_{ii} / \big(1 + z_i \operatorname{E}[z_i]\big) & \text{if } i = j, \\ q_{ij} / \big(1 + z_i \operatorname{E}[z_i] + z_j \operatorname{E}[z_j] + z_i z_j \operatorname{E}[z_i z_j]\big) & \text{otherwise.} \end{cases}$$

If one could extend Lemma 4 to non-uniform first-order assignment probabilities, the proof of conservativeness of $\widetilde{L}$ would be identical to the proof of Proposition 4.

A5 Extensions

In this section, we prove the group balanced property of the Balanced Gram–Schmidt Walk design.

A5.1 Balanced Gram–Schmidt Walk Design (Proposition 6)

The following discussion proves the group-balancing property of the version of the Gram–Schmidt Walk design presented in Section 13.2 in the main paper. Namely, the treatment group sizes differ by at most one and under appropriate conditions, the group sizes are fixed.

Proposition 6. *With probability one under the group-balanced Gram–Schmidt Walk design,*

$$\big| n_+ - \operatorname{E}[n_+] \big| < 1.$$

If $\operatorname{E}[n_+]$ *is an integer, then* $n_+ = \operatorname{E}[n_+]$ *with probability one.*

Proof. Note that for any assignment vector $\boldsymbol{z} \in \{\pm 1\}^n$, the difference between the sizes of the two treatment groups is given by $n_+ - n_- = \langle \mathbf{1}, \boldsymbol{z} \rangle$. Taking expectations, we arrive at

$$\operatorname{E}[n_+ - n_-] = \operatorname{E}[\langle \mathbf{1}, \boldsymbol{z} \rangle] = \langle \mathbf{1}, \operatorname{E}[\boldsymbol{z}] \rangle = \langle \mathbf{1}, \boldsymbol{z}_1 \rangle,$$

where $\boldsymbol{z}_1$ is the initial fractional assignment and the last equality follows by the martingale property. We can express the difference between group sizes in terms of the iterative updates made by the group-balanced Gram–Schmidt Walk as

$$n_+ - n_- = \langle \mathbf{1}, \boldsymbol{z} \rangle = \langle \mathbf{1}, \sum_{t=1}^{T} \delta_t \boldsymbol{u}_t + \boldsymbol{z}_1 \rangle = \sum_{t=1}^{T} \delta_t \langle \mathbf{1}, \boldsymbol{u}_t \rangle + \langle \mathbf{1}, \boldsymbol{z}_1 \rangle = \sum_{t=1}^{T} \delta_t \langle \mathbf{1}, \boldsymbol{u}_t \rangle + \mathrm{E}[n_+ - n_-].$$

For all but the final iteration, there is at least one alive unit which is not the pivot. Thus, by the additional constraint in the group-balanced Gram–Schmidt Walk, we have that $\langle \mathbf{1}, \boldsymbol{u}_t \rangle = 0$ for $t = 1, 2, \ldots T-1$. This means that in the above sum, all terms are zero except possibly the last term corresponding to the final iteration T. Applying this and rearranging the expressions above yields

$$(n_+ - n_-) - \mathrm{E}[n_+ - n_-] = \delta_T \langle \mathbf{1}, \boldsymbol{u}_T \rangle.$$

The remainder of the proof considers two cases of the final iteration. The first case is that there is more than one alive unit at the final iteration. In this case, the additional balancing constraint ensures that $\langle \mathbf{1}, \boldsymbol{u}_T \rangle = 0$. Thus, we have that $n_+ - n_- = \mathrm{E}[n_+ - n_-]$ when there is more than one alive unit at the last iteration.

The second case to consider is that the pivot is the only alive unit at the last iteration. In this case, we have that the update vector $\boldsymbol{u}_T$ has 1 in the entry corresponding to the pivot and 0 in the remaining entries. Thus, we have that $\langle \mathbf{1}, \boldsymbol{u}_T \rangle = 1$ in this case. The two possible values of the step size δ_T are $1 - \boldsymbol{z}_T(p)$ and $1 + \boldsymbol{z}_T(p)$. Because $\boldsymbol{z}_T(p) \in (-1, 1)$, we have that $|\delta_T| < 2$, regardless of which possible value is chosen. Thus, we obtain the upper bound

$$|(n_+ - n_-) - \mathrm{E}[n_+ - n_-]| = |\delta_T \langle \mathbf{1}, \boldsymbol{u}_T \rangle| = |\delta_T| \cdot |\langle \mathbf{1}, \boldsymbol{u}_T \rangle| = |\delta_T| < 2.$$

The desired result follows from simple manipulation of terms. Because $n_+ + n_- = n$, we have that $n_- = n - n_+$. Substituting this into the term on the left hand side, we obtain

$$(n_+ - n_-) - \mathrm{E}[n_+ - n_-] = (2n_+ - n) - (2\,\mathrm{E}[n_+] - n) = 2(n_+ - \mathrm{E}[n_+])$$

and now the upper bound above yields that $|n_+ - \mathrm{E}[n_+]| < 1$, as desired.

It directly follows that if $\mathrm{E}[n_+]$ is an integer, then $n_+ = \mathrm{E}[n_+]$ with probability one. To see this, observe that if $\mathrm{E}[n_+]$ is an integer, then $n_+ - \mathrm{E}[n_+]$ is also an integer. Thus, the condition $|n_+ - \mathrm{E}[n_+]| < 1$ implies that $n_+ - \mathrm{E}[n_+] = 0$ so that these two quantities are equal. □

A6 Proofs of corollaries

For completeness and clarity, we here provide proofs of all corollaries.

Corollary 1. *Under the Gram–Schmidt Walk design,* $\Pr(z_i = 1) = 1/2$ *for all* $i \in [n]$.

Proof. Lemma 2 implies that $\mathrm{E}[z_i] = 0$ for all units under the uniform version of the Gram–Schmidt Walk design. Because $z_i \in \{\pm 1\}$, we also have

$$\mathrm{E}[z_i] = 1 \times \Pr(z_i = 1) + (-1) \times \Pr(z_i = -1).$$

This implies that $\Pr(z_i = 1) = \Pr(z_i = -1)$. By the probability axioms, $\Pr(z_i = 1) + \Pr(z_i = -1) = 1$, so the terms must be one half if they are equal. □

Corollary 2. *The Horvitz–Thompson estimator is unbiased for the average treatment effect under the Gram–Schmidt Walk design.*

Proof. Lemma A1 showed that $\widehat{\tau} - \tau = 2\langle \boldsymbol{z}, \boldsymbol{\mu} \rangle / n$. Taking expectations yields

$$\mathrm{E}[\widehat{\tau}] - \tau = \frac{2}{n} \big\langle \mathrm{E}[\boldsymbol{z}], \boldsymbol{\mu} \big\rangle.$$

Lemma 2 implies that $\mathrm{E}[\boldsymbol{z}] = \boldsymbol{0}$. □

Corollary 3. *Under the Gram–Schmidt Walk design, the imbalance of any linear function* $\boldsymbol{\theta}$ *of the covariates is bounded by*

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] \leq \frac{\xi^2}{1 - \phi} \boldsymbol{\theta}^\intercal \boldsymbol{H} \boldsymbol{\theta}.$$

Proof. Following the derivation in Section 8.1, we have

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] = \boldsymbol{\theta}^\intercal \operatorname{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z}) \boldsymbol{\theta}.$$

Applying the matrix bound in Lemma 3 yields the upper bound in the corollary. □

Corollary 4. *Suppose that a sequence of covariate matrices* $\boldsymbol{X}$ *and linear functions* $\boldsymbol{\theta}$ *satisfies* $\xi = \mathcal{O}(\sqrt{d \log(n)})$ *and* $\|\boldsymbol{\theta}\|^2 / \|\boldsymbol{X}\boldsymbol{\theta}\|^2 = \mathcal{O}(1/n)$. *Then the relative imbalance between the Gram–Schmidt Walk design and the fully randomized design is*

$$\frac{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big]}{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big]} = \mathcal{O}\bigg(\frac{d \log(n)}{(1 - \phi) n}\bigg).$$

Proof. We first show that $\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big] = \|\boldsymbol{X}\boldsymbol{\theta}\|^2$ when $\boldsymbol{z}_{\text{FR}}$ is the assignment vector under the fully randomized design. Write

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big] = \mathrm{E}\big[\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}} \boldsymbol{z}_{\text{FR}}^\intercal \boldsymbol{X}\boldsymbol{\theta}\big] = \boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \,\mathrm{E}\big[\boldsymbol{z}_{\text{FR}} \boldsymbol{z}_{\text{FR}}^\intercal\big] \boldsymbol{X}\boldsymbol{\theta},$$

where the final equality follows from that $\boldsymbol{\theta}$ and $\boldsymbol{X}$ are not random. Note that $\mathrm{E}\big[\boldsymbol{z}_{\text{FR}} \boldsymbol{z}_{\text{FR}}^\intercal\big] = \mathrm{Cov}(\boldsymbol{z}_{\text{FR}}) = \boldsymbol{I}$ because $\mathrm{E}[\boldsymbol{z}_{\text{FR}}] = \boldsymbol{0}$. It follows that

$$\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \,\mathrm{E}\big[\boldsymbol{z}_{\text{FR}} \boldsymbol{z}_{\text{FR}}^\intercal\big] \boldsymbol{X}\boldsymbol{\theta} = \boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{X}\boldsymbol{\theta} = \|\boldsymbol{X}\boldsymbol{\theta}\|^2.$$

We have from Proposition 1 that $\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] \leq \xi^2 \|\boldsymbol{\theta}\|^2/(1-\phi)$. Taken together, we have

$$\frac{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big]}{\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z}_{\text{FR}})^2\big]} = \frac{\xi^2 \|\boldsymbol{\theta}\|^2}{(1-\phi)\|\boldsymbol{X}\boldsymbol{\theta}\|^2}.$$

By assumption, $\xi^2 = \mathcal{O}(d\log(n))$ and $\|\boldsymbol{\theta}\|^2/\|\boldsymbol{X}\boldsymbol{\theta}\|^2 = \mathcal{O}(1/n)$. □

Corollary 5. *Under the Gram–Schmidt Walk design, the imbalance of any linear function $\boldsymbol{\theta}$ of the covariates is bounded by*

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] \leq \Big(\phi\lambda_G^{-1} + (1-\phi)\xi^{-2}\Big)^{-1} \|\boldsymbol{\theta}\|^2.$$

Proof. By Proposition 2, we have

$$\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big] = \boldsymbol{\theta}^\intercal \,\mathrm{Cov}(\boldsymbol{X}^\intercal \boldsymbol{z})\boldsymbol{\theta} \leq \boldsymbol{\theta}^\intercal \Big(\phi(\boldsymbol{X}^\intercal \boldsymbol{X})^\dagger + (1-\phi)(\xi^2 \boldsymbol{\Pi})^\dagger\Big)^\dagger \boldsymbol{\theta}.$$

We denoted the maximum eigenvalue of the Gram matrix $\boldsymbol{X}^\intercal \boldsymbol{X}$ with λ_G, and by virtue of being a projection matrix, the maximum eigenvalue of $\boldsymbol{\Pi}$ is one. Hence, the maximum eigenvalue of the matrix

$$\Big(\phi(\boldsymbol{X}^\intercal \boldsymbol{X})^\dagger + (1-\phi)(\xi^2 \boldsymbol{\Pi})^\dagger\Big)^\dagger \qquad \text{is} \qquad \Big(\phi\lambda_G^{-1} + (1-\phi)\xi^{-2}\Big)^{-1}.$$

This implies that

$$\boldsymbol{\theta}^\intercal \Big(\phi(\boldsymbol{X}^\intercal \boldsymbol{X})^\dagger + (1-\phi)(\xi^2 \boldsymbol{\Pi})^\dagger\Big)^\dagger \boldsymbol{\theta} \leq \Big(\phi\lambda_G^{-1} + (1-\phi)\xi^{-2}\Big)^{-1} \|\boldsymbol{\theta}\|^2.$$ □

Corollary 6. *When the covariates are decorrelated, so that $\boldsymbol{X}^\intercal \boldsymbol{X} = n\boldsymbol{I}$, the upper bounds on $\mathrm{E}\big[(\boldsymbol{\theta}^\intercal \boldsymbol{X}^\intercal \boldsymbol{z})^2\big]$ in Corollaries 3 and 5 coincide for all linear functions $\boldsymbol{\theta} \in \mathbb{R}^d$:*

$$\frac{\xi^2}{1-\phi} \boldsymbol{\theta}^\intercal \boldsymbol{H}\boldsymbol{\theta} = \Big(\phi n^{-1} + (1-\phi)\xi^{-2}\Big)^{-1} \|\boldsymbol{\theta}\|^2.$$

Proof. The proof of Proposition 2 shows that

$$\frac{\xi^2}{1-\phi}\boldsymbol{H} = \Big(\phi(\boldsymbol{X}^\intercal\boldsymbol{X})^\dagger + (1-\phi)(\xi^2\boldsymbol{\Pi})^\dagger\Big)^\dagger.$$

When the covariates is in a scaled isotropic position, we have $\boldsymbol{X}^\intercal\boldsymbol{X} = n\boldsymbol{I}$ and $\boldsymbol{\Pi} = \boldsymbol{I}$. We can therefore write

$$\frac{\xi^2}{1-\phi}\boldsymbol{H} = \Big(\phi(n\boldsymbol{I})^\dagger + (1-\phi)(\xi^2\boldsymbol{I})^\dagger\Big)^\dagger = \Big(\phi n^{-1} + (1-\phi)\xi^{-2}\Big)^{-1}\boldsymbol{I}.$$

The corollary then follows from the fact that $\boldsymbol{\theta}^\intercal\boldsymbol{I}\boldsymbol{\theta} = \|\boldsymbol{\theta}\|^2$. □

Corollary 7. *The random interval centered at $\widehat{\tau}$ with radius $\gamma_\alpha = \sqrt{8\log(2/\alpha)L/n}$ is a valid $(1-\alpha)$-confidence interval:*

$$\Pr\big(\widehat{\tau} - \gamma_\alpha \le \tau \le \widehat{\tau} + \gamma_\alpha\big) \ge 1 - \alpha.$$

Proof. Write the probability as

$$\Pr\big(\widehat{\tau} - \gamma_\alpha \le \tau \le \widehat{\tau} + \gamma_\alpha\big) = \Pr\big(|\tau - \widehat{\tau}| \le \gamma_\alpha\big) \ge 1 - \Pr\big(|\tau - \widehat{\tau}| \ge \gamma_\alpha\big).$$

The right hand side is bound by Proposition 3 by

$$1 - \Pr\big(|\tau - \widehat{\tau}| \ge \gamma_\alpha\big) \ge 1 - 2\exp\left(\frac{-\gamma_\alpha^2 n}{8L}\right).$$

Finally, note that $\gamma_\alpha = \sqrt{8\log(2/\alpha)L/n}$ yields

$$2\exp\left(\frac{-\gamma_\alpha^2 n}{8L}\right) = \alpha.$$ □

Corollary 8. *Under the non-uniform Gram–Schmidt Walk design,*

$$\Pr(z_i = 1) = \pi_i \qquad \textit{for all} \qquad i \in [n].$$

Proof. We have $\mathrm{E}[z_i] = 2\pi_i - 1$ for all units under the non-uniform version of the Gram–Schmidt Walk design with $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n)$. This follows from the martingale property of fractional assignments (Lemma 2) together with the fact that fractional assignments are initialized as $\boldsymbol{z}_1 \leftarrow 2\boldsymbol{\pi} - \mathbf{1}$. Note that $\Pr(z_i = -1) = 1 - \Pr(z_i = 1)$. It follows that

$$\mathrm{E}[z_i] = \Pr(z_i = 1) - \Pr(z_i = -1) = 2\Pr(z_i = 1) - 1$$

and the corollary follows by rearranging terms. □

Supplement B: Implementation details

Contents

B1 Implementation of the Gram–Schmidt Walk design

The most computationally intensive aspect of the Gram–Schmidt Walk is the computation of the step direction $\boldsymbol{u}_t$. Although it is defined as the solution to an optimization problem, it may be obtained efficiently by solving a system of linear equations. Computational speed ups may be obtained by pre-computing and maintaining a certain matrix factorization, decreasing the cost of repeated linear system solves at each iteration. In this section, we provide details of such an efficient implementation.

B1.1 Derivation of the step direction

Recall that at each iteration t, the step direction $\boldsymbol{u}_t$ is defined as the vector which has coordinates $\boldsymbol{u}_t(i) = 0$ for $i \notin \mathcal{A}_t$, coordinate $\boldsymbol{u}_t(p_t) = 1$ for the pivot unit p_t, and the remaining coordinates are the solution to

$$\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = \arg\min_{\boldsymbol{u}} \|\boldsymbol{b}_{p_t} + \sum_{i \notin \mathcal{A}_t \setminus p_t} \boldsymbol{u}(i)\boldsymbol{b}_i\|^2 \ .$$

The minimization above is a least squares problem and the solution may be obtained by solving a system of linear equations. Let k be the number of units which are alive and not

the pivot, i.e., $k = |\mathcal{A}_t \setminus p_t|$, and let $\boldsymbol{B}_t$ be the $(n+d)$-by-k matrix with columns $\boldsymbol{b}_i$ for $i \in \mathcal{A}_t \setminus p_t$. As the augmented covariate vectors are linearly independent, the coordinates $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)$ that minimize the quantity $\|\boldsymbol{b}_{p_t} + \boldsymbol{B}_t \boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)\|^2$ are given by the normal equations

$$\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = -\left(\boldsymbol{B}_t^\intercal \boldsymbol{B}_t\right)^{-1} \boldsymbol{B}_t^\intercal \boldsymbol{b}_{p_t} \ .$$

Let $\boldsymbol{X}_t$ denote the row-submatrix of $\boldsymbol{X}$ with rows $\mathcal{A}_t \setminus p_t$. Using our specific form of $\boldsymbol{B}$, and by direct calculation and application of the Woodbury identity lemma, we obtain that

$$\left(\boldsymbol{B}_t^\intercal \boldsymbol{B}_t\right)^{-1} = \left(\phi \boldsymbol{I}_k + \xi^{-2}(1-\phi)\boldsymbol{X}_t \boldsymbol{X}_t^\intercal\right)^{-1} = \phi^{-1}\left[\boldsymbol{I}_k - \boldsymbol{X}_t\left(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \frac{\xi^2 \phi}{1-\phi}\boldsymbol{I}_d\right)^{-1} \boldsymbol{X}_t^\intercal\right] \ .$$

By again using our specific form of input matrix $\boldsymbol{B}$, a direct calculation yields that

$$\boldsymbol{B}_t^\intercal \boldsymbol{b}_{p_t} = \xi^{-2}(1-\phi)\boldsymbol{X}_t \boldsymbol{x}_{p_t} \ .$$

Thus, we obtain a form for the relevant coordinates in the update direction vector $\boldsymbol{u}_t$

$$\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = -\left(\frac{1-\phi}{\xi^2 \phi}\right) \underbrace{\boldsymbol{X}_t}_{n \times d}\left[\boldsymbol{x}_{p_t} - \underbrace{\left(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\right)^{-1}}_{d \times d} \underbrace{\boldsymbol{X}_t^\intercal \boldsymbol{X}_t}_{d \times d} \boldsymbol{x}_{p_t}\right] \ , \tag{B1}$$

which involves smaller matrices of size $d \times d$, rather than $n \times n$. In the next few paragraphs, we show how computing and maintaining factorizations of these smaller matrices results in faster computations of the step direction $\boldsymbol{u}_t$. We are chiefly concerned with computing and maintaining a factorization of the matrix $(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$. We describe an implementation which uses the Cholesky factorization, although there are several appropriate alternatives.

B1.2 Cholesky factorizations

Here, we briefly review Cholesky factorizations and their computational properties. The *Cholesky factorizatio*n of an n-by-n symmetric positive definite matrix $\boldsymbol{A}$ is the unique factorization $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\intercal$, where $\boldsymbol{L}$ is lower triangular. Given the matrix $\boldsymbol{A}$, the matrix $\boldsymbol{L}$ may be obtained using $\mathcal{O}(n^3)$ arithmetic operations. Once the Cholesky factorization $\boldsymbol{L}$ is obtained, solutions $\boldsymbol{x}$ to the system of linear equations $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ may be computed using $\mathcal{O}(n^2)$ arithmetic operations by using a forward-backward algorithm which leverages the triangular structure of $\boldsymbol{L}$. In general, solving systems of linear equations takes $\mathcal{O}(n^3)$

arithmetic operations[3] and so if many linear system solves are required, then computing the factorization and using the faster forward-backward algorithm yields computational speed-ups. Suppose that $\boldsymbol{A}$ is a positive definite matrix with Cholesky factorization $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^\intercal$ and that the rank-1 updated matrix $\boldsymbol{A} + \boldsymbol{v}\boldsymbol{v}^\intercal$ has Cholesky factorization $\boldsymbol{A} + \boldsymbol{v}\boldsymbol{v}^\intercal = \boldsymbol{L}_+\boldsymbol{L}_+^\intercal$. Given the original factorization $\boldsymbol{L}$ and the vector $\boldsymbol{v}$, the updated factorization $\boldsymbol{L}_+$ may be computed using $\mathcal{O}(n^2)$ arithmetic computations, without extra memory allocation. Updating in this way is a much more efficient way to maintain the factorization than explicitly computing $\boldsymbol{A} + \boldsymbol{v}\boldsymbol{v}^\intercal$ and its factorization directly. The same technique may be used for rank-1 downdates $\boldsymbol{A} - \boldsymbol{v}\boldsymbol{v}^\intercal$ when the updated matrix remains positive definite. For more details, see Stewart (1998); Trefethen & Bau (1997).

B1.3 Computing and maintaining factorizations

Before the first pivot is chosen, we have that $\boldsymbol{X}_t = \boldsymbol{X}$, as no rows of $\boldsymbol{X}$ have been decided. Thus, we compute $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$ directly and then compute a Cholesky factorization. Computing the matrix directly requires $\mathcal{O}(nd^2)$ time and computing the factorization requires $\mathcal{O}(d^3)$ time. Each time a variable $i \in [n]$ is frozen or chosen as the pivot, the set $\mathcal{A}_t \setminus p_t$ is updated and so we must update the factorization $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$. The update consists of removing the row vector $\boldsymbol{x}_i$ from $\boldsymbol{X}_t$. One can see that this corresponds to a rank-1 downdate to the entire matrix $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$. Rank-1 downdates to a Cholesky factorization may be computed in-place, using $\mathcal{O}(d^2)$ arithmetic operations. Because there will be at most n rank-1 updates to this factorization, the total update cost is $\mathcal{O}(nd^2)$ arithmetic operations. Thus, the total computational cost of maintaining this Cholesky factorization is $\mathcal{O}(nd^2)$ arithmetic operations and $\mathcal{O}(d^2)$ memory.

B1.4 Computing step directions

Assume that at each iteration, we have a Choleksy factorization of the matrix $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$. By (B1), we can solve for the relevant coordinates in the step direction $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)$ using the following three computations:

1. $\boldsymbol{a}_t^{(1)} = \boldsymbol{X}_t^\intercal\boldsymbol{X}_t\boldsymbol{x}_{p_t}$
2. $\boldsymbol{a}_t^{(2)} = \left(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d\right)^{-1}\boldsymbol{a}_t^{(1)}$
3. $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = -\xi^{-2}\phi^{-1}(1-\phi)\boldsymbol{X}_t\left(\boldsymbol{x}_{p_t} - \boldsymbol{a}_t^{(2)}\right)$

If the matrix $\boldsymbol{X}_t^\intercal\boldsymbol{X}_t$ is explicitly available at the beginning of each iteration, then computing $\boldsymbol{a}_t^{(1)}$ can be done in $\mathcal{O}(d^2)$ time by matrix-vector multiplication. While it is possible to

[3]While there are algorithms based on fast matrix multiplication that are asymptotically faster, they do not meaningfully change this discussion for realistic values of n.

maintain $\boldsymbol{X}_t^\intercal \boldsymbol{X}_t$ explicitly, it requires an extra $\mathcal{O}(d^2)$ memory. On the other hand, if $\boldsymbol{X}_t^\intercal \boldsymbol{X}_t$ is not explicitly available, then $\boldsymbol{a}_t^{(1)}$ may be obtained from a factorization of $(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$, as

$$\boldsymbol{a}_t^{(1)} = \left(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\right)\boldsymbol{x}_{p_t} - \left(\frac{\xi^2\phi}{1-\phi}\right)\boldsymbol{x}_{p_t} \ ,$$

which saves $\mathcal{O}(d^2)$ memory and incurs only a slightly larger arithmetic cost of $\mathcal{O}(d^2+d)$. Next, one may compute $\boldsymbol{a}_t^{(2)}$ using $\mathcal{O}(d^2)$ arithmetic operations via a forward-backward solver on the Cholesky factorization. Finally, computing $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)$ may be done in $\mathcal{O}(nd)$ operations via matrix-vector multiplication. Thus, the per iteration cost of computing $\boldsymbol{u}_t$ given a factorized $(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$ is $\mathcal{O}(nd+d^2)$ arithmetic operations. Because there are at most n iterations, this leads to a total cost of $\mathcal{O}(n^2d + nd^2)$ arithmetic operations. We remark that $\mathcal{O}(n)$ memory is required for storing vectors such as $\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t)$.

Thus, an assignment may be sampled from the Gram–Schmidt Walk design using $\mathcal{O}(n^2d)$ arithmetic computations and $\mathcal{O}(n+d^2)$ extra storage when implemented with these matrix factorizations. There are several practical considerations when implementing this algorithm. First, for what values of n and d is this practically feasible? Of course, this depends on the computing infrastructure which is available to experimenters, but roughly speaking, sampling from the Gram–Schmidt Walk is as computationally intensive as computing all pairs of inner products of covariates $\boldsymbol{x}_1, \boldsymbol{x}_2 \ldots \boldsymbol{x}_n \in \mathbb{R}^d$. Computing these inner products requires $\mathcal{O}(n^2d)$ arithmetic operations and computing this matrix of inner products $\boldsymbol{X}\boldsymbol{X}^\intercal$ is a pre-processing step of our implementation. The analysis above shows that the remainder of the algorithm requires roughly the same number of arithmetic operations. Thus, sampling from the Gram–Schmidt Walk should be practically feasible in cases where computing all inner products is practically feasible. A second practical consideration are the computational speed-ups for sampling more than one assignment from the design. When sampling many assignments from the Gram–Schmidt Walk, we may greatly reduce the run time by computing the initial cholesky factorization of $(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$ and re-using it for each sample. Finally, we remark that although our focus is to speed up the Gram–Schmidt Walk when we use the augmented covariate vectors, similar matrix factorizations may also be used to decrease the asymptotic run time of the general Gram–Schmidt Walk.

B1.5 Proof of asymptotic runtime (Proposition 5)

Proposition 5. *Assignments from the Gram–Schmidt Walk design can be sampled using $\mathcal{O}(dn^2)$ arithmetic operations and $\mathcal{O}(d^2+n)$ additional storage.*

Proof. As detailed in Section B1, an implementation which achieves these computational resource guarantees is to store and maintain a Cholesky factorization of the matrix $(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$, where $\boldsymbol{X}_t$ denotes the row-submatrix of $\boldsymbol{X}$ with rows $\mathcal{A}_t \setminus p_t$. Constructing

the matrix $\boldsymbol{X}^\intercal \boldsymbol{X}$ requires $\mathcal{O}(nd^2)$ arithmetic operations and $\mathcal{O}(d^2)$ space. Initially computing a Cholesky factorization of this matrix requires $\mathcal{O}(d^3)$ arithmetic operations and may be done in place. Updating the Cholesky factorization may be done using $\mathcal{O}(nd)$ arithmetic operations in place and this is done at most n times. Thus, constructing and maintaining the Cholesky factorization requires at most $\mathcal{O}(n^2 d)$ arithmetic operations and $\mathcal{O}(d^2)$ space, assuming that $d \leq n$.

Finally, computing the step direction $\boldsymbol{u}_t$ at each iteration requires $\mathcal{O}(nd)$ arithmetic operations and $\mathcal{O}(n)$ space given the above Cholesky factorization. This happens for at most n iterations, yielding a total of $\mathcal{O}(n^2 d)$ arithmetic operations and $\mathcal{O}(n)$ space. Thus, combining the computational requirements of maintaining the Cholesky factorizaiton and computing the step directions $\boldsymbol{u}_t$ yields a total requirement of $\mathcal{O}(n^2 d)$ arithmetic operations and $\mathcal{O}(n+d^2)$ additional storage to generate one assignment vector using the Gram–Schmidt Walk. □

B2 Efficient computation of the ridge loss estimator

In this section, we show that the entries of the ridge loss matrix

$$\boldsymbol{Q} = (\boldsymbol{B}^\intercal \boldsymbol{B})^{-1} = (\phi \boldsymbol{I} + \xi^{-2}(1-\phi)\boldsymbol{X}\boldsymbol{X}^\intercal)^{-1},$$

may be computed using $\mathcal{O}(n^2 d)$ arithmetic operations and $\mathcal{O}(d^2)$ additional storage.

B2.1 Derivation of the entries in the $\boldsymbol{Q}$ matrix

By the Woodbury identity,

$$\boldsymbol{Q} = (\phi \boldsymbol{I} + \xi^{-2}(1-\phi)\boldsymbol{X}\boldsymbol{X}^\intercal)^{-1} = \phi^{-1}\left[\boldsymbol{I} - \boldsymbol{X}\left(\boldsymbol{X}^\intercal \boldsymbol{X} + \frac{\xi^2 \phi}{1-\phi}\boldsymbol{I}\right)^{-1}\boldsymbol{X}^\intercal\right]$$

and so the entires of $\boldsymbol{Q}$ are given by

$$\begin{aligned} q_{ij} = \boldsymbol{e}_i^\intercal \boldsymbol{Q} \boldsymbol{e}_j &= \phi^{-1}\boldsymbol{e}_i^\intercal \left[\boldsymbol{I} - \boldsymbol{X}\left(\boldsymbol{X}^\intercal \boldsymbol{X} + \frac{\xi^2 \phi}{1-\phi}\boldsymbol{I}\right)^{-1}\boldsymbol{X}^\intercal\right]\boldsymbol{e}_j \\ &= \phi^{-1}\left[\mathbb{1}[i=j] - \boldsymbol{x}_i^\intercal \left(\boldsymbol{X}^\intercal \boldsymbol{X} + \frac{\xi^2 \phi}{1-\phi}\boldsymbol{I}\right)^{-1}\boldsymbol{x}_j\right]. \end{aligned}$$

B2.2 Efficient computation

To compute $\boldsymbol{Q}$, first explicitly compute the d-by-d matrix $(\boldsymbol{X}^\intercal \boldsymbol{X} + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I})$. This can be done with $\mathcal{O}(nd^2)$ operations. We then use $\mathcal{O}(d^3)$ operations to compute its inverse, and another $\mathcal{O}(n^2 d)$ to multiply on the left and right by $\boldsymbol{X}$ and $\boldsymbol{X}^\intercal$. Subtracting the result from the identity and dividing all entries by ϕ requires $\mathcal{O}(n^2)$ operations. Thus, the total run time is $\mathcal{O}(n^2 d)$, and the computation can be performed in $\mathcal{O}(n^2)$ space.

An important insight is that computing the entires of the ridge loss matrix $\boldsymbol{Q}$ is no more computationally intensive than sampling a design from the Gram–Schmidt Walk. In other words, estimating the cross moments will be the main computational bottleneck when constructing the ridge loss estimator $\widehat{L}$, which is used for confidence intervals and variance estimates.

B3 Balanced Gram–Schmidt Walk design

The Balanced Gram–Schmidt Walk design is obtained by applying a modified Gram–Schmidt Walk algorithm to a carefully constructed set of input vectors. As before, the set of $(n+d)$-dimensional vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ are defined as

$$\boldsymbol{b}_i = \begin{bmatrix} \sqrt{\phi}\boldsymbol{e}_i \\ \xi^{-1}\sqrt{1-\phi}\boldsymbol{x}_i \end{bmatrix} \ .$$

The modification which ensures that a balanced assignment is produced is to constrain the step direction to be balanced at each iteration. As before, the step direction $\boldsymbol{u}_t$ is chosen so that $\boldsymbol{u}_t(i) = 0$ for $i \notin \mathcal{A}_t$ and $\boldsymbol{u}_t(p_t) = 1$ for the pivot unit p_t. However, the remaining coordinates are chosen as the solution to the following *constrained* minimization problem,

$$\boldsymbol{u}_t(\mathcal{A}_t \setminus p_t) = \underset{\sum_{i \notin \mathcal{A}_t \setminus p_t} \boldsymbol{u}(i) = -1}{\arg\min} \|\boldsymbol{b}_{p_t} + \sum_{i \notin \mathcal{A}_t \setminus p_t} \boldsymbol{u}(i)\boldsymbol{b}_i\|^2 \ .$$

As before, the solution to this constrained minimization problem may be exactly computed as the solution to a system of linear equations. Furthermore, pre-computing and maintaining similar matrix factorizations allow for the same $O(n^2 d)$ arithmetic operations and $O(n + d^2)$ storage as the Gram–Schmidt Walk. In the remainder of this section, we derive the system of linear equations which define the step direction and describe a fast implementation solving them.

B3.1 Derivation of the step direction

As before, let k be the number of units which are alive and not the pivot, i.e., $k = |\mathcal{A}_t \setminus p_t|$, and let $\boldsymbol{B}_t$ be the $(n+d)$-by-k matrix with columns $\boldsymbol{b}_i$ for $i \in \mathcal{A}_t \setminus p_t$. For notational clarity, we write the coordinates $\boldsymbol{u}(\mathcal{A}_t \setminus p_t)$ simply as $\boldsymbol{u}$ and vectors like $\mathbf{1}$ are assumed to be k-dimensional. The step direction is computed as the minimizer of the constrained minimization

$$\min_{\langle \mathbf{1}, \boldsymbol{u} \rangle = -1} \|\boldsymbol{b}_{p_t} + \boldsymbol{B}_t \boldsymbol{u}\|^2 \ ,$$

which has the Lagrangian form

$$\min_{\substack{\boldsymbol{u} \in \mathbb{R}^k \\ \nu \in \mathbb{R}}} \|\boldsymbol{b}_{p_t} + \boldsymbol{B}_t \boldsymbol{u}\|^2 + \nu(1 + \langle \mathbf{1}, \boldsymbol{u} \rangle) \ .$$

By strong duality, the minimizer of the Lagrangian is the minimizer of the primal and so it suffices to solve for the minima of the Lagrangian (see Chapter 5 of Boyd & Vandenberghe, 2004 for more details). By convexity, this may be done by solving for first order optimality conditions. Computing the gradients of the Lagrangian objective and setting them to 0 yields the system of equations

$$\begin{aligned} \nabla_{\boldsymbol{u}} &= 2\boldsymbol{B}_t^\intercal \boldsymbol{B}\boldsymbol{u} + 2\boldsymbol{B}_t^\intercal \boldsymbol{b}_{p_t} + \nu \mathbf{1} = 0 \\ \nabla_{\nu} &= 1 + \langle \mathbf{1}, \boldsymbol{u} \rangle = 0 \ , \end{aligned}$$

which define the step direction $\boldsymbol{u}$. Multiplying the first equation by $\frac{1}{2}\big(\boldsymbol{B}_t^\intercal \boldsymbol{B}\big)^{-1}$ and rearranging, we obtain the equivalent system of equations

$$\begin{gathered} \boldsymbol{u} = -\big(\boldsymbol{B}_t^\intercal \boldsymbol{B}\big)^{-1} \boldsymbol{B}_t^\intercal \boldsymbol{b}_{p_t} - \nu \tfrac{1}{2}\big(\boldsymbol{B}_t^\intercal \boldsymbol{B}\big)^{-1} \mathbf{1} \\ \langle \mathbf{1}, \boldsymbol{u} \rangle = -1 \ . \end{gathered}$$

Recall that by construction of the input vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_n$ we have that

$$\begin{aligned} \big(\boldsymbol{B}_t^\intercal \boldsymbol{B}\big)^{-1} &= \big(\phi \boldsymbol{I}_k + \xi^{-2}(1-\phi)\boldsymbol{X}_t \boldsymbol{X}_t^\intercal\big)^{-1} = \phi^{-1}\left[\boldsymbol{I}_k - \boldsymbol{X}_t\left(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \frac{\xi^2 \phi}{1-\phi}\boldsymbol{I}_d\right)^{-1} \boldsymbol{X}_t^\intercal\right] \\ \boldsymbol{B}_t^\intercal \boldsymbol{b}_{p_t} &= \xi^{-2}(1-\phi)\boldsymbol{X}_t \boldsymbol{x}_{p_t} \ . \end{aligned}$$

Thus, the step direction may be obtained by soling the system of equations

$$\begin{gathered} \boldsymbol{u} = -\Big(\tfrac{1-\phi}{\xi^2 \phi}\Big)\boldsymbol{X}_t\left[\boldsymbol{x}_{p_t} - \Big(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \tfrac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\Big)^{-1} \boldsymbol{X}_t^\intercal \boldsymbol{X}_t \boldsymbol{x}_{p_t}\right] - \tfrac{1}{2\phi}\left[\mathbf{1} - \boldsymbol{X}_t\Big(\boldsymbol{X}_t^\intercal \boldsymbol{X}_t + \tfrac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\Big)^{-1} \boldsymbol{X}_t^\intercal \mathbf{1}\right] \\ \langle \mathbf{1}, \boldsymbol{u} \rangle = -1 \end{gathered}$$

B3.2 Efficient computation via matrix factorizations

We now demonstrate how this system of equations may be efficiently solved at each iteration of the algorithm. To this end, define the vectors

$$\boldsymbol{a}_t = \left(\frac{1-\phi}{\xi^2\phi}\right)\boldsymbol{X}_t\left[\boldsymbol{x}_{p_t} - \left(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\right)^{-1}\boldsymbol{X}_t^\intercal\boldsymbol{X}_t\boldsymbol{x}_{p_t}\right]$$
$$\boldsymbol{r}_t = \frac{1}{2\phi}\left[\boldsymbol{1} - \boldsymbol{X}_t\left(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \frac{\xi^2\phi}{1-\phi}\boldsymbol{I}_d\right)^{-1}\boldsymbol{X}_t^\intercal\boldsymbol{1}\right] .$$

In Section B1, we discussed that $\boldsymbol{a}_t$ may be computed using $\mathcal{O}(nd)$ arithmetic operations given the Cholesky factorization of $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)$. Now we will show that given the same Cholesky factorization, the vector $\boldsymbol{r}_t$ may also be computed using $\mathcal{O}(nd)$ arithmetic operations. To achieve this, we may compute $\boldsymbol{r}_t$ using the following steps.

1. $\boldsymbol{r}_t^{(1)} = \boldsymbol{X}_t^\intercal\boldsymbol{1} = \sum_{i\in\mathcal{A}_t\setminus p_t}\boldsymbol{x}_i$
2. $\boldsymbol{r}_t^{(2)} = (\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)^{-1}\boldsymbol{I}_d)^{-1}\boldsymbol{r}_t^{(1)}$
3. $\boldsymbol{r}_t^{(3)} = \boldsymbol{X}_t\boldsymbol{r}_t^{(2)}$
4. $\boldsymbol{r}_t = \frac{1}{2\phi}\left[\boldsymbol{1} - \boldsymbol{r}_t^{(3)}\right]$

Note that $\boldsymbol{r}_t^{(1)}$ is the sum of the covariates of units which are alive but not the pivot. Although this may be computed using $\mathcal{O}(nd)$ arithmetic operations per iteration, it is more efficient to maintain this quantity throughout the algorithm. For instance, initially computing a vector of covariate sums $\boldsymbol{r} = \boldsymbol{X}\boldsymbol{1} = \sum_{i=1}^n\boldsymbol{x}_i$ and then updating it as units are frozen or chosen as the pivot requires $\mathcal{O}(nd)$ once to pre-compute and then $O(d)$ arithmetic operations per update. This leads to using $O(nd)$ arithmetic operations for maintaining this sum, which is an improvement over naively computing it at each iteration. Given a Cholesky factorization of $(\boldsymbol{X}_t^\intercal\boldsymbol{X}_t + \xi^2\phi(1-\phi)\boldsymbol{I}_d)$, the vector $\boldsymbol{r}_t^{(2)}$ may be computed in place using $\mathcal{O}(d^2)$ arithmetic operations via a backward-forward solve. The next vector $\boldsymbol{r}_t^{(3)}$ is obtained by multiplying the matrix of covariates $\boldsymbol{X}_t$ by $\boldsymbol{r}_t^{(2)}$, which requires $\mathcal{O}(nd)$ arithmetic operations. Finally, $\boldsymbol{r}_t$ is obtained by using $\mathcal{O}(n)$ arithmetic operations in place.

Lastly, to construct the balanced step direction $\boldsymbol{u}$, we must choose ν to ensure that $\langle\boldsymbol{1},\boldsymbol{u}\rangle = -1$. Choosing $\nu = (1-\langle\boldsymbol{1},\boldsymbol{a}_t\rangle)\langle\boldsymbol{1},\boldsymbol{r}_t\rangle^{-1}$ yields

$$\langle\boldsymbol{1},\boldsymbol{u}\rangle = \langle\boldsymbol{1},-\boldsymbol{a}_t-\nu\boldsymbol{r}_t\rangle = -\langle\boldsymbol{1},\boldsymbol{a}_t\rangle - \nu\langle\boldsymbol{1},\boldsymbol{r}_t\rangle = -\langle\boldsymbol{1},\boldsymbol{a}_t\rangle - (1-\langle\boldsymbol{1},\boldsymbol{a}_t\rangle)\langle\boldsymbol{1},\boldsymbol{r}_t\rangle^{-1}\langle\boldsymbol{1},\boldsymbol{r}_t\rangle = -1,$$

and this choice of ν may be computed using $\mathcal{O}(n)$ arithmetic operations. Finally, the remaining coordinates of the step direction are computed as $\boldsymbol{u}_t(\mathcal{A}_t\setminus p_t) = -(\boldsymbol{a}_t+\nu\boldsymbol{r}_t)$.

Thus, using the technique described above, we may sample from the balanced Gram–Schmidt walk in $\mathcal{O}(n^2 d)$ arithmetic operations and $\mathcal{O}(n + d^2)$ additional space.

Supplement C: Additional simulation results

This supplement presents additional results from the simulation study:

- Tables C1 to C3 present the same information as Table 1 in the main paper for all sample sizes: $n \in \{30, 296, 2960\}$. Table C2 is identical to Table 1, and it is included in this supplement only for reference.
- Tables C4 to C6 present additional balance results for all sample sizes and all designs. Compared to the previous set of tables, these tables also adds balance metrics for raw covariates (denoted "Raw $\lambda_{\boldsymbol{z}}$" and "Raw $\lambda_{\boldsymbol{X}}$") and normalized covariates (denoted "Norm $\lambda_{\boldsymbol{z}}$" and "Norm $\lambda_{\boldsymbol{X}}$"). The columns "Iso $\lambda_{\boldsymbol{z}}$" and "Iso $\lambda_{\boldsymbol{X}}$" correspond to the columns "$\lambda_{\boldsymbol{z}}$" and "$\lambda_{\boldsymbol{X}}$" in the previous table. These tables include versions of the KAK, GSW and BGSW designs that uses normalized rather that decorrelated covariates, which is denoted with "norm".
- Tables C7 to C9 present root mean square errors for three estimators, for all sample sizes, for all designs and for several outcomes. The estimators are the Horvitz–Thompson estimator discussed in the main paper (denoted "HT"), the difference-in-means estimator (denoted "DiM") and Lin's OLS estimator briefly discussed in the main paper (denoted "OLS"). The outcomes are the same as in the main paper, with one addition. The additional outcome, denoted "AB", sets one potential outcome to a normalized copy of Outcome A, and it sets the other potential outcome to a normalized copy of Outcome B. Hence, Outcome AB has non-zero and heterogeneous treatment effects. The root mean square errors are normalized by the root mean square error of the Horvitz–Thompson estimator under the fully randomized design. The OLS estimator is not reported for $n = 30$ because it is highly unstable in that case; the number of covariates is $d = 13$, so the number of regressors in the estimator would be 28.
- Tables C10 to C12 present the bias of the estimators in the previous set of tables. This is reported as the ratio between the squared bias and the mean square error: $(\tau - \mathrm{E}[\widehat{\tau}])^2 / \mathrm{E}[(\tau - \widehat{\tau})^2]$. Hence, this metric is naturally on the unit interval, where zero indicates no bias, and larger values indicate that the bias is a larger share of the estimation error.
- Tables C13 to C18 present coverage probabilities for three confidence intervals, for confidence levels 0.95 and 0.99. This is presented for the Gram–Schmidt Walk designs and two comparisons design. The three confidence intervals are those mentioned in the main paper, namely: intervals based on Chebyshev's inequality using a direct, conservative estimator of the variance; intervals based on the subgaussian bound

using a conservative estimator of the variance bound; and intervals based on a normal approximation using a direct, conservative estimator of the variance.

- Tables C19 to C24 present the average width of the confidence intervals in the previous set of tables. The widths are normalized by the width of the interval based on a normal approximation for the fully randomized design.
- Tables C25 to C27 present the ratio of the bound on the root mean square error given by Theorem 3 in the main paper and the true root mean square error: $\sqrt{4n^{-1}L/\,\mathrm{E}[(\tau-\widehat{\tau})^2]}$. This metric gives an indication of the conservativenss of the bound used in the subgaussian confidence intervals.

Table C1: Robustness, balance and precision, $n = 30$

		Covariate balance			Root mean square error			
Design	$\lambda_{\boldsymbol{z}}$	$\lambda_{\boldsymbol{X}}$	$\boldsymbol{X}$	$\mathbf{1}$	A	B	C	D
Fully random	1.01	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.04	1.03	1.03	0.00	0.95	0.66	1.01	1.02
Matched pairs	2.01	1.97	0.65	0.00	1.01	0.44	1.32	1.02
BJK	30.00	16.24	1.26	0.00	1.00	1.21	1.21	3.07
KAK	4.58	0.54	0.14	0.00	0.83	0.67	2.10	0.39
Rerand 0.50	2.29	0.74	0.42	0.00	0.89	0.67	1.48	0.67
Rerand 0.30	3.57	0.49	0.26	0.00	0.85	0.70	1.85	0.53
Rerand 0.20	4.66	0.35	0.17	0.00	0.81	0.71	2.13	0.46
Rerand 0.15	5.32	0.36	0.13	0.00	0.78	0.73	2.28	0.42
GSW 0.99	1.01	1.00	0.99	0.99	1.00	1.00	1.00	1.00
GSW 0.90	1.07	0.98	0.95	0.94	0.99	0.98	1.03	0.97
GSW 0.50	1.47	0.90	0.67	0.64	0.97	0.91	1.19	0.83
GSW 0.10	2.63	0.65	0.29	0.26	0.90	0.79	1.60	0.57
GSW 0.01	3.25	0.61	0.24	0.20	0.89	0.76	1.78	0.52
BGSW 0.99	1.05	1.03	1.03	0.00	0.95	0.66	1.02	1.02
BGSW 0.90	1.11	1.02	0.98	0.00	0.95	0.66	1.05	0.99
BGSW 0.50	1.53	0.93	0.68	0.00	0.93	0.66	1.22	0.84
BGSW 0.10	2.66	0.69	0.31	0.00	0.87	0.66	1.61	0.59
BGSW 0.01	3.20	0.68	0.27	0.00	0.86	0.64	1.76	0.54

Table C2: Robustness, balance and precision, $n = 296$

		Covariate balance			Root mean square error			
Design	$\lambda_{\boldsymbol{z}}$	$\lambda_{\boldsymbol{X}}$	$\boldsymbol{X}$	$\mathbf{1}$	A	B	C	D
Fully random	1.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.04	1.00	1.00	0.00	0.94	0.71	1.00	1.00
Matched pairs	2.05	0.42	0.18	0.00	0.95	0.54	1.08	0.37
BJK	296.00	11.01	0.85	0.00	1.15	0.14	2.72	0.26
KAK	1.63	0.08	0.01	0.00	0.93	0.55	1.28	0.05
Rerand 0.50	1.26	0.43	0.41	0.00	0.94	0.62	1.12	0.64
Rerand 0.20	1.45	0.17	0.17	0.00	0.93	0.58	1.20	0.41
Rerand 0.15	1.49	0.13	0.13	0.00	0.93	0.57	1.22	0.36
Rerand 0.10	1.54	0.09	0.09	0.00	0.93	0.57	1.24	0.29
GSW 0.99	1.03	0.98	0.98	0.97	1.00	0.99	1.00	0.99
GSW 0.90	1.08	0.81	0.79	0.78	0.99	0.92	1.03	0.89
GSW 0.50	1.30	0.33	0.29	0.29	0.95	0.71	1.14	0.54
GSW 0.10	1.50	0.08	0.05	0.05	0.94	0.58	1.22	0.23
GSW 0.01	1.58	0.03	0.02	0.02	0.93	0.57	1.26	0.14
BGSW 0.99	1.04	0.98	0.98	0.00	0.94	0.70	1.01	0.99
BGSW 0.90	1.08	0.81	0.79	0.00	0.94	0.68	1.04	0.89
BGSW 0.50	1.30	0.33	0.29	0.00	0.94	0.60	1.14	0.54
BGSW 0.10	1.50	0.08	0.05	0.00	0.93	0.56	1.22	0.23
BGSW 0.01	1.58	0.04	0.02	0.00	0.93	0.56	1.26	0.15

Table C3: Robustness, balance and precision, $n = 2960$

		Covariate balance			Root mean square error			
Design	$\lambda_{\boldsymbol{z}}$	$\lambda_{\boldsymbol{X}}$	$\boldsymbol{X}$	$\mathbf{1}$	A	B	C	D
Fully random	1.11	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.11	1.00	1.00	0.00	0.93	0.73	1.00	1.00
Matched pairs	2.16	0.12	0.05	0.00	0.93	0.60	1.07	0.24
BJK	2960.00	9.19	0.71	0.00	1.59	0.15	35.28	0.49
KAK	2.16	0.01	0.00	0.00	0.91	0.61	1.04	0.02
Rerand 0.50	1.11	0.41	0.40	0.00	0.92	0.66	1.00	0.64
Rerand 0.25	1.12	0.21	0.21	0.00	0.91	0.64	1.00	0.46
Rerand 0.15	1.12	0.13	0.13	0.00	0.91	0.63	1.00	0.36
Rerand 0.10	1.12	0.09	0.09	0.00	0.91	0.62	1.02	0.29
GSW 0.99	1.11	0.84	0.84	0.84	0.99	0.95	1.00	0.92
GSW 0.90	1.11	0.33	0.33	0.33	0.94	0.76	1.00	0.57
GSW 0.50	1.12	0.05	0.05	0.05	0.92	0.63	1.00	0.23
GSW 0.10	1.12	0.01	0.01	0.01	0.91	0.61	1.00	0.08
GSW 0.01	1.12	0.00	0.00	0.00	0.91	0.61	1.01	0.04
BGSW 0.99	1.11	0.84	0.84	0.00	0.92	0.72	1.00	0.92
BGSW 0.90	1.11	0.33	0.33	0.00	0.92	0.65	1.00	0.57
BGSW 0.50	1.12	0.05	0.05	0.00	0.91	0.62	1.00	0.23
BGSW 0.10	1.12	0.01	0.01	0.00	0.91	0.61	1.00	0.08
BGSW 0.01	1.12	0.00	0.00	0.00	0.91	0.61	1.00	0.04

Table C4: All balance metrics, $n = 30$

Design	$\lambda_{\boldsymbol{z}}$	Raw $\lambda_{\boldsymbol{X}}$	Norm $\lambda_{\boldsymbol{X}}$	Iso $\lambda_{\boldsymbol{X}}$	Raw $\boldsymbol{X}$	Norm $\boldsymbol{X}$	Iso $\boldsymbol{X}$	$\mathbf{1}$
Fully random	1.01	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.04	0.51	1.03	1.03	0.60	1.03	1.03	0.00
Matched pairs	2.01	0.53	1.17	1.97	0.62	0.57	0.65	0.00
BJK	30.00	0.24	4.42	16.24	0.20	0.80	1.26	0.00
KAK	4.58	0.12	0.31	0.54	0.12	0.15	0.14	0.00
KAK norm	4.55	0.09	0.16	0.72	0.09	0.11	0.20	0.00
Rerand 0.50	2.29	0.27	0.44	0.74	0.32	0.43	0.42	0.00
Rerand 0.30	3.57	0.18	0.29	0.49	0.20	0.26	0.26	0.00
Rerand 0.20	4.66	0.12	0.22	0.35	0.12	0.18	0.17	0.00
Rerand 0.15	5.32	0.10	0.22	0.36	0.10	0.13	0.13	0.00
GSW 0.99	1.01	1.00	0.99	1.00	1.00	0.99	0.99	0.99
GSW 0.90	1.07	0.95	0.94	0.98	0.96	0.95	0.95	0.94
GSW 0.50	1.47	0.72	0.66	0.90	0.74	0.67	0.67	0.64
GSW 0.10	2.63	0.37	0.33	0.65	0.39	0.29	0.29	0.26
GSW 0.01	3.25	0.33	0.31	0.61	0.34	0.24	0.24	0.20
BGSW 0.99	1.05	0.50	1.03	1.03	0.60	1.03	1.03	0.00
BGSW 0.90	1.11	0.48	0.97	1.02	0.58	0.97	0.98	0.00
BGSW 0.50	1.53	0.37	0.67	0.93	0.46	0.68	0.68	0.00
BGSW 0.10	2.66	0.22	0.36	0.69	0.27	0.31	0.31	0.00
BGSW 0.01	3.20	0.21	0.34	0.68	0.25	0.27	0.27	0.00
GSW norm 0.99	1.01	0.99	0.98	1.00	0.99	0.99	0.99	0.99
GSW norm 0.90	1.07	0.93	0.83	1.03	0.94	0.91	0.95	0.94
GSW norm 0.50	1.40	0.63	0.40	1.13	0.68	0.58	0.73	0.63
GSW norm 0.10	2.42	0.32	0.24	0.92	0.37	0.25	0.37	0.25
GSW norm 0.01	3.26	0.28	0.23	0.73	0.32	0.20	0.26	0.19
BGSW norm 0.99	1.05	0.50	1.01	1.03	0.60	1.02	1.03	0.00
BGSW norm 0.90	1.11	0.46	0.85	1.06	0.56	0.94	0.98	0.00
BGSW norm 0.50	1.44	0.30	0.42	1.16	0.41	0.59	0.74	0.00
BGSW norm 0.10	2.44	0.20	0.27	0.94	0.25	0.28	0.39	0.00
BGSW norm 0.01	3.21	0.19	0.27	0.77	0.22	0.23	0.29	0.00

Table C5: All balance metrics, $n = 296$

Design	$\lambda_{\boldsymbol{z}}$	Raw $\lambda_{\boldsymbol{X}}$	Norm $\lambda_{\boldsymbol{X}}$	Iso $\lambda_{\boldsymbol{X}}$	Raw $\boldsymbol{X}$	Norm $\boldsymbol{X}$	Iso $\boldsymbol{X}$	$\mathbf{1}$
Fully random	1.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.04	0.54	1.00	1.00	0.56	1.00	1.00	0.00
Matched pairs	2.05	0.13	0.23	0.42	0.13	0.17	0.18	0.00
BJK	296.00	0.07	5.22	11.01	0.07	0.90	0.85	0.00
KAK	1.63	0.00	0.05	0.08	0.00	0.01	0.01	0.00
KAK norm	1.75	0.00	0.02	0.05	0.00	0.01	0.01	0.00
Rerand 0.50	1.26	0.22	0.41	0.43	0.23	0.41	0.41	0.00
Rerand 0.20	1.45	0.09	0.17	0.17	0.09	0.17	0.17	0.00
Rerand 0.15	1.49	0.07	0.13	0.13	0.07	0.13	0.13	0.00
Rerand 0.10	1.54	0.05	0.09	0.09	0.05	0.09	0.09	0.00
GSW 0.99	1.03	0.97	0.97	0.98	0.98	0.98	0.98	0.97
GSW 0.90	1.08	0.79	0.79	0.81	0.79	0.79	0.79	0.78
GSW 0.50	1.30	0.29	0.29	0.33	0.29	0.29	0.29	0.29
GSW 0.10	1.50	0.05	0.05	0.08	0.05	0.05	0.05	0.05
GSW 0.01	1.58	0.02	0.02	0.03	0.02	0.02	0.02	0.02
BGSW 0.99	1.04	0.53	0.98	0.98	0.54	0.98	0.98	0.00
BGSW 0.90	1.08	0.42	0.79	0.81	0.44	0.79	0.79	0.00
BGSW 0.50	1.30	0.16	0.29	0.33	0.16	0.29	0.29	0.00
BGSW 0.10	1.50	0.03	0.05	0.08	0.03	0.05	0.05	0.00
BGSW 0.01	1.58	0.01	0.02	0.04	0.01	0.02	0.02	0.00
GSW norm 0.99	1.04	0.93	0.90	0.99	0.93	0.94	0.95	0.95
GSW norm 0.90	1.06	0.55	0.44	0.96	0.56	0.60	0.67	0.64
GSW norm 0.50	1.20	0.13	0.08	0.68	0.15	0.16	0.22	0.17
GSW norm 0.10	1.43	0.03	0.02	0.19	0.03	0.03	0.05	0.03
GSW norm 0.01	1.57	0.02	0.01	0.05	0.02	0.02	0.02	0.02
BGSW norm 0.99	1.04	0.49	0.90	1.00	0.51	0.94	0.96	0.00
BGSW norm 0.90	1.07	0.25	0.44	0.96	0.28	0.60	0.67	0.00
BGSW norm 0.50	1.20	0.05	0.08	0.68	0.07	0.16	0.22	0.00
BGSW norm 0.10	1.43	0.01	0.02	0.19	0.02	0.03	0.05	0.00
BGSW norm 0.01	1.57	0.01	0.02	0.05	0.01	0.02	0.02	0.00

Table C6: All balance metrics, $n = 2960$

Design	$\lambda_{\boldsymbol{z}}$	Raw $\lambda_{\boldsymbol{X}}$	Norm $\lambda_{\boldsymbol{X}}$	Iso $\lambda_{\boldsymbol{X}}$	Raw $\boldsymbol{X}$	Norm $\boldsymbol{X}$	Iso $\boldsymbol{X}$	$\mathbf{1}$
Fully random	1.11	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Group-balanced	1.11	0.54	1.00	1.00	0.55	1.00	1.00	0.00
Matched pairs	2.16	0.04	0.07	0.12	0.04	0.04	0.05	0.00
BJK	2960.00	0.32	4.12	9.19	0.31	0.72	0.71	0.00
KAK	2.16	0.00	0.00	0.01	0.00	0.00	0.00	0.00
KAK norm	2.58	0.00	0.00	0.01	0.00	0.00	0.00	0.00
Rerand 0.50	1.11	0.22	0.41	0.41	0.22	0.40	0.40	0.00
Rerand 0.25	1.12	0.11	0.21	0.21	0.12	0.21	0.21	0.00
Rerand 0.15	1.12	0.07	0.13	0.13	0.07	0.13	0.13	0.00
Rerand 0.10	1.12	0.05	0.09	0.09	0.05	0.09	0.09	0.00
GSW 0.99	1.11	0.84	0.84	0.84	0.84	0.84	0.84	0.84
GSW 0.90	1.11	0.33	0.33	0.33	0.33	0.33	0.33	0.33
GSW 0.50	1.12	0.05	0.05	0.05	0.05	0.05	0.05	0.05
GSW 0.10	1.12	0.01	0.01	0.01	0.01	0.01	0.01	0.01
GSW 0.01	1.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.99	1.11	0.45	0.84	0.84	0.46	0.84	0.84	0.00
BGSW 0.90	1.11	0.17	0.32	0.33	0.18	0.32	0.33	0.00
BGSW 0.50	1.12	0.03	0.05	0.05	0.03	0.05	0.05	0.00
BGSW 0.10	1.12	0.00	0.01	0.01	0.00	0.01	0.01	0.00
BGSW 0.01	1.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.99	1.11	0.54	0.44	0.95	0.55	0.60	0.66	0.64
GSW norm 0.90	1.12	0.10	0.07	0.66	0.12	0.13	0.19	0.14
GSW norm 0.50	1.12	0.01	0.01	0.18	0.02	0.02	0.03	0.02
GSW norm 0.10	1.12	0.00	0.00	0.03	0.00	0.00	0.01	0.00
GSW norm 0.01	1.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.99	1.11	0.25	0.43	0.95	0.26	0.59	0.66	0.00
BGSW norm 0.90	1.12	0.04	0.06	0.66	0.05	0.13	0.19	0.00
BGSW norm 0.50	1.12	0.01	0.01	0.18	0.01	0.02	0.03	0.00
BGSW norm 0.10	1.12	0.00	0.00	0.03	0.00	0.00	0.01	0.00
BGSW norm 0.01	1.12	0.00	0.00	0.01	0.00	0.00	0.00	0.00

Table C7: RMSE for all estimators, $n = 30$

	Outcome A		Outcome B		Outcome C		Outcome D		Outcome AB	
Design	HT	DiM	HT	DiM	HT	DiM	HT	DiM	HT	DiM
Fully random	1.00	0.97	1.00	0.67	1.00	1.04	1.00	1.04	1.00	0.78
Group-balanced	0.95	0.95	0.66	0.66	1.01	1.01	1.02	1.02	0.76	0.76
Matched pairs	1.01	1.01	0.44	0.44	1.32	1.32	1.02	1.02	0.64	0.64
BJK	1.00	1.00	1.21	1.21	1.21	1.21	3.07	3.07	1.33	1.33
KAK	0.83	0.83	0.67	0.67	2.10	2.10	0.39	0.39	0.71	0.71
KAK norm	0.89	0.89	0.71	0.71	2.07	2.07	0.66	0.66	0.74	0.74
Rerand 0.50	0.89	0.89	0.67	0.67	1.48	1.48	0.67	0.67	0.73	0.73
Rerand 0.30	0.85	0.85	0.70	0.70	1.85	1.85	0.53	0.53	0.73	0.73
Rerand 0.20	0.81	0.81	0.71	0.71	2.13	2.13	0.46	0.46	0.70	0.70
Rerand 0.15	0.78	0.78	0.73	0.73	2.28	2.28	0.42	0.42	0.68	0.68
GSW 0.99	1.00	0.97	1.00	0.67	1.00	1.04	1.00	1.03	1.00	0.78
GSW 0.90	0.99	0.96	0.98	0.67	1.03	1.06	0.97	1.01	0.99	0.77
GSW 0.50	0.97	0.94	0.91	0.66	1.19	1.22	0.83	0.85	0.93	0.75
GSW 0.10	0.90	0.87	0.79	0.66	1.60	1.61	0.57	0.58	0.82	0.72
GSW 0.01	0.89	0.86	0.76	0.65	1.78	1.79	0.52	0.53	0.80	0.70
BGSW 0.99	0.95	0.95	0.66	0.66	1.02	1.02	1.02	1.02	0.76	0.76
BGSW 0.90	0.95	0.95	0.66	0.66	1.05	1.05	0.99	0.99	0.75	0.75
BGSW 0.50	0.93	0.93	0.66	0.66	1.22	1.22	0.84	0.84	0.74	0.74
BGSW 0.10	0.87	0.87	0.66	0.66	1.61	1.61	0.59	0.59	0.71	0.71
BGSW 0.01	0.86	0.86	0.64	0.64	1.76	1.76	0.54	0.54	0.70	0.70
GSW norm 0.99	1.00	0.97	1.00	0.67	1.00	1.04	1.00	1.04	1.00	0.78
GSW norm 0.90	1.00	0.97	0.98	0.66	1.03	1.06	1.00	1.03	0.99	0.77
GSW norm 0.50	0.99	0.96	0.87	0.62	1.17	1.19	0.95	0.97	0.92	0.75
GSW norm 0.10	0.94	0.91	0.76	0.64	1.53	1.54	0.73	0.74	0.83	0.73
GSW norm 0.01	0.90	0.88	0.74	0.63	1.78	1.79	0.61	0.61	0.80	0.70
BGSW norm 0.99	0.95	0.95	0.66	0.66	1.02	1.02	1.02	1.02	0.76	0.76
BGSW norm 0.90	0.95	0.95	0.65	0.65	1.05	1.05	1.01	1.01	0.75	0.75
BGSW norm 0.50	0.95	0.95	0.62	0.62	1.18	1.18	0.96	0.96	0.74	0.74
BGSW norm 0.10	0.91	0.91	0.64	0.64	1.54	1.54	0.74	0.74	0.73	0.73
BGSW norm 0.01	0.88	0.88	0.63	0.63	1.76	1.76	0.62	0.62	0.70	0.70

Table C8: RMSE for all estimators, $n = 296$

	Outcome A			Outcome B			Outcome C			Outcome D			Outcome AB		
Design	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS
Fully random	1.00	0.94	4.43	1.00	0.71	3.05	1.00	1.00	2.35	1.00	1.00	0.69	1.00	0.79	3.87
Group-balanced	0.94	0.94	4.43	0.71	0.71	3.04	1.00	1.00	2.34	1.00	1.00	0.64	0.79	0.79	3.86
Matched pairs	0.95	0.95	4.46	0.54	0.54	2.86	1.08	1.08	1.97	0.37	0.37	0.00	0.69	0.69	3.84
BJK	1.15	1.15	4.93	0.14	0.14	0.99	2.72	2.72	1.43	0.26	0.26	0.00	0.60	0.60	3.31
KAK	0.93	0.93	4.37	0.55	0.55	3.02	1.28	1.28	2.36	0.05	0.05	0.00	0.70	0.70	3.84
KAK norm	0.93	0.93	4.37	0.56	0.56	3.02	1.28	1.28	2.35	0.12	0.12	0.00	0.70	0.70	3.84
Rerand 0.50	0.94	0.94	4.40	0.62	0.62	3.03	1.12	1.12	2.33	0.64	0.64	0.00	0.74	0.74	3.84
Rerand 0.20	0.93	0.93	4.39	0.58	0.58	3.02	1.20	1.20	2.33	0.41	0.41	0.00	0.72	0.72	3.84
Rerand 0.15	0.93	0.93	4.38	0.57	0.57	3.02	1.22	1.22	2.33	0.36	0.36	0.00	0.71	0.71	3.84
Rerand 0.10	0.93	0.93	4.38	0.57	0.57	3.02	1.24	1.24	2.33	0.29	0.29	0.00	0.71	0.71	3.84
GSW 0.99	1.00	0.94	4.44	0.99	0.70	3.05	1.00	1.01	2.35	0.99	0.99	0.64	0.99	0.79	3.87
GSW 0.90	0.99	0.94	4.42	0.92	0.68	3.05	1.03	1.04	2.34	0.89	0.89	0.32	0.94	0.77	3.86
GSW 0.50	0.95	0.93	4.39	0.71	0.60	3.02	1.14	1.14	2.32	0.54	0.54	0.00	0.80	0.73	3.85
GSW 0.10	0.94	0.93	4.38	0.58	0.56	3.02	1.22	1.22	2.32	0.23	0.23	0.00	0.72	0.71	3.84
GSW 0.01	0.93	0.93	4.38	0.57	0.56	3.01	1.26	1.26	2.32	0.14	0.14	0.00	0.71	0.70	3.83
BGSW 0.99	0.94	0.94	4.42	0.70	0.70	3.04	1.01	1.01	2.34	0.99	0.99	0.59	0.78	0.78	3.86
BGSW 0.90	0.94	0.94	4.41	0.68	0.68	3.04	1.04	1.04	2.33	0.89	0.89	0.24	0.77	0.77	3.85
BGSW 0.50	0.94	0.94	4.39	0.60	0.60	3.02	1.14	1.14	2.32	0.54	0.54	0.00	0.73	0.73	3.84
BGSW 0.10	0.93	0.93	4.38	0.56	0.56	3.02	1.22	1.22	2.32	0.23	0.23	0.00	0.71	0.71	3.84
BGSW 0.01	0.93	0.93	4.38	0.56	0.56	3.02	1.26	1.26	2.33	0.15	0.15	0.00	0.70	0.70	3.84
GSW norm 0.99	1.00	0.94	4.43	0.98	0.70	3.05	1.00	1.01	2.35	0.98	0.99	0.61	0.98	0.78	3.87
GSW norm 0.90	0.97	0.93	4.41	0.84	0.63	3.04	1.03	1.03	2.32	0.85	0.85	0.13	0.89	0.74	3.86
GSW norm 0.50	0.94	0.93	4.38	0.64	0.57	3.02	1.09	1.09	2.29	0.50	0.50	0.00	0.75	0.71	3.84
GSW norm 0.10	0.93	0.93	4.38	0.57	0.56	3.02	1.19	1.19	2.31	0.22	0.22	0.00	0.71	0.70	3.84
GSW norm 0.01	0.93	0.93	4.38	0.56	0.55	3.02	1.25	1.25	2.33	0.15	0.15	0.00	0.71	0.70	3.84
BGSW norm 0.99	0.94	0.94	4.43	0.69	0.69	3.04	1.01	1.01	2.34	0.98	0.98	0.54	0.78	0.78	3.86
BGSW norm 0.90	0.93	0.93	4.40	0.62	0.62	3.03	1.03	1.03	2.32	0.85	0.85	0.08	0.74	0.74	3.85
BGSW norm 0.50	0.93	0.93	4.38	0.57	0.57	3.02	1.09	1.09	2.29	0.49	0.49	0.00	0.71	0.71	3.84
BGSW norm 0.10	0.93	0.93	4.38	0.56	0.56	3.02	1.19	1.19	2.31	0.23	0.23	0.00	0.70	0.70	3.84
BGSW norm 0.01	0.93	0.93	4.37	0.56	0.56	3.02	1.25	1.25	2.33	0.16	0.16	0.00	0.70	0.70	3.84

Table C9: RMSE for all estimators, $n = 2960$

	Outcome A			Outcome B			Outcome C			Outcome D			Outcome AB		
Design	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS
Fully random	1.00	0.93	3.96	1.00	0.73	2.74	1.00	1.00	4.26	1.00	1.00	0.00	1.00	0.78	5.92
Group-balanced	0.93	0.93	3.96	0.73	0.73	2.73	1.00	1.00	4.27	1.00	1.00	0.00	0.77	0.77	5.92
Matched pairs	0.93	0.93	3.99	0.60	0.60	2.71	1.07	1.07	4.43	0.24	0.24	0.00	0.69	0.69	5.89
BJK	1.59	1.59	0.42	0.15	0.15	3.24	35.28	35.28	34.82	0.49	0.49	0.00	0.86	0.86	5.48
KAK	0.91	0.91	3.96	0.61	0.61	2.73	1.04	1.04	4.26	0.02	0.02	0.00	0.69	0.69	5.92
KAK norm	0.91	0.91	3.96	0.61	0.61	2.74	1.01	1.01	4.27	0.04	0.04	0.00	0.69	0.69	5.94
Rerand 0.50	0.92	0.92	3.96	0.66	0.66	2.74	1.00	1.00	4.26	0.64	0.64	0.00	0.73	0.73	5.92
Rerand 0.25	0.91	0.91	3.96	0.64	0.64	2.74	1.00	1.00	4.25	0.46	0.46	0.00	0.71	0.71	5.91
Rerand 0.15	0.91	0.91	3.95	0.63	0.63	2.73	1.00	1.00	4.26	0.36	0.36	0.00	0.70	0.70	5.91
Rerand 0.10	0.91	0.91	3.95	0.62	0.62	2.74	1.02	1.02	4.26	0.29	0.29	0.00	0.70	0.70	5.91
GSW 0.99	0.99	0.92	3.96	0.95	0.72	2.74	1.00	1.00	4.26	0.92	0.92	0.00	0.96	0.76	5.92
GSW 0.90	0.94	0.92	3.95	0.76	0.65	2.73	1.00	1.00	4.26	0.57	0.57	0.00	0.80	0.72	5.91
GSW 0.50	0.92	0.91	3.95	0.63	0.62	2.74	1.00	1.00	4.25	0.23	0.23	0.00	0.71	0.70	5.91
GSW 0.10	0.91	0.91	3.95	0.61	0.61	2.74	1.00	1.00	4.25	0.08	0.08	0.00	0.69	0.69	5.91
GSW 0.01	0.91	0.91	3.95	0.61	0.61	2.74	1.01	1.01	4.25	0.04	0.04	0.00	0.69	0.69	5.92
BGSW 0.99	0.92	0.92	3.96	0.72	0.72	2.74	1.00	1.00	4.26	0.92	0.92	0.00	0.76	0.76	5.91
BGSW 0.90	0.92	0.92	3.96	0.65	0.65	2.74	1.00	1.00	4.25	0.57	0.57	0.00	0.72	0.72	5.92
BGSW 0.50	0.91	0.91	3.95	0.62	0.62	2.74	1.00	1.00	4.25	0.23	0.23	0.00	0.69	0.69	5.91
BGSW 0.10	0.91	0.91	3.95	0.61	0.61	2.74	1.00	1.00	4.25	0.08	0.08	0.00	0.69	0.69	5.91
BGSW 0.01	0.91	0.91	3.95	0.61	0.61	2.73	1.00	1.00	4.25	0.04	0.04	0.00	0.69	0.69	5.91
GSW norm 0.99	0.97	0.92	3.96	0.86	0.67	2.74	1.00	1.00	4.26	0.84	0.84	0.00	0.89	0.73	5.92
GSW norm 0.90	0.92	0.91	3.95	0.67	0.62	2.73	1.00	1.00	4.25	0.45	0.45	0.00	0.73	0.70	5.92
GSW norm 0.50	0.91	0.91	3.95	0.62	0.61	2.74	1.00	1.00	4.26	0.18	0.18	0.00	0.69	0.69	5.91
GSW norm 0.10	0.91	0.91	3.95	0.61	0.61	2.73	1.00	1.00	4.25	0.07	0.07	0.00	0.69	0.69	5.92
GSW norm 0.01	0.91	0.91	3.95	0.61	0.61	2.74	1.00	1.00	4.25	0.04	0.04	0.00	0.69	0.69	5.91
BGSW norm 0.99	0.92	0.92	3.96	0.67	0.67	2.73	1.00	1.00	4.26	0.84	0.84	0.00	0.73	0.73	5.92
BGSW norm 0.90	0.91	0.91	3.95	0.62	0.62	2.73	1.00	1.00	4.25	0.45	0.45	0.00	0.70	0.70	5.91
BGSW norm 0.50	0.91	0.91	3.95	0.61	0.61	2.73	1.00	1.00	4.25	0.18	0.18	0.00	0.69	0.69	5.91
BGSW norm 0.10	0.91	0.91	3.95	0.61	0.61	2.74	1.00	1.00	4.26	0.07	0.07	0.00	0.69	0.69	5.91
BGSW norm 0.01	0.91	0.91	3.95	0.61	0.61	2.73	1.00	1.00	4.25	0.04	0.04	0.00	0.69	0.69	5.91

Table C10: Bias for all estimators, $n = 30$

	Outcome A		Outcome B		Outcome C		Outcome D		Outcome AB	
Design	HT	DiM	HT	DiM	HT	DiM	HT	DiM	HT	DiM
Fully random	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Group-balanced	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Matched pairs	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BJK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
KAK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
KAK norm	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Rerand 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Rerand 0.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Rerand 0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Rerand 0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BGSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table C11: Bias for all estimators, $n = 296$

	Outcome A			Outcome B			Outcome C			Outcome D			Outcome AB		
Design	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS
Fully random	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
Group-balanced	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
Matched pairs	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.23
BJK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.51
KAK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
KAK norm	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
Rerand 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
Rerand 0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
Rerand 0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
Rerand 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
GSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
GSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
BGSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
BGSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
GSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
GSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.21
BGSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
BGSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22

Table C12: Bias for all estimators, $n = 2960$

	Outcome A			Outcome B			Outcome C			Outcome D			Outcome AB		
Design	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS	HT	DiM	OLS
Fully random	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
Group-balanced	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
Matched pairs	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BJK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.91
KAK	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
KAK norm	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97	0.97	0.00	0.00	0.00	0.74
Rerand 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
Rerand 0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
Rerand 0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
Rerand 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
GSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW norm 0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW norm 0.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW norm 0.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW norm 0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74
BGSW norm 0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74

Table C13: Coverage of confidence intervals: $n = 30$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		0.876	1.000		0.952	1.000		0.976
Group-balanced	1.000		0.900	1.000		0.951	1.000		0.986
GSW 0.99	1.000	1.000	0.875	1.000	0.996	0.952	1.000	1.000	0.976
GSW 0.90	1.000	1.000	0.878	1.000	0.997	0.952	1.000	1.000	0.977
GSW 0.50	1.000	1.000	0.890	1.000	0.999	0.950	1.000	1.000	0.978
GSW 0.10	1.000	1.000	0.921	1.000	1.000	0.934	1.000	1.000	0.976
GSW 0.01	1.000	1.000	0.927	1.000	1.000	0.923	1.000	1.000	0.973
BGSW 0.99	1.000	1.000	0.901	1.000	1.000	0.951	1.000	1.000	0.986
BGSW 0.90	1.000	1.000	0.902	1.000	1.000	0.951	1.000	1.000	0.986
BGSW 0.50	1.000	1.000	0.911	1.000	1.000	0.950	1.000	1.000	0.986
BGSW 0.10	1.000	1.000	0.933	1.000	1.000	0.931	1.000	1.000	0.981
BGSW 0.01	1.000	1.000	0.938	1.000	1.000	0.920	1.000	1.000	0.978
GSW norm 0.99	1.000	1.000	0.875	1.000	0.996	0.952	1.000	1.000	0.976
GSW norm 0.90	1.000	1.000	0.875	1.000	0.997	0.952	1.000	1.000	0.976
GSW norm 0.50	1.000	1.000	0.880	1.000	1.000	0.950	1.000	1.000	0.977
GSW norm 0.10	1.000	1.000	0.904	1.000	1.000	0.942	1.000	1.000	0.979
GSW norm 0.01	1.000	1.000	0.921	1.000	1.000	0.926	1.000	1.000	0.974
BGSW norm 0.99	1.000	1.000	0.900	1.000	1.000	0.951	1.000	1.000	0.986
BGSW norm 0.90	1.000	1.000	0.899	1.000	1.000	0.951	1.000	1.000	0.985
BGSW norm 0.50	1.000	1.000	0.900	1.000	1.000	0.950	1.000	1.000	0.985
BGSW norm 0.10	1.000	1.000	0.918	1.000	1.000	0.942	1.000	1.000	0.984
BGSW norm 0.01	1.000	1.000	0.932	1.000	1.000	0.924	1.000	1.000	0.979

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C14: Coverage of confidence intervals: $n = 30$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		1.000	1.000		0.993	1.000		1.000
Group-balanced	1.000		1.000	1.000		0.992	1.000		1.000
GSW 0.99	1.000	1.000	1.000	1.000	1.000	0.993	1.000	1.000	1.000
GSW 0.90	1.000	1.000	1.000	1.000	1.000	0.993	1.000	1.000	1.000
GSW 0.50	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
GSW 0.10	1.000	1.000	1.000	1.000	1.000	0.984	1.000	1.000	0.998
GSW 0.01	1.000	1.000	1.000	1.000	1.000	0.975	1.000	1.000	0.995
BGSW 0.99	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW 0.90	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW 0.50	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW 0.10	1.000	1.000	1.000	1.000	1.000	0.981	1.000	1.000	0.999
BGSW 0.01	1.000	1.000	1.000	1.000	1.000	0.971	1.000	1.000	0.997
GSW norm 0.99	1.000	1.000	1.000	1.000	1.000	0.993	1.000	1.000	1.000
GSW norm 0.90	1.000	1.000	1.000	1.000	1.000	0.993	1.000	1.000	1.000
GSW norm 0.50	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	0.999
GSW norm 0.10	1.000	1.000	1.000	1.000	1.000	0.989	1.000	1.000	0.999
GSW norm 0.01	1.000	1.000	1.000	1.000	1.000	0.979	1.000	1.000	0.996
BGSW norm 0.99	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW norm 0.90	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW norm 0.50	1.000	1.000	1.000	1.000	1.000	0.992	1.000	1.000	1.000
BGSW norm 0.10	1.000	1.000	1.000	1.000	1.000	0.988	1.000	1.000	1.000
BGSW norm 0.01	1.000	1.000	1.000	1.000	1.000	0.975	1.000	1.000	0.997

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C15: Coverage of confidence intervals: $n = 296$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		0.953	1.000		0.950	1.000		0.978
Group-balanced	1.000		0.966	1.000		0.951	1.000		0.987
GSW 0.99	1.000	0.992	0.953	1.000	0.994	0.950	1.000	0.999	0.979
GSW 0.90	1.000	0.997	0.957	1.000	0.996	0.950	1.000	0.999	0.981
GSW 0.50	1.000	1.000	0.963	1.000	1.000	0.950	1.000	1.000	0.986
GSW 0.10	1.000	1.000	0.966	1.000	1.000	0.950	1.000	1.000	0.990
GSW 0.01	1.000	1.000	0.966	1.000	1.000	0.950	1.000	1.000	0.990
BGSW 0.99	1.000	0.996	0.966	1.000	1.000	0.950	1.000	1.000	0.987
BGSW 0.90	1.000	0.999	0.966	1.000	1.000	0.950	1.000	1.000	0.988
BGSW 0.50	1.000	1.000	0.966	1.000	1.000	0.950	1.000	1.000	0.989
BGSW 0.10	1.000	1.000	0.965	1.000	1.000	0.950	1.000	1.000	0.990
BGSW 0.01	1.000	1.000	0.965	1.000	1.000	0.950	1.000	1.000	0.990
GSW norm 0.99	1.000	0.993	0.954	1.000	0.994	0.950	1.000	0.999	0.979
GSW norm 0.90	1.000	0.994	0.960	1.000	0.996	0.950	1.000	0.999	0.983
GSW norm 0.50	1.000	1.000	0.966	1.000	1.000	0.950	1.000	1.000	0.988
GSW norm 0.10	1.000	1.000	0.965	1.000	1.000	0.950	1.000	1.000	0.990
GSW norm 0.01	1.000	1.000	0.965	1.000	1.000	0.950	1.000	1.000	0.990
BGSW norm 0.99	1.000	0.996	0.967	1.000	1.000	0.950	1.000	1.000	0.987
BGSW norm 0.90	1.000	0.999	0.965	1.000	1.000	0.950	1.000	1.000	0.989
BGSW norm 0.50	1.000	1.000	0.960	1.000	1.000	0.950	1.000	1.000	0.990
BGSW norm 0.10	1.000	1.000	0.964	1.000	1.000	0.950	1.000	1.000	0.990
BGSW norm 0.01	1.000	1.000	0.964	1.000	1.000	0.950	1.000	1.000	0.990

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C16: Coverage of confidence intervals: $n = 296$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		0.992	1.000		0.991	1.000		0.997
Group-balanced	1.000		0.987	1.000		0.991	1.000		0.999
GSW 0.99	1.000	0.999	0.992	1.000	0.999	0.990	1.000	1.000	0.997
GSW 0.90	1.000	0.999	0.993	1.000	0.999	0.990	1.000	1.000	0.998
GSW 0.50	1.000	1.000	0.986	1.000	1.000	0.990	1.000	1.000	0.999
GSW 0.10	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
GSW 0.01	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW 0.99	1.000	1.000	0.987	1.000	1.000	0.991	1.000	1.000	0.999
BGSW 0.90	1.000	1.000	0.987	1.000	1.000	0.991	1.000	1.000	0.999
BGSW 0.50	1.000	1.000	0.987	1.000	1.000	0.990	1.000	1.000	0.999
BGSW 0.10	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW 0.01	1.000	1.000	0.988	1.000	1.000	0.991	1.000	1.000	0.999
GSW norm 0.99	1.000	0.999	0.993	1.000	0.999	0.991	1.000	1.000	0.997
GSW norm 0.90	1.000	0.999	0.994	1.000	0.999	0.990	1.000	1.000	0.998
GSW norm 0.50	1.000	1.000	0.987	1.000	1.000	0.990	1.000	1.000	0.999
GSW norm 0.10	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
GSW norm 0.01	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW norm 0.99	1.000	1.000	0.987	1.000	1.000	0.991	1.000	1.000	0.999
BGSW norm 0.90	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW norm 0.50	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW norm 0.10	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999
BGSW norm 0.01	1.000	1.000	0.988	1.000	1.000	0.990	1.000	1.000	0.999

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C17: Coverage of confidence intervals: $n = 2960$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		0.950	1.000		0.950	1.000		0.981
Group-balanced	1.000		0.951	1.000		0.950	1.000		0.990
GSW 0.99	1.000	0.994	0.948	1.000	0.994	0.950	1.000	0.999	0.982
GSW 0.90	1.000	0.996	0.948	1.000	0.996	0.950	1.000	1.000	0.988
GSW 0.50	1.000	1.000	0.954	1.000	1.000	0.950	1.000	1.000	0.992
GSW 0.10	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
GSW 0.01	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
BGSW 0.99	1.000	0.997	0.952	1.000	1.000	0.950	1.000	1.000	0.990
BGSW 0.90	1.000	0.998	0.954	1.000	1.000	0.950	1.000	1.000	0.992
BGSW 0.50	1.000	1.000	0.953	1.000	1.000	0.950	1.000	1.000	0.993
BGSW 0.10	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
BGSW 0.01	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
GSW norm 0.99	1.000	0.993	0.953	1.000	0.994	0.950	1.000	0.999	0.985
GSW norm 0.90	1.000	0.995	0.952	1.000	0.996	0.950	1.000	1.000	0.991
GSW norm 0.50	1.000	1.000	0.953	1.000	1.000	0.950	1.000	1.000	0.993
GSW norm 0.10	1.000	1.000	0.953	1.000	1.000	0.950	1.000	1.000	0.993
GSW norm 0.01	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
BGSW norm 0.99	1.000	0.997	0.953	1.000	1.000	0.950	1.000	1.000	0.991
BGSW norm 0.90	1.000	0.998	0.953	1.000	1.000	0.950	1.000	1.000	0.993
BGSW norm 0.50	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993
BGSW norm 0.10	1.000	1.000	0.953	1.000	1.000	0.950	1.000	1.000	0.993
BGSW norm 0.01	1.000	1.000	0.952	1.000	1.000	0.950	1.000	1.000	0.993

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C18: Coverage of confidence intervals: $n = 2960$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	1.000		0.990	1.000		0.990	1.000		0.998
Group-balanced	1.000		0.990	1.000		0.990	1.000		0.999
GSW 0.99	1.000	0.999	0.990	1.000	0.999	0.990	1.000	1.000	0.998
GSW 0.90	1.000	0.999	0.989	1.000	0.999	0.990	1.000	1.000	0.999
GSW 0.50	1.000	1.000	0.991	1.000	1.000	0.990	1.000	1.000	0.999
GSW 0.10	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
GSW 0.01	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW 0.99	1.000	1.000	0.991	1.000	1.000	0.990	1.000	1.000	0.999
BGSW 0.90	1.000	1.000	0.990	1.000	1.000	0.990	1.000	1.000	1.000
BGSW 0.50	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW 0.10	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW 0.01	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
GSW norm 0.99	1.000	0.999	0.990	1.000	0.999	0.990	1.000	1.000	0.999
GSW norm 0.90	1.000	0.999	0.991	1.000	0.999	0.990	1.000	1.000	0.999
GSW norm 0.50	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
GSW norm 0.10	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
GSW norm 0.01	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW norm 0.99	1.000	1.000	0.991	1.000	1.000	0.990	1.000	1.000	0.999
BGSW norm 0.90	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW norm 0.50	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW norm 0.10	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000
BGSW norm 0.01	1.000	1.000	0.989	1.000	1.000	0.990	1.000	1.000	1.000

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C19: Relative width of confidence intervals: $n = 30$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	2.282		1.000	2.282		1.000	2.282		1.000
Group-balanced	2.168		0.950	1.510		0.662	1.915		0.839
GSW 0.99	2.282	1.388	1.000	2.279	1.387	0.999	2.281	1.387	1.000
GSW 0.90	2.267	1.407	0.994	2.249	1.395	0.986	2.263	1.404	0.992
GSW 0.50	2.213	1.581	0.970	2.070	1.483	0.907	2.172	1.554	0.952
GSW 0.10	2.055	2.735	0.901	1.804	2.205	0.791	2.006	2.589	0.879
GSW 0.01	2.022	7.996	0.886	1.737	5.929	0.761	1.962	7.426	0.860
BGSW 0.99	2.166	1.389	0.949	1.510	1.391	0.662	1.914	1.389	0.839
BGSW 0.90	2.161	1.416	0.947	1.506	1.441	0.660	1.911	1.428	0.838
BGSW 0.50	2.111	1.653	0.925	1.498	1.828	0.657	1.893	1.742	0.830
BGSW 0.10	1.988	3.109	0.871	1.495	3.860	0.655	1.848	3.499	0.810
BGSW 0.01	1.967	9.388	0.862	1.464	12.047	0.642	1.824	10.772	0.800
GSW norm 0.99	2.281	1.389	1.000	2.275	1.386	0.997	2.279	1.387	0.999
GSW norm 0.90	2.283	1.420	1.001	2.230	1.387	0.977	2.261	1.406	0.991
GSW norm 0.50	2.258	1.667	0.990	1.987	1.470	0.871	2.156	1.592	0.945
GSW norm 0.10	2.143	2.911	0.939	1.745	2.231	0.765	2.022	2.695	0.886
GSW norm 0.01	2.057	8.084	0.902	1.695	5.954	0.743	1.962	7.484	0.860
BGSW norm 0.99	2.168	1.390	0.950	1.507	1.390	0.661	1.913	1.389	0.839
BGSW norm 0.90	2.175	1.429	0.953	1.479	1.431	0.648	1.909	1.430	0.837
BGSW norm 0.50	2.171	1.736	0.951	1.413	1.812	0.619	1.891	1.774	0.829
BGSW norm 0.10	2.076	3.260	0.910	1.452	3.870	0.636	1.873	3.574	0.821
BGSW norm 0.01	2.000	9.461	0.877	1.438	12.055	0.630	1.832	10.809	0.803

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C20: Relative width of confidence intervals: $n = 30$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	3.882		1.000	3.882		1.000	3.882		1.000
Group-balanced	3.689		0.950	2.569		0.662	3.258		0.839
GSW 0.99	3.883	1.266	1.000	3.877	1.264	0.999	3.881	1.265	1.000
GSW 0.90	3.858	1.283	0.994	3.826	1.272	0.986	3.850	1.280	0.992
GSW 0.50	3.765	1.442	0.970	3.522	1.353	0.907	3.695	1.417	0.952
GSW 0.10	3.497	2.494	0.901	3.070	2.011	0.791	3.413	2.361	0.879
GSW 0.01	3.441	7.292	0.886	2.955	5.407	0.761	3.338	6.772	0.860
BGSW 0.99	3.686	1.266	0.949	2.569	1.268	0.662	3.257	1.267	0.839
BGSW 0.90	3.677	1.291	0.947	2.563	1.314	0.660	3.252	1.302	0.838
BGSW 0.50	3.592	1.508	0.925	2.549	1.667	0.657	3.221	1.588	0.830
BGSW 0.10	3.382	2.835	0.871	2.544	3.520	0.655	3.145	3.191	0.810
BGSW 0.01	3.347	8.561	0.862	2.491	10.985	0.642	3.104	9.824	0.800
GSW norm 0.99	3.881	1.267	1.000	3.870	1.264	0.997	3.877	1.265	0.999
GSW norm 0.90	3.884	1.295	1.001	3.794	1.264	0.977	3.848	1.282	0.991
GSW norm 0.50	3.843	1.520	0.990	3.382	1.340	0.871	3.668	1.452	0.945
GSW norm 0.10	3.646	2.654	0.939	2.969	2.034	0.765	3.439	2.457	0.886
GSW norm 0.01	3.500	7.372	0.902	2.884	5.429	0.743	3.339	6.824	0.860
BGSW norm 0.99	3.688	1.267	0.950	2.565	1.267	0.661	3.256	1.267	0.839
BGSW norm 0.90	3.700	1.303	0.953	2.516	1.305	0.648	3.248	1.304	0.837
BGSW norm 0.50	3.693	1.583	0.951	2.404	1.652	0.619	3.218	1.617	0.829
BGSW norm 0.10	3.533	2.973	0.910	2.470	3.529	0.636	3.187	3.259	0.821
BGSW norm 0.01	3.403	8.627	0.877	2.447	10.993	0.630	3.117	9.857	0.803

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C21: Relative width of confidence intervals: $n = 296$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	2.282		1.000	2.282		1.000	2.282		1.000
Group-balanced	2.144		0.940	1.615		0.708	1.941		0.851
GSW 0.99	2.279	1.390	0.999	2.261	1.380	0.991	2.271	1.386	0.995
GSW 0.90	2.248	1.431	0.985	2.102	1.339	0.921	2.187	1.393	0.959
GSW 0.50	2.172	1.830	0.952	1.624	1.368	0.712	1.961	1.652	0.859
GSW 0.10	2.133	3.992	0.935	1.331	2.475	0.584	1.840	3.438	0.807
GSW 0.01	2.130	12.575	0.933	1.290	7.495	0.565	1.824	10.738	0.799
BGSW 0.99	2.142	1.392	0.939	1.608	1.389	0.705	1.938	1.391	0.849
BGSW 0.90	2.141	1.451	0.938	1.546	1.426	0.678	1.915	1.440	0.839
BGSW 0.50	2.135	1.919	0.936	1.374	1.805	0.602	1.854	1.871	0.812
BGSW 0.10	2.129	4.260	0.933	1.283	3.918	0.562	1.823	4.117	0.799
BGSW 0.01	2.128	13.455	0.933	1.269	12.329	0.556	1.817	12.984	0.796
GSW norm 0.99	2.274	1.385	0.997	2.233	1.360	0.979	2.256	1.375	0.989
GSW norm 0.90	2.217	1.408	0.971	1.924	1.221	0.843	2.101	1.334	0.921
GSW norm 0.50	2.143	1.806	0.939	1.454	1.220	0.637	1.887	1.587	0.827
GSW norm 0.10	2.128	3.983	0.933	1.302	2.410	0.571	1.828	3.412	0.801
GSW norm 0.01	2.130	12.572	0.934	1.286	7.475	0.564	1.823	10.730	0.799
BGSW norm 0.99	2.139	1.390	0.937	1.581	1.377	0.693	1.927	1.385	0.845
BGSW norm 0.90	2.129	1.441	0.933	1.426	1.373	0.625	1.869	1.413	0.819
BGSW norm 0.50	2.120	1.909	0.929	1.294	1.762	0.567	1.822	1.849	0.799
BGSW norm 0.10	2.126	4.256	0.932	1.268	3.902	0.556	1.816	4.108	0.796
BGSW norm 0.01	2.126	13.454	0.932	1.268	12.324	0.556	1.816	12.981	0.796

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C22: Relative width of confidence intervals: $n = 296$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	3.882		1.000	3.882		1.000	3.882		1.000
Group-balanced	3.648		0.940	2.748		0.708	3.303		0.851
GSW 0.99	3.878	1.267	0.999	3.848	1.259	0.991	3.864	1.263	0.995
GSW 0.90	3.825	1.305	0.985	3.577	1.221	0.921	3.721	1.271	0.959
GSW 0.50	3.696	1.669	0.952	2.763	1.247	0.712	3.336	1.506	0.859
GSW 0.10	3.630	3.641	0.935	2.265	2.257	0.584	3.131	3.135	0.807
GSW 0.01	3.624	11.467	0.933	2.195	6.835	0.565	3.103	9.792	0.799
BGSW 0.99	3.645	1.269	0.939	2.736	1.267	0.705	3.298	1.268	0.849
BGSW 0.90	3.643	1.323	0.938	2.630	1.300	0.678	3.259	1.313	0.839
BGSW 0.50	3.633	1.750	0.936	2.338	1.646	0.602	3.154	1.706	0.812
BGSW 0.10	3.622	3.884	0.933	2.183	3.572	0.562	3.101	3.754	0.799
BGSW 0.01	3.620	12.270	0.933	2.158	11.243	0.556	3.092	11.840	0.796
GSW norm 0.99	3.869	1.263	0.997	3.800	1.240	0.979	3.839	1.254	0.989
GSW norm 0.90	3.771	1.284	0.971	3.274	1.114	0.843	3.574	1.216	0.921
GSW norm 0.50	3.645	1.646	0.939	2.475	1.113	0.637	3.211	1.447	0.827
GSW norm 0.10	3.621	3.632	0.933	2.216	2.198	0.571	3.110	3.112	0.801
GSW norm 0.01	3.624	11.464	0.934	2.189	6.817	0.564	3.102	9.785	0.799
BGSW norm 0.99	3.639	1.267	0.937	2.690	1.256	0.693	3.279	1.263	0.845
BGSW norm 0.90	3.622	1.314	0.933	2.426	1.252	0.625	3.180	1.288	0.819
BGSW norm 0.50	3.607	1.741	0.929	2.202	1.607	0.567	3.100	1.686	0.799
BGSW norm 0.10	3.618	3.881	0.932	2.157	3.558	0.556	3.090	3.746	0.796
BGSW norm 0.01	3.617	12.269	0.932	2.158	11.238	0.556	3.090	11.838	0.796

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C23: Relative width of confidence intervals: $n = 2960$ and $1 - \alpha = 0.95$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	2.282		1.000	2.282		1.000	2.282		1.000
Group-balanced	2.113		0.926	1.676		0.735	1.938		0.849
GSW 0.99	2.249	1.373	0.986	2.163	1.321	0.948	2.213	1.351	0.970
GSW 0.90	2.143	1.369	0.939	1.732	1.107	0.759	1.980	1.265	0.868
GSW 0.50	2.087	1.785	0.915	1.449	1.241	0.635	1.844	1.579	0.808
GSW 0.10	2.071	3.972	0.908	1.396	2.674	0.612	1.819	3.485	0.797
GSW 0.01	2.072	12.552	0.908	1.391	8.418	0.610	1.816	11.004	0.796
BGSW 0.99	2.106	1.389	0.923	1.633	1.374	0.716	1.918	1.383	0.841
BGSW 0.90	2.087	1.442	0.915	1.488	1.374	0.652	1.857	1.415	0.814
BGSW 0.50	2.074	1.925	0.909	1.406	1.794	0.616	1.823	1.873	0.799
BGSW 0.10	2.071	4.301	0.908	1.393	3.993	0.610	1.817	4.180	0.796
BGSW 0.01	2.071	13.598	0.908	1.390	12.619	0.609	1.816	13.214	0.796
GSW norm 0.99	2.203	1.343	0.966	1.964	1.198	0.861	2.108	1.285	0.924
GSW norm 0.90	2.100	1.342	0.920	1.524	0.974	0.668	1.879	1.201	0.824
GSW norm 0.50	2.073	1.778	0.909	1.407	1.205	0.617	1.823	1.563	0.799
GSW norm 0.10	2.075	3.970	0.909	1.393	2.665	0.610	1.819	3.481	0.797
GSW norm 0.01	2.072	12.552	0.908	1.391	8.415	0.609	1.817	11.002	0.796
BGSW norm 0.99	2.091	1.380	0.916	1.526	1.326	0.669	1.873	1.359	0.821
BGSW norm 0.90	2.074	1.436	0.909	1.413	1.340	0.619	1.825	1.398	0.800
BGSW norm 0.50	2.070	1.923	0.907	1.393	1.786	0.611	1.816	1.870	0.796
BGSW norm 0.10	2.073	4.300	0.909	1.391	3.991	0.609	1.816	4.179	0.796
BGSW norm 0.01	2.073	13.598	0.908	1.390	12.618	0.609	1.817	13.213	0.796

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C24: Relative width of confidence intervals: $n = 2960$ and $1 - \alpha = 0.99$

	Outcome A			Outcome B			Outcome AB		
Design	C	S	N	C	S	N	C	S	N
Fully random	3.882		1.000	3.882		1.000	3.882		1.000
Group-balanced	3.596		0.926	2.852		0.735	3.297		0.849
GSW 0.99	3.826	1.252	0.986	3.680	1.205	0.948	3.765	1.232	0.970
GSW 0.90	3.647	1.249	0.939	2.947	1.009	0.759	3.369	1.154	0.868
GSW 0.50	3.552	1.628	0.915	2.465	1.131	0.635	3.138	1.440	0.808
GSW 0.10	3.524	3.622	0.908	2.376	2.439	0.612	3.094	3.178	0.797
GSW 0.01	3.526	11.447	0.908	2.366	7.677	0.610	3.090	10.034	0.796
BGSW 0.99	3.583	1.266	0.923	2.778	1.253	0.716	3.264	1.261	0.841
BGSW 0.90	3.551	1.315	0.915	2.532	1.253	0.652	3.160	1.290	0.814
BGSW 0.50	3.529	1.755	0.909	2.393	1.636	0.616	3.101	1.708	0.799
BGSW 0.10	3.524	3.922	0.908	2.369	3.641	0.610	3.092	3.812	0.796
BGSW 0.01	3.523	12.400	0.908	2.365	11.507	0.609	3.090	12.050	0.796
GSW norm 0.99	3.749	1.225	0.966	3.342	1.093	0.861	3.586	1.171	0.924
GSW norm 0.90	3.573	1.223	0.920	2.594	0.889	0.668	3.198	1.095	0.824
GSW norm 0.50	3.527	1.622	0.909	2.394	1.099	0.617	3.102	1.425	0.799
GSW norm 0.10	3.530	3.620	0.909	2.370	2.430	0.610	3.095	3.174	0.797
GSW norm 0.01	3.525	11.446	0.908	2.366	7.674	0.609	3.091	10.033	0.796
BGSW norm 0.99	3.557	1.258	0.916	2.596	1.209	0.669	3.187	1.239	0.821
BGSW norm 0.90	3.528	1.309	0.909	2.404	1.222	0.619	3.106	1.275	0.800
BGSW norm 0.50	3.522	1.754	0.907	2.370	1.629	0.611	3.091	1.705	0.796
BGSW norm 0.10	3.528	3.921	0.909	2.366	3.639	0.609	3.091	3.811	0.796
BGSW norm 0.01	3.527	12.400	0.908	2.364	11.507	0.609	3.091	12.050	0.796

C = Chebyshev, S = Subgaussian bound, N = Normal approximation

Table C25: Conservativeness of RMSE bound, $n = 30$

Design	A	B	AB
GSW 0.99	1.00	1.00	1.00
GSW 0.90	1.02	1.02	1.02
GSW 0.50	1.18	1.18	1.15
GSW 0.10	2.19	2.01	1.93
GSW 0.01	6.51	5.63	5.39
BGSW 0.99	1.06	1.52	1.33
BGSW 0.90	1.08	1.57	1.37
BGSW 0.50	1.29	2.01	1.71
BGSW 0.10	2.57	4.24	3.59
BGSW 0.01	7.85	13.49	11.33
GSW norm 0.99	1.00	1.00	1.00
GSW norm 0.90	1.02	1.02	1.02
GSW norm 0.50	1.22	1.22	1.20
GSW norm 0.10	2.24	2.10	2.03
GSW norm 0.01	6.47	5.79	5.46
BGSW norm 0.99	1.06	1.52	1.33
BGSW norm 0.90	1.08	1.59	1.37
BGSW norm 0.50	1.32	2.11	1.74
BGSW norm 0.10	2.58	4.39	3.59
BGSW norm 0.01	7.78	13.75	11.30

Table C26: Conservativeness of RMSE bound, $n = 296$

Design	A	B	AB
GSW 0.99	1.00	1.00	1.00
GSW 0.90	1.05	1.05	1.05
GSW 0.50	1.39	1.39	1.39
GSW 0.10	3.08	3.06	3.07
GSW 0.01	9.72	9.57	9.66
BGSW 0.99	1.07	1.42	1.28
BGSW 0.90	1.12	1.52	1.35
BGSW 0.50	1.48	2.16	1.83
BGSW 0.10	3.29	5.03	4.13
BGSW 0.01	10.41	16.00	13.10
GSW norm 0.99	1.00	1.00	1.00
GSW norm 0.90	1.05	1.04	1.05
GSW norm 0.50	1.39	1.38	1.38
GSW norm 0.10	3.08	3.05	3.07
GSW norm 0.01	9.72	9.57	9.65
BGSW norm 0.99	1.07	1.43	1.28
BGSW norm 0.90	1.11	1.58	1.37
BGSW norm 0.50	1.48	2.24	1.86
BGSW norm 0.10	3.30	5.07	4.15
BGSW norm 0.01	10.42	16.00	13.11

Table C27: Conservativeness of RMSE bound, $n = 2960$

Design	A	B	AB
GSW 0.99	1.01	1.01	1.01
GSW 0.90	1.05	1.05	1.05
GSW 0.50	1.41	1.41	1.41
GSW 0.10	3.16	3.15	3.16
GSW 0.01	9.97	9.97	9.98
BGSW 0.99	1.09	1.39	1.31
BGSW 0.90	1.14	1.52	1.40
BGSW 0.50	1.53	2.10	1.91
BGSW 0.10	3.42	4.72	4.28
BGSW 0.01	10.81	14.95	13.54
GSW norm 0.99	1.00	1.00	1.00
GSW norm 0.90	1.05	1.05	1.05
GSW norm 0.50	1.41	1.41	1.41
GSW norm 0.10	3.15	3.15	3.15
GSW norm 0.01	9.97	9.96	9.97
BGSW norm 0.99	1.09	1.43	1.33
BGSW norm 0.90	1.14	1.56	1.42
BGSW norm 0.50	1.53	2.11	1.91
BGSW norm 0.10	3.41	4.73	4.28
BGSW norm 0.01	10.80	14.95	13.53