PUMP: Estimating power, MDES, and sample size with multiple outcomes

1 Introduction

The PUMP R package fills an important gap in open source software and tools needed by researchers to design multi-level randomized controlled trials (RCTs) with sufficient statistical power. With this package, researchers can estimate statistical power, minimum detectable effect size (MDES), and needed sample size for multi-level experimental designs, in which units are nested within hierarchical structures such as students nested within schools nested with school districts. The statistical power is calculated for estimating the impact of a single intervention on multiple outcomes. The package focuses on a frequentist framework of mixed effects regression models, which is currently the prevailing framework for estimating impacts from experiments in education and other social policy research.[1] To our knowledge, none of the existing software options or tools for power calculations power allow researchers to account for multiple hypothesis tests and the use of a multiple testing procedure (MTP). MTPs adjust p-values to reduce the likelihood of spurious findings when researchers are testing for effects on multiple outcomes. This adjustment can result in a substantial change in statistical power, greatly reducing the probability of detecting effects when they do exist. Unfortunately, when designing studies, researchers who plan to test for effects on multiple outcomes and employ MTPs frequently ignore the power implications of the MTPs.

Also, as researchers change their focus from one outcome to multiple outcomes, multiple definitions of statistical power emerge (Chen (2011); Dudoit (2003); Senn and Bretz (2007); T. Westfall Peter H and Wolfinger (2011)). The PUMP package allows researchers to consider multiple definitions of power, as some may be more appropriate for the goals of their study. For example, one might consider *individual power,* the probability of detecting an effect of a particular size (specified by the researcher) or larger for each hypothesis test. Individual power corresponds to how power is defined when there is focus on a single outcome. However, researchers might also consider *1-minimal power,* the probability of detecting effects of at least a particular size on at least one outcome. In contrast, one might consider *complete power,* the power to detect effects of at least a particular size on *all* outcomes.

The prevailing default in many studies—individual power—may or may not be the most appropriate type of power. If the researchers' goal is to find statistically significant estimates of effects on most or all primary outcomes of interest, then even after taking multiplicity adjustments into account, estimates of individual power can grossly understate the actual power required. On the other hand, if the researchers' goal is to find statistically significant estimates of effects on at least one or a small proportion of outcomes, their power may be much better than anticipated. They may be able to reduce their sample size, or they may be able to detect smaller effects. The PUMP package allows for directly answering questions such as "How many schools would I need to detect a given effect on at least three of my five outcomes," or "What size effect can I reliably detect on each outcome, given a planned MTP across all my outcomes," or "How would the power to detect a given effect change if only half my outcomes truly had impacts."

PUMP implements new, rigorously validated methods developed by the authors to estimate statistical power for multiple definitions of statistical power across a wide range of common experimental designs. PUMP extends functionality of the popular PowerUp! R package (and related tools in the form of a spreadsheet and Shiny application), which compute power or MDES for multi-level RCTs with a single outcome (Dong and Maynard (2013)). For a wide variety of RCT designs with a single outcome, researchers can take advantage

[1] Other options include nonparametric or Bayesian methods, but these are less prevalent in applied research (for example, see @GELMANETAL2012).

of closed-form solutions and numerous power estimation tools. For example, in education and social policy research, see Dong and Maynard (2013); Hedges and Rhoads (2010); Raudenbush et al. (2011); Spybrook et al. (2011). However, closed-form solutions are difficult or impossible to derive when an MTP is applied to a setting with multiple outcomes. Instead, we use a simulation-based approach to achieve estimates of power.

In order to calculate power, the researcher specifies information about the sample size at each level, the minimum detectable effect size (MDES) for each outcome (the smallest true effect sizes to detect with statistical significance, in units of standard deviations), the level of statistical significance, and parameters of the data generating distribution. An "effect size" generally refers to the standardized mean difference effect size, which "equals the difference in mean outcomes for the treatment group and control group, divided by the standard deviation of outcomes across subjects within experimental groups" (Bloom (2006)). Researchers often use effect sizes to standardize outcomes so that outcomes with different scales can be directly compared. With PUMP, the user can also choose to apply any of five common MTPs — Bonferroni, Holm, single-step and step-down versions of Westfall-Young, and Benjamini-Hochberg. In addition to estimating power, the researcher can target a level of power and estimate either MDES or sample size requirements. Therefore, the package includes three core functions:

- `pump_power()` for calculating power given a experimental design and assumed model, parameters, and minimum detectable effect size.
- `pump_mdes()` for calculating minimum detectable effect size given a target power.
- `pump_sample()` for calculating the required sample size for achieving a given target power for a given minimum detectable effect size.

The package also includes functions that allow users to easily explore power, MDES or sample size estimates over a range of possible values of parameters to determine sensitivity of calculations to ranges of assumptions. PUMP also visually displays results. These additional functions include:

- `pump_power_grid()`, `pump_mdes_grid()`, and 'pump_sample_grid() for calculating the given output over a range of possible parameter values.
- `update()` to re-run an existing calculation with a small number of parameters updated.
- `plot()` on any pump-generated object to generate plots (including grid outputs).

The PUMP package covers a range of multi-level designs that researchers typically use in practice, in which research units are nested in hierarchical groups. These designs are common in education research, for example, in which students are nested within schools, which can be then nested within school districts. Our power calculations assume the user will be analyzing these RCTs using frequentist mixed-effects regression models, containing a combination of fixed or random intercepts and treatment impacts at different levels, as we explain in detail in Section~4.1 and in the Supplementary Materials.

The remainder of this paper is organized as follows: In Section 2, we provide a summary of the multiple testing problem. To do so, we begin with a motivating example from education research, which we refer to throughout the paper. We note, however, that the problem of power estimation for multi-level RCTs is not exclusive to the educational setting. Also in Section 2, we present an overview of the statistical challenges introduced by multiple hypothesis testing and how MTPs protect against spurious impact findings. In Section 3, we introduce our methodology for estimating power when taking the use of MTPs into account. This section includes a summary of the RCT designs our methods cover, a discussion of how to specify key parameters, and a discussion of our validation processes. Section 4 provides a detailed presentation of the PUMP package with multiple examples of using the packages functions to conduct calculations for our education RCT example. Section 5 is a brief conclusion.

2 Diplomas Now

We illustrate our package with a published RCT evaluation of a secondary school model called Diplomas Now. The Diplomas Now model is designed to increase high school graduation rates and post-secondary readiness. Evaluators evaluated whether the program was effective by conducting a RCT comparing schools

who implemented the model to versus business as usual. We refer to this example to illustrate key concepts and to illustrate the application of the PUMP package.

The Diplomas Now model, created by three national organizations, Talent Development, City Year, and Communities In Schools, targets underfunded urban middle and high schools with many students who are not performing well academically. The model is designed to be robust and intense enough to transform or turn around high-poverty and high-needs middle grade and high schools attended by many students who fall off the path to high school graduation. Diplomas Now, with MDRC as a partner, was one of the first validation grants awarded as part of the Investing in Innovation (i3) competition administered by the federal Department of Education. We follow the general design of the Diplomas Now evaluation, conducted by MDRC. The RCT contains three levels (students within schools within districts) with random assignment at level 2 (schools). The initial evaluation, which included two cohorts of schools with each cohort implementing for two years (2011-2013 for Cohort 1 and 2012-2014 for Cohort 2), included 62 secondary schools (both middle and high schools) in 11 school districts that agreed to participate. Schools in the active treatment group were assigned to implement the Diplomas Now model, while the schools in the control group continued their existing school programs or implemented other reform strategies of their choosing (See Corrin W. (2016).)

The MDRC researchers conducted randomization of the schools within blocks defined by district, school type, and year of roll-out. After having to drop some schools due to various reasons, the researchers were left with 29 high schools and 29 middle schools grouped in 21 random assignment blocks. Within each block, schools were randomized to the active treatment or business-as-usual, resulting in 32 schools in the treatment group, and 30 schools in the control group.

The evaluation focused on three categories of outcomes (Attendance, Behavior, Course performance, called the "ABC's," with multiple measures for each category), along with overall ABC composite measures of whether a student is above given thresholds on all three categories. This grouping constitutes 12 total outcomes of interest. Evaluating each of the 12 outcomes independently would not be good practice, as the chance of a spurious finding is not well controlled. The authors of the MDRC report pre-identified three of these outcomes as *primary* outcomes before the start of the study in order to protect against such spurious findings that might arise from conducting multiple hypothesis tests without adjustment. We, by contrast, use this example to illustrate what could be done if there was uncertainty as to what should be the primary outcome. In particular, we illustrate how to conduct a power analysis to plan a study where one uses multiple testing adjustment, rather than predesignation, to account for the multiple outcome problem.

Due to the grouped nature of the outcomes, we elect to do a power analysis separately for each outcome group (mimicking the three chosen outcomes of the original study) to control family-wise error rather than overall error. We would then adjust for the number of outcomes within each group independently. We note that there are different guidelines for when to adjust for multiple outcomes in education studies.[2] This paper would apply to either case. In this paper, the word "outcome" refers to either a single outcome or an outcome domain, and the paper focuses on any situation in which an analyst would apply adjustments to account for multiple outcomes.] We do not provide recommendations for which guidelines to follow when investigating impacts on multiple outcomes. Rather, we address the fact that researchers across many domains are increasingly applying MTPs and therefore need estimate power, MDES and sample size requirements. For illustration purposes, we focus on an alternative approach in the Diplomas Now study on assessing impacts of all 12 outcomes and applying MTPs to protect against the potential for spurious findings.

3 Overview of multiple testing

Our motivating example illustrates that researchers are often interested in testing the effectiveness of an intervention on multiple outcomes. The resulting multiplicity of statistical hypothesis tests can lead to

[2]For example, Schochet (2008) recommends organizing primary outcomes into domains, conducting tests on composite domain outcomes, and applying multiplicity corrections to composites across domains. The What Works Clearinghouse applies multiplicity corrections to findings within the same domain rather than across different domains.

spurious findings of effects.[3] Multiple testing procedures (MTPs) counteract this problem by adjusting p-values for effect estimates; generally, p-values are adjusted upward to require a higher burden of proof. When not using an MTP, the probability of finding false positives increases, sometimes dramatically, with the number of tests. When using an MTP, this probability is controlled to a specified level.

Consider that a researcher is interested in testing the impact of an intervention on M outcomes. In our running example of the Diplomas Now study, if we had 5 outcomes in the attendance group, we would have $M = 5$. In the frequentist framework, when framing impacts in terms of effect sizes (ES), for the outcome m, one typically tests a null hypothesis of no effect, $H_{0_m} : ES_m = 0$, against an alternative hypothesis $H_{1_m} : ES_m \neq 0$ for a two-sided tests or $H_{1_m} : ES_m > 0$ or $H1_m : ES_m < 0$ for a one-sided test. A significance test, such as a two- or one-sided t-test, would then typically be driven by whether a test statistic given by

$$t_m = \frac{\hat{ES}_m}{SE(\hat{ES}_m)}, \tag{1}$$

from which a raw p-value is computed. Here, the term "raw" is used to distinguish this p-value from a p-value that has been adjusted using a procedure for multiple hypothesis tests, as discussed below. The raw p-value is the probability of a test statistic being at least as extreme as the one observed, given that the null hypothesis is true.

For a two-sided test, which is the focus of the discussion going forward (although the PUMP package also allows for one-sided tests), the raw p-value for test m is $p_m = 2 * Pr(T_m \geq |t_m|)$.[4] To calculate the raw p-value, we use our knowledge of the sampling distribution of the t-statistic, and we identify where our observed test statistic falls in that distribution.

When testing a *single* hypothesis under this framework (effects are being assessed on just one outcome, so that $M = 1$), researchers typically specify an acceptable maximum probability of making a Type I error, α. A Type I error is the probability of erroneously rejecting the null hypothesis when it is true. The quantity α is also referred to as the significance level. If $\alpha = 0.05$, then the null hypothesis is rejected if the p-value is less than 0.05, and it is concluded that the intervention had an effect because there is less than a 5% chance that this finding is a false positive.

When one tests *multiple* hypotheses under this framework (such that $M > 1$) and one conducts a separate test for each of the hypotheses with $\alpha = 0.05$, there is a *greater* than 5% chance of a false positive finding in the study. If the multiple tests are independent, the probability that at least one of the null hypothesis tests will be erroneously rejected is 1-Pr(none of the null hypotheses will be erroneously rejected) $= 1 - (1 - \alpha)^M$. Therefore, if researchers are estimating effects on three outcomes (and if these outcomes are independent) the probability of at least one false positive finding is $1 - (1 - 0.05)^3 = 0.14$. If the researchers were instead estimating effects on five independent outcomes, the probability of at least one false positive finding rises to 0.23. This Type I error inflation for independent outcomes demonstrates the crux of the multiple testing problem. In practice, however, the multiple outcomes are usually at least somewhat correlated, which makes the test statistics correlated and reduces the extent of Type I error inflation. Nonetheless, any error inflation can still make it problematic to draw reliable conclusions about the existence of effects.

3.1 Using MTPs to protect against spurious impact findings

As introduced above, to counteract the multiple testing problem, multiple testing procedures (MTPs) adjust p-values.[5] We next describe how using a MTP protects against false positives.

[3] Testing the effectiveness of an intervention for multiple subgroups, at multiple points in time, or across multiple treatment groups also results in a multiplicity of statistical hypotheses and can also lead to spurious findings of effects, but this is beyond the scope of this paper.

[4] For a one-sided test, depending on the direction of our alternative hypothesis, the raw p-value for test m is computed as $p_m = Pr(T_m \leq t_m)$ or $p_m = Pr(T_m \geq t_m)$.

[5] Alternatively, MTPs can decrease the critical values for rejecting hypothesis tests. For ease of presentation, this paper focuses only on the approach of increasing p-values.

Recall that the power of an individual hypothesis test is the probability of correctly rejecting a null hypothesis of at least a specified size. We refer to a setting in which the impact is truly at least the size of the desired effect size as a "false null" hypothesis, while a "true null" is an impact less than the desired effect size. If raw p-values are adjusted upward, one is less likely to reject the null hypotheses that are true (meaning there is truly no effect of at least a specified size), which reduces the probability of Type I errors, or false positive findings. At the same time, MTPs reduce individual power (the power of an individual hypothesis test) compared with the situation when no multiplicity adjustments are made or the situation when there is only one hypothesis test.

MTPs also reduce all other definitions of power compared with the situation when no multiplicity adjustments are made – but not necessarily compared with the situation when there is only one hypothesis test. For example, 1-minimal power, the probability of detecting effects (of at least a specified size) on at least one outcome – after adjusting for multiplicity – is typically greater than the probability of detecting an effect of the same size on a single, prespecified, outcome. This increase may or may not occur with other definitions of power (e.g., the probability of detecting a third, half, or all false null hypotheses).

The MTPs that are the focus of this paper have three key features that affect statistical power (or MDES or needed sample size): (1) whether the MTP is a familywise procedure or a false discovery rate procedure (2) whether the MTP is single-step or stepwise; and (3) whether the MTP takes the correlation between test statistics into account. Below we explain each of these features of MTPs and provide discussion of the new parameter specifications they require when estimating power, MDES or sample size requirements.

3.1.1 Familywise error rate vs. false discovery error rate

Familywise procedures reframe Type I error as a rate across the entire set or "family" of multiple hypothesis tests. This rate is called the familywise error rate (FWER; Tukey (1953)). The FWER is typically set to the same value as the probability of a Type I error for a single test, e.g., α. MTPs that control the FWER at 5% adjust p-values in a way that ensures that the probability of at least one Type I error across the entire set of hypothesis tests is no more than 5%. The MTPs introduced by Bonferroni (Dunn (1959), Dunn (1961)), Holm (1979), and P. H. Westfall and Young (1993) control the FWER.

MTPS that control false discovery rate (FDR) take an entirely different approach to the multiple testing problem.

FDR is a less stringent criteria than FWER. Introduced by Benjamini and Hochberg (1995), the FDR is the expected proportion of all rejected hypotheses that are erroneously rejected. The two-by-two representation in Table~1 is often found in articles on multiple hypothesis testing, and helps to illustrate the difference between FWER and FDR. Let M be the total number of tests. Therefore, we have M unobserved truths: whether or not each null hypothesis is true or false. We also have M observed decisions: whether or not the null hypotheses were rejected, because the p-values were less than α. In Table~1, A, B, C and D are four possible scenarios: the numbers of true or false hypotheses not rejected or rejected. M_0 and M_1 are the unobservable numbers of true null and false null hypotheses. R is the number of null hypotheses that were rejected, and $M - R$ is the number of null hypotheses that were not rejected.

	Observed decisions		
Unobserved truths	Number not rejected	Number rejected	Total
Number of true null hypotheses	A	B	M0
Number of false null hypotheses	C	D	M1
Total	M-R	R	M

Table 1: Numbers of hypothesis types and decisions.

In Table~1, F is the number of erroneously rejected null hypotheses, or the number of false positive findings. Therefore, the FWER is equivalent to $Pr(F > 0)$, the probability of at least one false positive finding. Recall that Type I error is inflated when testing for effects on independent outcomes when no MTPs are applied. The Type I error was almost 10% when testing effects on two independent outcomes and 23% when testing

effects on five independent outcomes. These Type I error rates both correspond to the FWER. The goal of MTPs that control the FWER is to bring these percentages back down to 5%.

Also in Table~1, the FDR is equal to $E(\frac{F}{R})$ but is defined to be 0 when $R = 0$, or when no hypotheses are rejected. As is frequently noted in the literature (e.g., Shaffer (1995); Schochet (2008)), the FWER and FDR have different objectives. Control of the FWER protects researchers from spurious findings and so may be preferred when even a single false positive could lead to the wrong conclusion about the effectiveness of an intervention. On the other hand, the FDR is more lenient with false positives. Researchers may be willing to accept a few false positives, E, when the total number of rejected hypotheses, R, is large. Note that under the complete null hypothesis that all M null hypotheses are null, the FDR is equal to the FWER, because when referring back to Table~1 we have $FWER = P(R > 0) = E(\frac{E}{R}) = FDR$. However, if any effects truly exist, then $FWER \geq FDR$.As a result of the difference in objective between FWER and FDR, in the case where there is at least one false null hypothesis (at least one true effect at least as large as a specified effect size), an MTP that controls the FDR at 5% will have a Type I error rate that is greater than 5%.

MTPs may provide either weak or strong control of the error rate they target. An MTP provides weak control of the FWER or the FDR at level α if the control can only be guaranteed when all nulls are true, or when the effects on all outcomes are zero. An MTP provides strong control of the FWER or FDR at level if the control is guaranteed when some null hypotheses are true and some are false, or when there may be effects on at least some outcomes. Of course, strong control is preferred.[6]

3.1.2 Single-step vs. stepwise procedures

A second feature of an MTP that affects its statistical power is whether it is "single-step" or "stepwise" procedure. Single-step procedures adjust each p-value independently of the other p-values. For example, the Bonferroni MTP multiplies all raw p-values by M. Therefore, one p-value adjustment does not depend on other p-value adjustments, only on the number of tests. In contrast, stepwise procedures first order raw p-values (or test statistics), and then adjust according to the order of the tests. The adjustments depend on null hypotheses already rejected in previous steps. For example, the Holm MTP — the stepwise counterpart to the Bonferroni MTP — orders raw p-values from smallest to largest. The procedure then multiplies the smallest p-value by M, the second smallest p-value by $M - 1$, and so on, but also enforces that each adjusted p-value is greater than or equal to the previous adjusted p-value and that it is not greater than one. Overall, stepwise MTPs allow for less adjustment than single-step MTPs in later steps, and therefore preserve more power (for outcomes that do not correspond to the most significant p-value). The Bonferroni and one of the Westfall-Young MTPs are single-step; the Holm and Benjamini-Hochberg MTPs and the other Westfall-Young MTP are stepwise. Note that stepwise procedures may be "step-down" or "step-up," referring to whether a procedure begins with the smallest p-value, and thus the largest effect size (step-down) or the largest p-value (step-up).

Due to the dependencies of adjustments in stepwise MTPs, a new assumption must be considered when estimating power, MDES or sample size under multiplicity - the proportion of outcomes on which there are truly impacts of at least the size of the researchers' desired ESs, or, equivalently, the number of false null hypotheses. There is one scenario in which this assumption does not matter, which is the scenario when one focuses on individual power and uses a single-step MTP. In this case, when adjusting a p-value for a single test, the information from other tests is disregarded. For all other scenarios, however, this assumption can be an important one.

Researchers may be inclined to assume that there will be effects on all outcomes, as hypotheses of effects probably drive the selection of outcomes in the first place. And when estimating power for a single hypothesis test, power is only defined when a true effect exists. However, if the researchers are incorrect and there turn

[6]The single-step and step-down Westfall Young MTPs (which we discuss below) always provide at least weak control of the FWER. In order for these procedures to provide strong control of the FWER, they require the assumption of subset pivotality (RN33093). The distribution of the unadjusted test statistics or p-values is said to have subset pivotality if for any subset of null hypotheses, the joint distribution of the test statistics or of the $p-$values for the subset is identical to the distribution under the complete null. A consequence of this assumption is that the resampling of test statistics or p-values can be done under the complete null hypothesis rather than under the unknown partial hypothesis (RN33093).

out not to be effects on all outcomes, the probability of detecting the effects that do actually exist can be diminished, sometimes substantially.

It is important to point out that under the assumption that there are not truly effects on every outcome under study, the definitions of the d-minimal powers (e.g., 1-minimal power, 1/3-minimal power, etc.) and of complete power become fuzzy. For example, 1/3-minimal power is defined as the probability of detecting effects (of a specified size or larger) on at least 1/3 of the total outcomes (M), regardless of the number of outcomes with actual effects. That is, 1/3-power is not defined as the probability of detecting effects among the M outcomes on which the effects truly exist. Therefore, while individual power is defined based on false nulls, the definition is loosened here and includes the probability of erroneous rejections of true nulls (which are controlled to occur at no more than 5% for those MTPs that control the FWER). This fuzziness of definition is needed because the researcher would only ever define power based on the total number of tests. Moreover, if the $d-$minimal powers are defined only based on false nulls, then their levels could increase when the proportion of false nulls decreases. Complete power has the same issue. If there are truly only effects on two of the three outcomes, then complete power is not the probability of rejecting just two false null hypotheses. In this case, complete power is undefined.

There is an additional technical note about the calculation of complete power. To calculate complete power, we do not need to adjust the p-values, and can instead reject each individual test based on the unadjusted p-values. Complete power is the power of the test constructed by whether or not we reject all the null hypotheses.[7] This test was originally introduced as the intersection-union test because the null hypothesis is expressed as a union and the alternative hypothesis is expressed as an intersection Berger and Hsu (1996). Berger (1982) showed that if all the individual tests are level α, the intersection-union test is also a levelα test. To provide some intuition, we do not need to adjust p-values for complete power because it is a special case where we must reject *all* the hypothesis tests. Thus, there is no way for the omnibus test to be rejected by chance because of a favorable configuration (Chen 2011). For example, consider if we have 4 tests, with 2 false nulls and 2 true nulls. If we consider 3-minimal power, we just need one of the two true nulls to be rejected by chance alone, and there are two ways for this to occur. For complete power, there is only one way for us to reject all of the nulls. The downside of an intersection-union test is that it is conservative: the FWER is generally less than α. For example, if we have two independent tests with type I error α, then if both of the nulls are true, the probability of a type I error is α^2 (Deng, Xu, and Wang 2008).

3.1.3 Correlation between test statistics

Finally, a third feature of an MTP that affects its statistical power is whether or not it takes into account the correlation of test statistics. The Bonferroni and Holm procedures strongly control the FWER in call cases, even when the multiple tests' statistics are correlated, but they adjust p-values more than is necessary in that case. Along with the Bonferroni and Holm MTPs, the Benjamin-Hochberg MTP also does not take correlations into account.[8] In contrast, both of the Westfall-Young MTPs rely on the estimation of the joint distribution of test statistics when the "complete null hypothesis" (that there are not effects on any of the outcomes) is true. This joint distribution of the test statistics is estimated from the study's data. For example, permutations of the treatment indicator can be used to estimate impacts when the association between treatment status and the outcome is broken. Random permutations of the research group assignments are conducted a large number of times, resulting in a distribution of test statistics under the complete null. Because the actual data are used to generate this null distribution, correlations among the test statistics are captured. Then observed test statistics can be compared with the distribution of test statistics under the complete null hypothesis.[9]

The correlation between test statistics is a parameter a researcher must specify in order to estimate power,

[7] Complete power has also been referred to as "conjunctive power" (Bretz et al., 2011) and "all pairs power" (@RN33097).

[8] The Benjamini-Hochberg procedure was originally shown to control the FDR for independent test statistics. However, Benjamini and Yekutieli (2001) showed that it also controls the FDR for true null hypotheses with "positive regression dependence." This condition is satisfied for most applications in practice.

[9] Instead of using test statistics, the Westfall-Young MTPs can alternatively compare raw p-values with the estimated joint null distribution of p-values.

MDES or sample size requirements when using an MTP. The M pairwise correlations are equal to the M pairwise correlations between the residuals in the M impact models. If there are no covariates in the impact models or if the R^2's of the covariates are equivalent in all impact models, then the correlations between the test statistics are equal to the correlations between the outcomes. However, having different R^2's across the impact models reduces the correlations between the residuals and therefore between test statistics.[10] Models of outcomes that are highly correlated are more likely to have residuals that are highly correlated because baseline covariates will tend to have similar R^2's. The gaps between the correlations between outcomes and the correlations between residuals — and therefore the test statistics — may be wider for moderately or weakly correlated outcomes. In any case, the upper bounds of correlations between the test statistics are the correlations between the outcomes.

To review, the list of multiple testing procedures supported by the PUMP package is:

- *Bonferroni*: adjusts p-values by multiplying them by M to ensure strong control of the FWER. Bonferroni is a simple procedure, but the most conservative.
- *Holm*: a step-down version of Bonferroni. Starting from smallest to largest, p-values are sequentially adjusted by different multipliers. Holm is less conservative than Bonferroni for larger p-values.
- *Benjamini-Hochberg*: A sequential, step-up procedure that controls the FDR. Using the BH method, only null hypotheses with p-values below a certain threshold are rejected, where the threshold is determined by the number of tests and the level α.
- *Single-step Westfall-Young*: A permutation-based procedure for controlling the FWER, which directly takes into account the joint correlation structure of the outcomes. In the single-step approach, all outcomes are adjusted by using the permuted distribution of the minimum p-value. Although Westfall-Young procedures are less conservative while still protecting against false discoveries, they are computationally very intensive.
- *Step-down Westfall-Young*: A similar approach to the single-step procedure, except that outcomes are adjusted sequentially from smallest to largest according to the permuted distributions of the corresponding sequential p-values. For a more detailed explanation of each MTP, see Appendix A of Porter (2018).

The following table summarizes the important features for each of the MTPs supported by PUMP:

Procedure	Control	Single-step or stepwise	Accounts for correlation
Bonferroni (BF)	FWER	single-step	No
Holm (HO)	FWER	stepwise	No
Westfall-Young Single-step (WY-SS)	FWER	single-step	Yes
Westfall-Young Step-down (WY-SD)	FWER	stepwise	Yes
Benjamini-Hochberg (BH)	FDR	stepwise	No

Table 2: Summary of MTP procedures.

4 Estimating power, MDES and sample size in studies with multiple outcomes

4.1 Power estimation strategy

We take an innovative simulation-based approach to estimating power for multiple outcomes, as introduced in Porter (2018). We then build on this approach to estimating MDES and sample size. We view simulation as necessary: in order to estimate power for a single outcome, we can often use closed-form algebraic expressions, which are derived from the assumed model. However, with multiple outcomes, finding such expressions can be quite difficult, or even impossible, depending on the multiple testing procedure. In cases where it is possible

[10] For example, one of the multiple outcomes may have a baseline covariate with a high R^2 while another may have a baseline covariate with a smaller R^2. Also, block dummies may explain more variation in some outcomes than in others.

to find a closed-form expression, we would need to find expressions for every design, MTP, and definition of power. Importantly, we would *also* need to find new expressions for each number of outcomes, which quickly becomes an intractable problem! Furthermore, in some cases, such as permutation-based procedures like Westfall-Young approaches, a closed-form solution does not exist. To avoid these complexities, we rely on simulation to calculate estimated power. The approach outlined below can estimate power for any scenario.

If we were to rely on a *full* simulation approach, we could use the following method to estimate power. We note that we introduce this full simulation approach to provide intuition, but use a simplified and less computationally intensive approach in the package, discussed below. We illustrate using the Diplomas Now example.

1. *Simulate a data sample according to the joint alternative hypothesis.* First, we formulate what we will refer to as the *joint alternative hypothesis*, which is the set of outcomes we assume to have nonzero treatment effects. We define ψ_m to be the treatment effect for outcome m, with M total outcomes. If we have $M = 5$ outcomes, as in the Diplomas Now study, one possible joint alternative hypothesis is that all outcomes have effects: $H_A : \psi_1 > 0.125, \psi_2 > 0.2, \psi_3 > 0.1, \psi_4 > 0.1, \psi_5 > 0.05$. Another possible joint alternative hypothesis is one where only the first two outcomes have nonzero effects: $H_A : \psi_1 > 0.125, \psi_2 > 0.2, , \psi_3 = \psi_4 = \psi_5 = 0$. Once our joint alternative hypothesis is specified, we would generate simulated data under this hypothesis. To simulate data, we need to specify the full set of parameters as mentioned in Section~4.2.1 that allow for data generation. See the Supplementary Details for more details about the assumed data-generating process. For example, for the Diplomas Now experiment, we would assume a specific data generating process to allow us to simulate synthetic students, schools, and districts, including covariates, outcomes, and treatment assignment. This process would involve specifying parameter values such as R^2, the amount of outcome variation explained by covariates at a particular level, and translating these parameter choices into data-generating parameters, such as the coefficient values for covariates in a linear model.

2. *Estimate impacts on the simulated data.* Given simulated data, we could fit M regression models (specified to match the experimental design and model assumptions). For the proposed mixed models, the relevant functions would be lm(), lmer() from the lme4 library (Bates et al. (2015)), and interacted_linear_estimators() from the blkvar library.[11] Each of these models result in test statistics t_m for the estimated impacts, one statistic for each outcome, along with estimated standard errors.

3. *Calculate unadjusted p-values.* The test statistics and standard errors would in turn give raw (unadjusted) p-values; these are the p-values routinely returned by regression functions in software packages. In Diplomas Now, for example, we would run a regression model of each attendance measure on treatment status and student and school covariates, and extract p-values from the regression outputs.

4. *Repeat above steps (1 through 3) for a large number of iterations.* Denote the number of iterations *tnum*. Repeating steps 1-3 *tnum* times results in a matrix of unadjusted p-values which we call $\mathbf{F}$, and is of dimension $tnum \times M$. One row corresponds to one set of simulated raw p-values from regressions for the 5 attendance outcomes of interest for Diplomas Now.

5. *Adjust p-values.* For each row, corresponding to one simulated dataset, the M raw p-values corresponding to the M hypothesis tests can be adjusted according to the desired multiple testing procedure. This process generates a new matrix $\mathbf{G}$ of adjusted p-values. For Bonferroni, Holm, and Benjamini-Hochberg adjustments, we use the function `p.adjust` in R (found in the `stats` package). We developed our own functions for implementing adjustment using the Westfall-Young procedures. One row corresponds to one set of simulated *adjusted* p-values for the 12 outcomes of interest for Diplomas Now.

6. *Calculate hypothesis rejection indicators.* For each MTP, the matrix of adjusted p-values $\mathbf{G}$ can then be compared with a specified value of α (the default is 0.05, but the value can be changed by the user). For each row, corresponding to one iteration of simulated data, we record whether or not the null hypothesis was rejected for each outcome. This process results in a new matrix $\mathbf{H}$, which contains hypothesis rejection indicators (still of dimension $tnum \times M$). Using $\mathbf{H}$, we can compute all definitions of power.

[11] This package is currently under development on GitHub; see https://github.com/lmiratrix/blkvar

7. *Calculate power.* To compute the different definitions of power:

- *Individual power* for outcome m is the proportion of the `tnum` rows in which the null hypothesis m was rejected (the mean of column m of $\mathbf{H}$). We would have 12 different power values for Diplomas Now, corresponding to each outcome of interest.
- *d-minimal power* is the proportion of the *tnum* rows in which at least d of the M hypotheses were rejected.[12] For example, for Diplomas Now, we could consider 1-minimal power, which is the probability at least one of our outcomes is statistically significant, or 3-minimal power, the probability 3/5 of the outcomes are significant.
- *Complete power* is the proportion of the *tnum* rows in which all of the null hypotheses were rejected based on the raw p-values rather than adjusted p-values. We would be interested in complete power if we want to evaluate whether Diplomas Now resulted in improvement for every single outcome of interest. With 5 outcomes, this criteria is a relatively strict indicator of success!

Above, we outlined a full simulation-based approach for calculating power. This approach would be computationally intensive because of the need to generate and analyze a full simulated dataset at each iteration. We can simplify this process by skipping the simulation of data and modeling steps. Given an assumed model and correlation structure for the test statistics, we can directly sample from $f(t_1, \ldots, t_M)$, the joint alternative distribution of the test statistics. This shortcut vastly improves both the simplicity and the speed of computation. In summary, our approach is:

1. **Generate** draws of *test statistics* $t_1, \ldots, t_M$ *under the joint alternative hypothesis.* This step produces a $tnum \times M$ matrix E.
2. *Calculate unadjusted p-values.* This produces the matrix $\mathbf{F}$, as in the procedure above.
3. *Adjust p-values.* This produces the matrix $\mathbf{G}$, as in the procedure above.
4. *Calculate hypothesis rejection indicators.* This produces the matrix $\mathbf{H}$, as in the procedure above.
5. *Calculate power.*

We now describe how to sample from $f(t_m)$ directly. First, we assume a particular research design and model. In our example based on the Diplomas Now study, the research design is a 3-level experiment, with randomization at level 2. That is, students are nested within schools, which are nested within randomization blocks (where similar schools, defined by districts and school type, were grouped together); schools were the unit of randomization. For our example, we plan for analyzing our data with a linear regression model with fixed intercepts at the district level, random intercepts at the school level, and a constant treatment effect across schools and districts. Define ψ_m as the treatment effect for outcome m. We express treatment effects in terms of effect sizes:

$$ES_m = \frac{\psi_m}{\sigma_m}$$

where σ_m is the standard deviation of outcome Y_m in the control group. In order to calculate power, we also need the standard error of the impact in effect size units, which we denote as

$$Q_m = SE(\hat{ES}_m).$$

The quantity Q_m is a consequence of the assumed model, the number of units at different levels, the percent of units treated, the assumed R^2, and other parameters; our technical appendix shows formula for Q_m for all the designs our package supports. In our Diplomas Now example, Q_m will be a function of the number of students, schools, and districts; the proportion of treated units; the number of student and school covariates; the explanatory power of the student and school covariates; the proportion of variation in the outcome explained by schools and districts; and the amount of impact variation relative to the amount of mean variation. Some parameters, such as the percent of units treated, will generally be known, while others, such as the R^2 at different levels, would need to be supplied by the user through either estimation on pilot data or assumptions based on prior knowledge.

[12] Note that others refer to 1-minimal power simply as "minimal power" (e.g., @RN33095; @RN23882; @MTSAS), "disjunctive power" (e.g., @RN33091), or "any pair" power (@RN33097). @RN23882 use the terminology of "r-power" for what is referred to here as d-minimal power for d>1.

Given the effect sizes ES_m and the standard errors Q_m, we can determine the distribution of the vector of test statistics. When testing the hypothesis for outcome m, the test statistic for a t-test is:

$$t_m = \frac{\hat{ES}_m}{\hat{Q}_m}$$

with degrees of freedom df, also defined by the assumed model. Under the alternative hypothesis for outcome m, t_m has a t distribution with degrees of freedom df and mean ES_m/Q_m. Finally, in addition to the parameters above, we also need to choose the correlation matrix between test statistics ρ to sample from the joint distribution of $t_m, m = 1, \ldots M$.

With these distributions specified, we can calculate p-value adjustment as described above. Note that this approach of simulating test statistics builds on work by Bang (2005), who use simulated test statistics to identify critical values based on the distribution of the maximum test statistics. Their approach produces the same estimates as the approach described here for the single-step Westfall-Young MTP. Chen (2011) derived explicit formulas for $d-$minimal powers of stepwise procedures and for complete power of single-step procedures, but only for 1, 2, or 3 tests. The approach presented here is more generally applicable, as it can be used for all MTPs, for any number of tests, and for all definitions of power discussed in the present paper.

Remark. The p-value adjustment using Westfall-Young procedures is the most complex correction procedure, so we briefly outline it here. Similar to above, we first explain a full simulation approach, and then discuss our simplification. Under a full simulation approach, we would first generate a single dataset under the joint alternative hypothesis and calculate a set of M observed test statistics. Then, we would permute the single simulated dataset, say $B = 3,000$ times, assuming the joint null hypothesis, and calculate test statistics on each of these permuted datasets. This process generates an empirical distribution of B test statistics under the joint null distribution. Next, we compare the distribution of observed test statistics to the generated distribution of test statistics under the joint null distribution to calculate p-values. We would then re-generate a new simulated dataset, and repeat the process. If we were to generate $tnum = 10,000$ datasets under the joint alternative hypothesis, for each of these datasets we generate $B = 3,000$ permuted datasets under the joint null, so we would have to generate $10,000 \times 3,000$ datasets!

When we skip the step of simulating data, then for each iteration t in $1, \ldots, tnum$ we first generate a set of M observed test statistics from the joint alternative distribution. Then, we draw B samples of test statistics under the joint null rather than resampling the data B times. Under the null hypothesis, t_m has a t distribution with degrees of freedom df and mean 0. As before, we then compare the distribution of observed test statistics to the distribution of test statistics under the joint null distribution to calculate p-values. Westfall-Young procedures are computationally intensive, so the approach of skipping the simulated data step is particularly helpful here. This approach substantially reduces computational time by drawing test statistics directly rather than resampling data.

4.2 Designs and models

When planning a study, the researcher first has to identify the design of the experiment, including the number of levels, and the level at which randomization occurs. These decisions can be a mix of the realities of the context (e.g., the treatment must be applied at the school level, and students are naturally nested in schools, making for a cluster randomization), or deliberate (e.g., the researcher groups similar schools to block their experiment in an attempt to improve power). Second, based on the design and the inferential goals of the study, the researchers chooses an assumed model, including whether intercepts and treatment effects should be treated as constant, fixed, or random. For the same experimental design, the analyst can sometimes choose from a variety of possible models, and these two decisions should be kept conceptually separated from each other.

The design. The PUMP package supports designs with one, two, or three levels, with randomization occurring at any level. For example, a design with two levels and randomization at level one is a blocked design (or equivalently a multisite experiment), where level two forms the blocks (blocks being groups of units, some of which are treated and some not). Ideally, the blocks in a trial will be groups of relatively homogenous units,

but frequently they are a consequence of the units being studied (e.g., evaluations of college supports, with students, the units, nested in colleges, the blocks) A design with two levels and randomization at level two is commonly called a cluster design (e.g., a collection of schools, with treatment applied to a subset of the schools, with outcomes at the student level); here the schools are the clusters, with a cluster being a collection of units which is entirely treated or entirely not. We can have both blocking and clustering; for example if we have a series of cluster-randomized experiments within different school districts, we have district at level three; this would be a blocked (by district), cluster-randomized experiment.

The model. Given a design, the researcher can select a model via a few modeling choices. In particular the researcher has to decide about the intercepts and the treatment impacts:

- Whether level 2 and level 3 intercepts are:
  - fixed: we have a separate intercept for each unit
  - random: we have a separate intercept for each unit as above, but model the collection of intercepts as Normally distributed, allowing for partial pooling.
- Whether level 2 and level 3 treatment effects are:
  - constant: we model all units in a level as having the same single average impact.
  - fixed: we allow each block or cluster within a level to have its own individual estimated impact (we can only do this if we have treated and control units within said block or cluster).
  - random: we allow variation as with fixed, but model the collection of treatment impacts as Normally distributed around a grand mean mean impact. This is implicitly allowing for the sample as being representative of a larger super-population, in terms of treatment impact estimation.

We denote the research design by d, followed by the number of levels and randomization level, so `d3.1` is a 3-level design with randomization at level 1. The model is denoted by m, followed by the level and the assumption for the intercept, either f or r and then the assumption for the treatment impacts, c, f, or r. For example, `m3ff2rc` means at level 3, we assume fixed intercepts and treatment impacts, and at level 2 we assume random intercepts and constant treatment impacts. The full design and model are specified by concatenating these together, e.g. `d3.2_m3ff2rc`. The Diplomas Now model, for example, is `d3.2_m3fc2rc`.

The full list of supported design and model combinations is below. We also include the corresponding names from the PowerUP! package where appropriate. For more details about each combination, see the appendix.

Code	Design	Model	PowerUp
d1.1_m1c	d1.1	m1c	n/a
d2.1_m2fc	d2.1	m2fc	bira2_1c
d2.1_m2ff	d2.1	m2ff	bira2_1f
d2.1_m2fr	d2.1	m2fr	bira2_1r
d2.1_m2rr	d2.1	m2rr	n/a
d2.2_m2rc	d2.2	m2rc	cra2_2r
d3.1_m3rr2rr	d3.1	m3rr2rr	bira3_1r
d3.2_m3ff2rc	d3.2	m3ff2rc	bcra3_2f
d3.2_m3fc2rc	d3.2	m3fc2rc	n/a
d3.2_m3rr2rc	d3.2	m3rr2rc	bcra3_2r
d3.3_m3rc2rc	d3.3	m3rc2rc	cra3_3r

4.2.1 Overview of function parameters

The table below shows the parameters that influence Q_m formulae for different designs.

Parameter	Description
nbar	the harmonic mean of the number of level 1 units per level 2 unit (students per school)
J	the number of level 2 units (schools)
K	the number of level 3 units (district)
Tbar	the proportion of units that are assigned to the treatment
numCovar.1	number of Level 1 (individual) covariates
numCovar.2	number of Level 2 (school) covariates
numCovar.3	number of Level 3 (district) covariates
R2.1	percent of variation explained by Level 1 covariates
R2.2	percent of variation explained by Level 2 covariates
R2.3	percent of variation explained by Level 3 covariates
ICC.2	level 2 intraclass correlation
ICC.3	level 3 intraclass correlation
omega.2	ratio of variance of level 2 average impacts to variance of level 2 random intercepts
omega.3	ratio of variance of level 3 average impacts to variance of level 3 random intercepts

A few parameters warrant more explanation.

- The quantity ICC is the Intraclass Correlation, and gives a measure of variation at different levels of the model.

For each outcome, the ICC for each level is defined as the ratio of the variance at that level divided by the overall variance of the individual outcomes. The ICC is for the unconditional model, and therefore includes the variation due to covariates.

- For each outcome, the quantity ω for each level is the ratio between impact variation at that level and mean variation at that level. It is a measure of treatment impact heterogeneity.
- The R^2 expressions are the percent of variation at a particular level predicted by covariates specific to that level. For simplicity we assume covariates at a level are group mean centered, so only covariates at a particular level explain variance at that level.

For precise formulae of these expressions, see the Supplementary materials, which outlines the assumed data-generating process, and the resulting expressions for ICC, ω, and R^2.

In addition to design parameters, there are additional parameters that control the precision of the power estimates themselves:

- *tnum* is the number of test statistics generated in order to estimate power. A larger number of test statistics results in greater computation time, but also a more precise estimate of power. (Note that the `pump_mdes()` and `pump_sample()` have multiple `tnum` parameters controlling the precision of the search).
- B is the number of Westfall-Young permutations. Again, there is a tradeoff between precision and computation time.

4.3 Determining MDES and sample size

Frequently, a researcher's main concern with power is prospectively calculating either the MDES for each outcome in a given study, or determining the necessary sample size to achieve a target power given a specified set of MDES values. For our example based on Diplomas Now, we might be interested in calculating the MDES values of the outcomes in the current study, or determining how much we would want to alter the size of our sample of schools and districts given different power agendas. For example, we might want to know what sized study we would need to detect at least one significant effect across our outcomes if all the outcomes had a specified effect size and we were planning on using the Holm procedure.

For `pump_mdes()` and `pump_sample()`, the user provides a particular target power, say 80%. The method then conducts a stochastic optimization problem to determine a value (of sample size or MDES) that is

within a specified tolerance of the target power with high probability. We discuss the algorithm for MDES, although the approach for the sample sizes is the same.

The algorithm first determines an initial range of MDES values that likely contain the target MDES. The initial range is calculated using formula based on the standard errors and degrees of freedom. In particular, from Dong and Maynard (2013), in general the MDES can be estimated as

$$MDES = MT_{df} \times SE/\sigma_m$$

where MT_{df} is known as the multiplier and is the sum of two t statistics with degrees of freedom df. For one-tailed tests, $MT_{df} = t^{\star}_{\alpha} + t^{\star}_{1-\beta}$ where α is the type I error rate and β is the desired power. For two-tailed tests, $MT_{df} = t^{\star}_{\alpha/2} + t^{\star}_{1-\beta}$. We do not explain the details of the derivations of the multiplier here; for more details and understanding, see Dong and Maynard (2013) or Bloom (2006). These expressions can be further manipulated to obtain sample size formulae; see the technical appendix for all formula in the package.

Using these formula we can calculate bounds by manipulating the β values. For example, if we are interested in complete power of 0.8, for the upper bound we would need each outcome to have an individual power of $0.8^{(1/M)}$ for the Bonferroni correction, assuming independence. If we are interested in minimal power, we must have a smaller lower bound; in order to have 1-minimal power of 0.8, each outcome needs to have individual power of $1 - (1 - 0.8)^{(1/M)}$. We ignore correlation in the setting of the initial bounds; the bounds do not need to be strict, given the adaptive nature of the subsequent search.

Once the initial range is established, we use `pump_power()` with the complete array of design parameters including the correlation between test statistics to obtain rough (using a small `tnum`, or number of simulation trials) estimates of power for five initial values across this range. We then fit a scaled logistic curve to these five points, and identify where the curve crosses the desired power level. After fitting an initial curve, we iterate, repeatedly calculating power for the targeted point and using the result to update the logistic curve model. With each iteration we increase `tnum` to increase precision as we narrow in on the final answer; with each update to our estimated power curve, we weigh the collection of observations by their precisions (determined by corresponding `tnum` value). If a test point achieves the target power to within tolerance, we conduct an additional simulation check using a high number of replicates to verify the proposed answer is within a specified tolerance of the target power; if it is not, we continue the iterative search. The default tolerance is 1, so given a target power of 80, we stop when we find a MDES that gives an estimated power between 79 and 81. In practice, due to the monotonic nature of the logistic functional form, our algorithm generally converges fairly rapidly.

4.4 Package Validation

We completed extensive validation checks to ensure our power calculation procedures are correct. First, we compared our power estimates in scenarios with only one outcome, $M = 1$, to those from the PowerUpR! package. Without a multiple testing procedure adjustment, our estimates match. Second, in order to validate our estimates under multiplicity, we followed the full simulation approach outlined above, in Section~4.1. The simulation approach involves generating many iterations of full datasets according to the assumed design and model, calculating p-values, and calculating an empirical estimate of power. Using a binomial distribution we constructed Monte Carlo confidence intervals for the power estimates from the full simulation approach. Then, we validated that the PUMP estimates fall within these confidence intervals.

A more detailed explanation of the validation procedure can be found in the Appendix, and full validation code and results are in our github repository pump_validate. For some scenarios, we have some apparent discrepancies from PowerUp resulting from different modeling choices. For example, for certain models PowerUp assumes the intraclass correlation is zero, while we allow for nonzero values. When there are discrepancies, these are noted in the appendix.

5 The PUMP package

In this section, we illustrate how to use the PUMP package, using our example motivated by the Diplomas Now study. Given the study's design, we ask a natural initial question: What size of impact could we reasonably detect after using an MTP to adjust p-values to account for our multiple outcomes?

We mimic the planning process one might use for planning a study similar to Diplomas Now (e.g., if we were planning a replication trial in a slightly different context). To answer this question we therefore first have to decide on our experimental design and modeling approach. We also have to determine values for the associated design parameters that accompany these choices, as listed on Table <>. In the following sections we walk through these parameters (sample size, control variables, intraclass correlation coefficients, impact variation, and correlation of outcomes). We next calculate MDES for the resulting context and then determine how necessary sample sizes change depending on what kind of power we desire. We finally illustrate some sensitivity checks, looking at how MDES changes as a function of rho, the correlation of the test statistics.

5.1 Establishing needed design parameters

To conduct power, MDES, and sample size calculations, we first must specify the design, model, and level of statistical significance. We must also must specify parameters of the data generating distribution (e.g., expected relationships between covariates, outcomes, and units in the study) that match the design and model. All of these numbers have to be determined given resource limitations, or estimated using prior knowledge, pilot studies, or other sources of information. We also must specify the sample sizes at each level. Our experiment has students nested in schools nested in randomization blocks that are a function of school type and district. We next discuss selection of all needed design parameters and modeling choices. For further discussion of selecting these parameters see, for example, see Bloom (2006), Dong and Maynard (2013) and Porter (2018).

Analytic model. We first need to specify how we will analyze our data; this choice can also determine which design parameters we will need to specify. Following the original Diplomas Now report, we plan on using a multi-level model (a common choice for cluster randomized experiments, and especially common in education) with fixed effects at level three, a random intercept at level two, and a single treatment coefficient. We represent this model as "m3fc2rc." The "3fc" means we are including block fixed effects, and not modeling any treatment impact variation at level three. The "2rc" means random intercept and no modeled variation of treatment within each block (the "c" is for "constant"). We note that the Diplomas Now report authors call their model a "two-level" model, but this is not quite aligned with the language of this package. In particular, fixed effects included at level two are actually accounting for variation at level three; we therefore identify their model as a three-level model with fixed effects at level three.

Sample sizes. We assume equal size randomization blocks and schools, as is typical of most power analysis packages. For our context, this gives about three schools per randomization block; we can later do a sensitivity check where we increase and decrease this to see how power changes. The Diplomas Now report states there were 14,950 students, yielding around 258 students per school. Normally we would use the geometric means of schools per randomization block and students per school as our design parameters, but that information is not available in the report. We assume 50% of the schools are treated; our calculates will be approximate here in that we could not actually treat exactly 50% in small and odd-sized blocks.

Control variables. We next need values for the R^2 of the possible covariates. The report does not provide these quantities, but it does mention covariate adjustment in the presentation of the model. Given the types of outcomes we are working with, it is unlikely that there are highly predictive individual-level covariates, but our prior year school-average attendance measures are likely to be highly predictive of corresponding school-average outcomes. We thus set $R^2_1 = 0.1$ and $R^2_2 = 0.5$. We assume five covariates at level one and three at level two; this decision, especially for level one, usually does not matter much in practice, unless sample sizes are very small (these with sample size determine the degrees of freedom for our planned tests).

ICCs. We also need a measure of where variation occurs: the individual, the school, or the randomization

block level. We capture this with Intraclass Correlation Coefficients (ICCs), one for level two and one for level three. ICC measures divide overall variation in outcome across levels: e.g., do we see relatively homogenous students within schools that are quite different, or are the schools generally the same with substantial variation within them? We typically would obtain ICCs from pilot data or external reports on similar data. We here specify a level-two ICC of 0.05, and a level-three ICC of 0.40. We set a relatively high level three ICC to capture the hoped-for purpose of blocking as a means of isolating variation; in particular we might imagine attendance changes markedly between middle and high school as well as across schools.

Impact variation. We next need to specify the assumed degree of treatment impact variation. We allow treatment variation across school type and district by setting omega.3 to 0.50 (a substantial amount). While most power analyses would assume no variation, we do here for illustration.

Correlation of outcomes. We finally need to specify the number and relationship among our outcomes and associated test-statistics. For illustration, we select attendance as our outcome group. We assume we have five different attendance measures. The main decision regarding outcomes is the correlation of our test statistics. As a rough proxy, we use the correlation of the outcomes at the level of randomization; in our case this would be the correlation of school-average attendance within block. We believe the attendance measures would be fairly related, so we select `rho = 0.40` for all pairs of outcomes. This value is an estimate, and we strongly encourage exploration of different values of this correlation choice as a sensitivity check for any conducted analysis. Selecting a candidate rho is difficult, and will be new for those only familiar with power analyses of single outcomes; we need to more research in the field, both empirical and theoretical, to further guide this choice.

Once we have established initial values for all needed parameters, we first conduct a baseline power calculation, and then explore how MDES or other quantities change as these parameters change.

If the information were avilable, we could different values for the design parameters such as the R^2s and ICCs for each outcome, if we thought they had different characteristics; for simplicity we do not do this here. The `PUMP` package also allows specifying different pairwise correlations between the test statistics of the different outcomes via a matrix of ρs rather than a single ρ; also for simplicity, we do not do that here.

5.2 Calculating MDES

We now have an initial planned design, with a set number of schools and students. But is this a large enough experiment to reliably detect reasonably sized effects? To answer this question we calculate the minimal detectable effect size (MDES), given our planned analytic strategy, for our outcomes.

To identify the MDES of a given design we use the `pump_mdes` method, which conducts a search for a MDES that achieves a target level of power. The MDES depends on all the design parameters discussed above, but also depends on the type of power and target level of power we are interested in. For example, we can ask what size effect can we reliably detect on our first outcome, after multiplicity adjustment? Or, we might ask what size effects would we need across our five outcomes to reliably detect an impact on at least one of them? We do this by specifying the type (`power.definition`) and desired power (`target.power`).

Here, for example, we find the MDES if we want an 80% chance of detecting an impact on our first outcome when using the Holm procedure:

```
m <- pump_mdes(
            design = "d3.2_m3fc2rc", # choice of design and analysis strategy
            MTP = "Holm", # multiple testing procedure
            target.power = 0.80, # desired power
            power.definition = "D1indiv", # power type
            M = 5, # number of outcomes
            J = 3, # number of schools/block
            K = 21, # number RA blocks
            nbar = 258, # average number of students per school
```

```
            Tbar = 0.50, # prop Tx
            alpha = 0.05, # significance level
            numCovar.1 = 5, numCovar.2 = 3, # number of covariates per level
            R2.1 = 0.1, R2.2 = 0.7, # Explanatory power of covariates for each level
            ICC.2 = 0.05, ICC.3 = 0.4, # Intraclass correlation coefficients
            omega.3 = 0.50, # Amount of treatment variation at level 3.
            rho = 0.4 ) # how correlated outcomes are
```

The results are easily made into a nice table via `knitr`'s `kable()` command:

MTP	Adjusted.MDES	D1indiv.power
Holm	0.11	0.81

The answers `pump_mdes()` gives are approximate as we are calculating them via monte carlo simulation. To control accuracy, we can specify a tolerance (`tol`) of how close the estimated power needs to be to the desired target along with the number of iterations in the search sequence (via `start.tnum`, `max.tnum`, and `final.tnum`). The search will stop when the estimated power is within `tol` of `target.power`, as estimated by `final.tnum` iterations. Lower tolerance and higher `tnum` values will give more exact results (and take more computational time).

Changing the type of power is straightforward: for example, to identify the MDES for min-1 power (i.e., what effect do we have to assume across all observations such that we will find some significant result with 80% power?), we simply update our result with our new power definition:

```
m2 <- update( m, power.definition = "min1" )
kable(m2, digits=2)
```

MTP	Adjusted.MDES	min1.power
Holm	0.08	0.8

The `update()` method can replace any number of arguments of the prior call with new ones, making exploration of different scenarios very straightforward.[13] Our results show that if we just want to detect at least one outcome with 80% power, we can reliably detect an effect of size 0.084 (assuming all three outcomes have effects of at least that size):

When estimating power for multiple outcomes, it is important to consider cases where some of the outcomes in fact have null, or very small, effects, to hedge against circumstances such as one of the outcomes not being well measured. One way to to do this here is to assume two of our outcomes have no effect; the `numZero` parameter allows for this as follows:

```
m3 <- update( m2, numZero = 2 )
kable(m3, digits=2)
```

MTP	Adjusted.MDES	min1.power
Holm	0.09	0.8

The MDES goes up, as expected.

5.3 Determining necessary sample size

The MDES calculator tells us what we can detect given a specific design. We might instead want to ask how much larger our design would need to be in order to achieve a desired MDES. In particular, we might want to determine the needed number of students per school, the number of schools, or the number of blocks needed to detect an effect of a given size. The `pump_sample` method will search over any one of these, as requested.

[13] The '*update*()' method re-runs the underlying call of '$pump_mdes$()', '$pump_sample$()', or '$pump_power$()' with the revised set of design parameters. You can even change which call to use via the 'type' parameter.

Assuming we have three schools per block, we first calculate how many blocks we would need to achieve a MDES of 0.10 for min-1 power (this answer the question of how big of an experiment do we need in order to have an 80% chance of finding at least one outcome significant, if all outcomes had a true effect size of 0.10).

```
smp <- pump_sample(
  design = "d3.2_m3fc2rc",
  MTP = "Holm",
  typesample = "K",
  target.power = 0.80, power.definition = "min1", tol = 0.01,
  MDES = 0.10, M = 5, nbar = 258, J = 3,
  Tbar = 0.50, alpha = 0.05, numCovar.1 = 5, numCovar.2 = 3,
  R2.1 = 0.1, R2.2 = 0.7, ICC.2 = 0.05, ICC.3 = 0.40, rho = 0.4 )
```

MTP	Sample.type	Sample.size	min1.power
Holm	K	15	0.798625

We would need 16 blocks, rather than the originally specified 21, giving 48 total schools in our study, to achieve 80% min-1 power.

We recommend checking MDES and sample-size calculators as the estimation error combined with the stochastic search can give results a bit off the target in some cases. A check is easy to do; simply run the found design through `pump_power()`, which directly calculates power for a given scenario, to see if we recover our originally targeted power (we can use `update()` and set the type to `power` to pass all the design parameters automatically). When we do this, we can also increase the number of iterations to get more precise estimates of power, as well:

```
p_check <- update( smp, type="power", tnum = 100000 )
kable(p_check, digits = 2)
```

MTP	D1indiv	D2indiv	D3indiv	D4indiv	D5indiv	indiv.mean	min1	min2	min3	min4	complete
None	0.71	0.71	0.70	0.70	0.71	0.70					
Holm	0.52	0.53	0.52	0.53	0.53	0.53	0.81	0.63	0.5	0.4	0.35

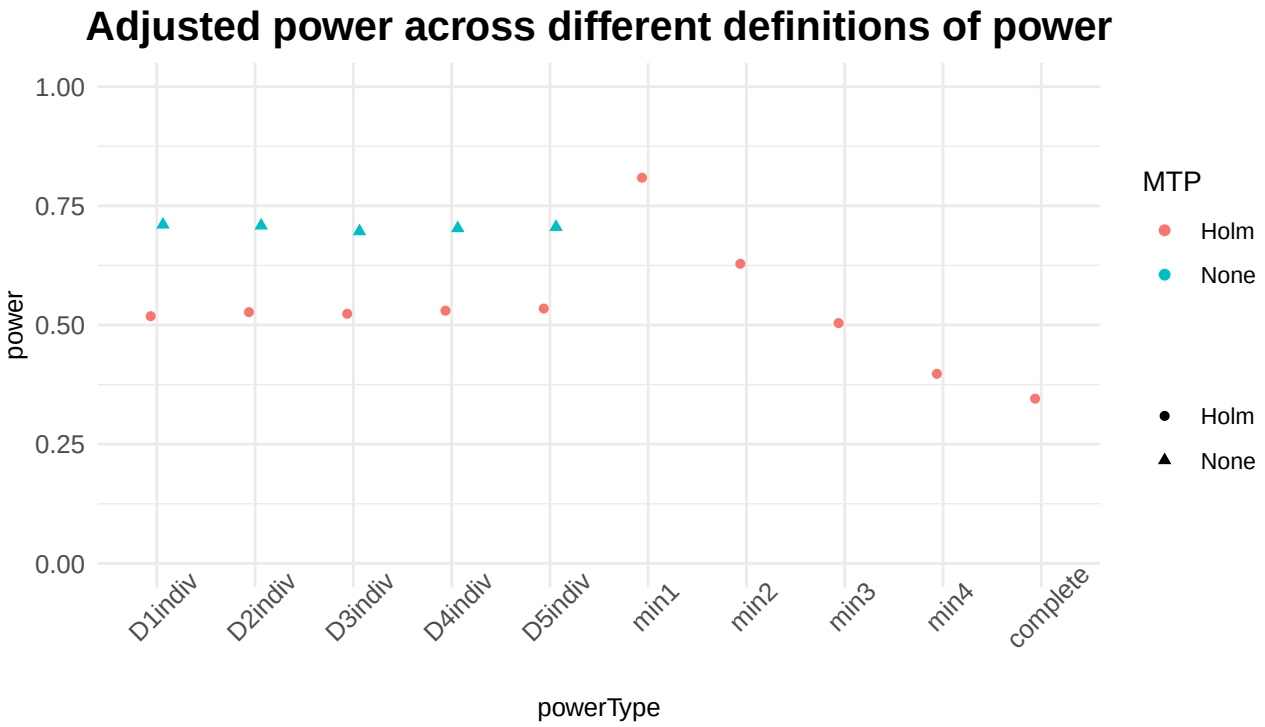
When calculating power directly, we get power for all the implemented definitions of power applicable to the design. In the above, the first five columns are the powers for rejecting each of the five outcomes—they are (up to simulation error) the same since we are assuming the same MDES and other design parameters for each. The `indiv.mean` is just the mean individual power across all outcomes. The first row is power without adjustment, and the second row has our power with the listed p-value adjustment.

The next columns show different multi-outcome definitions of power. In particular, `min1` and `min2` show the chance of rejecting at least one or two hypotheses, respectively. The `complete` column shows the power to reject all hypotheses; it is only defined if all outcomes are specified to have a non-zero effect.[14]

We can also plot the resulting power object, comparing the different MTPs and definitions of power.

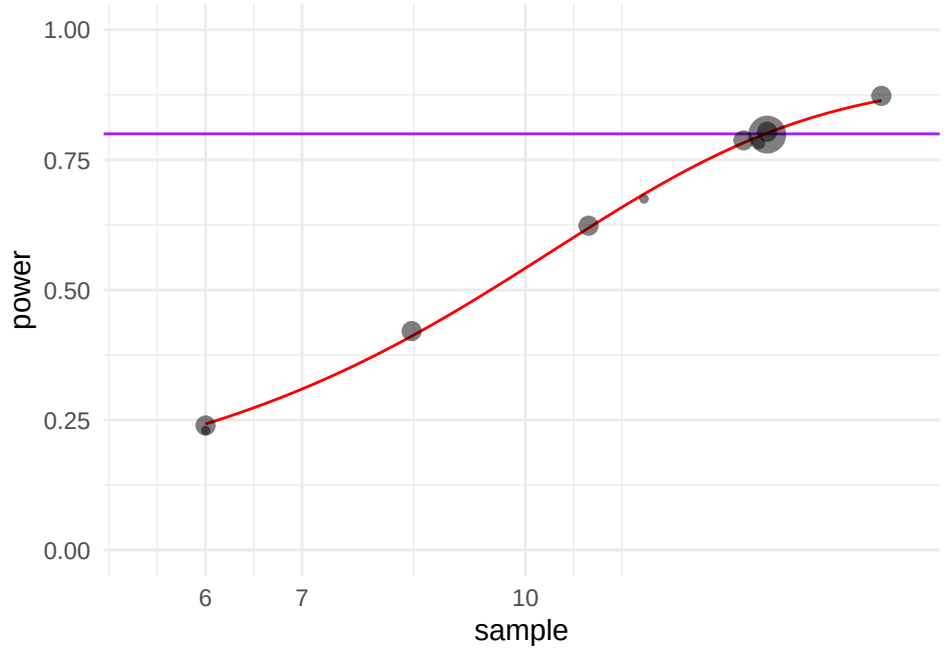
```
plot( p_check )
```

[14]The package does not show power for these without adjustment for multiple testing, as that power would be grossly inflated and meaningless.



We can look at a power curve to assess how sensitive power is to our level two sample size:[15]

```
plot_power_curve( smp )
```



[15]The points on the plots show the evaluated simulation trials, with larger points corresponding to more iterations and greater precision.

5.4 Comparing alternate approaches

The package works with a range of multiple testing procedures and a range of modeling options. In the next two sections we show how to compare these within a given experimental design.

5.4.1 Comparing adjustment procedures

It is easy to rerun the above using the Westfall-Young Stepdown procedure (this procedure is much more computationally intensive to run), or other procedures of interest. Alternatively, simply provide a list of procedures you wish to compare. If you provide a list, the package will re-run the power calculator for each item on the list; this can make the overall call computationally intensive. Here we obtain power for our scenario using Bonferroni, Holm and Westfall-Young adjustments: m

```
p2 <- update( p_check, MTP = c("Bonferroni", "Holm", "WY-SD") )
print( p2 )
#> power result: d3.2_m3fc2rc design with 5 outcomes
#>          MTP D1indiv D2indiv D3indiv D4indiv D5indiv indiv.mean  min1   min2
#>         None  0.7105  0.7145  0.7225  0.6925  0.7015     0.7083    NA     NA
#>   Bonferroni  0.4265  0.4285  0.4140  0.3830  0.4170     0.4138 0.799 0.5990
#>         Holm  0.5005  0.4995  0.5015  0.5140  0.5165     0.5064 0.794 0.6310
#>        WY-SD  0.4105  0.4095  0.4195  0.4145  0.4150     0.4138 0.718 0.5215
#>     min3   min4 complete
#>       NA     NA       NA
#>   0.3885 0.2060   0.3255
#>   0.4810 0.3645   0.3030
#>   0.3830 0.2680   0.3340
```

The more sophisticated (and less conservative) adjustment exploits the correlation in our outcomes (`rho = 0.4`) to provide higher individual power. Note, however, that we do not see elevated rates for min-1 power. Accounting for the correlation of the test statistics when adjusting p-values can drive some power (individual power) up, but on the flip side min-1 power can be driven down as the lack of independence between tests gives fewer chances for a significant result. See Porter (2018)*CITE porter* for further discussion; while the paper Porter (2017) focuses on the multisite randomized trial context, the lessons learned there apply to all designs as the only substantive differences between different design and modeling choices is in how we calculate the unadjusted distribution of their test statistics.

5.4.2 Comparing modeling choices

There are usually a range of modeling choices one might bring to a given experimental design. For example, for multisite experiments ("d2.1" designs), Miratrix et al. (*CITE Miratrix and Weiss*) identify 15 different estimation strategies. Different choices can imply different targeted estimands, which in turn can impact power. In particular, methods that target superpopulation averages vs. finite sample averages will generally have lower power if there is treatment impact variation.

In `PUMP` these choices are specified by different `design` arguments. For our context, for example, we could use a random effects model at level three instead of a fixed effects model, setting `design = "d3.2_m3rr2rc"` instead of `"d3.2_m3fc2rc"`; this would target a superpopulation average, viewing the blocks as a random sample, vs. a finite population where the blocks are considered fixed.

Random effects models allow for level three covariates, which we would need to specify via `numCovar.3` and `R2.3` to capture how many there are and how predictive they are:

```
p3 <- update( p_check, design="d3.2_m3rr2rc", numCovar.3 = 3, R2.3 = 0.40 )
print( p3 )
#> power result: d3.2_m3rr2rc design with 5 outcomes
```

```
#>   MTP D1indiv D2indiv D3indiv D4indiv D5indiv indiv.mean  min1  min2   min3
#>  None  0.6515  0.6530  0.6440  0.6460   0.643     0.6475    NA    NA     NA
#>  Holm  0.4105  0.4005  0.3905  0.4045   0.412     0.4036 0.685 0.495 0.3695
#>    min4 complete
#>      NA       NA
#>  0.2755   0.2575
```

5.5 Exploring sensitivity to design parameters

Given the above, we might wonder how power shifts if we change our design parameters. For some discussion of what parameters will affect power more generally, see Dong and Maynard (2013). For discussion of how design parameters can affect the overall power in the multiple testing context, especially with regards to the overall power measures such as min1 or complete power, see Porter (2018); the findings there are general, as they are a function of the final distribution of test statistics. The key insight into this approach is that power is a function of only a few summarizing elements: the individual-level standard errors, the degrees of freedom, and the correlation structure of the test statistics. Once we have these elements, regardless of the design, we can proceed.

Within the pump package we have two general ways of exploring design sensitivity. The first is with `update()`, which allows for quickly generating alternate scenarios that can each have very specific structure. To explore sensitivity to different design parameters more systematically, use the `grid()` functions, which calculate power, mdes, and sample size for all combinations of a set of passed parameter values. The main difference between the two approaches is the `update()` approach allows for different structures for the different outcomes. The `grid` approach is more limited in this regard, but is still a powerful tool for systematically exploring many possible combinations.

We first illustrate the `update()` approach, and then turn to illustrating `grid()` across three common areas of exploration: Intraclass Correlation Coefficients (ICCs), the correlation of test statistics, and the assumed number of non-zero effects. The last two are particularly important for multiple outcome contexts.

5.5.1 Exploring power with update()

Update allows for a quick change of some of the set of parameters used in a prior call; we saw `update()` used several times above. As a further example, here we examine what happens if the ICCs are more equally split across levels two and three:

```
p_b <- update( p_check, ICC.2 = 0.20, ICC.3 = 0.25 )
print( p_b )
#> power result: d3.2_m3fc2rc design with 5 outcomes
#>   MTP D1indiv D2indiv D3indiv D4indiv D5indiv indiv.mean  min1   min2  min3
#>  None  0.2475  0.2540  0.2450  0.2365  0.2525     0.2471    NA     NA    NA
#>  Holm  0.1115  0.0985  0.0955  0.0925  0.1040     0.1004 0.266 0.1275 0.062
#>   min4 complete
#>     NA       NA
#>  0.031   0.0275
```

We immediately see that our assumption of substantial variation in level three matters a great deal for power.

When calculating power for a given scenario, it is also easy to vary many of our design parameters by outcome. For example, if we thought we had better predictive covariates for our second outcome, we might try:

```
p_d = update( p_check,
              R2.1 = c( 0.1, 0.3, 0.1, 0.2, 0.2 ),
              R2.2 = c( 0.4, 0.8, 0.3, 0.2, 0.2 ),
              omega.3 = c( 0.2, 0.4, 0.3, 0.2, 0.2 ) )
```

```
print( p_d )
#> power result: d3.2_m3fc2rc design with 5 outcomes
#>   MTP D1indiv D2indiv D3indiv D4indiv D5indiv indiv.mean  min1  min2   min3
#>  None  0.4265  0.8490  0.3985   0.343   0.355     0.4744    NA    NA     NA
#>  Holm  0.2385  0.6545  0.2050   0.175   0.190     0.2926 0.717 0.361 0.2035
#>    min4 complete
#>      NA       NA
#>  0.1135   0.0845
```

Notice how the individual powers are heavily impacted. The min-d powers naturally take the varying outcomes into account as we are calculating a joint distribution of test statistics that will have the correct marginal distributions based on these different design parameter values.

After several `update()`s, we may lose track of where we are; to find out, we can always check details with `print_design()` or `summary()`:

```
summary(p_d)
#> power result: d3.2_m3fc2rc design with 5 outcomes
#>   MDES vector: 0.1, 0.1, 0.1, 0.1, 0.1
#>   nbar: 258  J: 3    K: 15   Tbar: 0.5
#>   alpha: 0.05
#>   Level:
#>     1: R2: 0.1 / 0.3 / 0.1 / 0.2 / 0.2 (5 covariate)
#>     2: R2: 0.4 / 0.8 / 0.3 / 0.2 / 0.2 (3 covariate) ICC: 0.05   omega: 0
#>     3:   fixed effects    rho = 0.4
#>   MTP D1indiv D2indiv D3indiv D4indiv D5indiv indiv.mean  min1  min2   min3
#>  None  0.4265  0.8490  0.3985   0.343   0.355     0.4744    NA    NA     NA
#>  Holm  0.2385  0.6545  0.2050   0.175   0.190     0.2926 0.717 0.361 0.2035
#>    min4 complete
#>      NA       NA
#>  0.1135   0.0845
#>  (tnum = 2000)
```

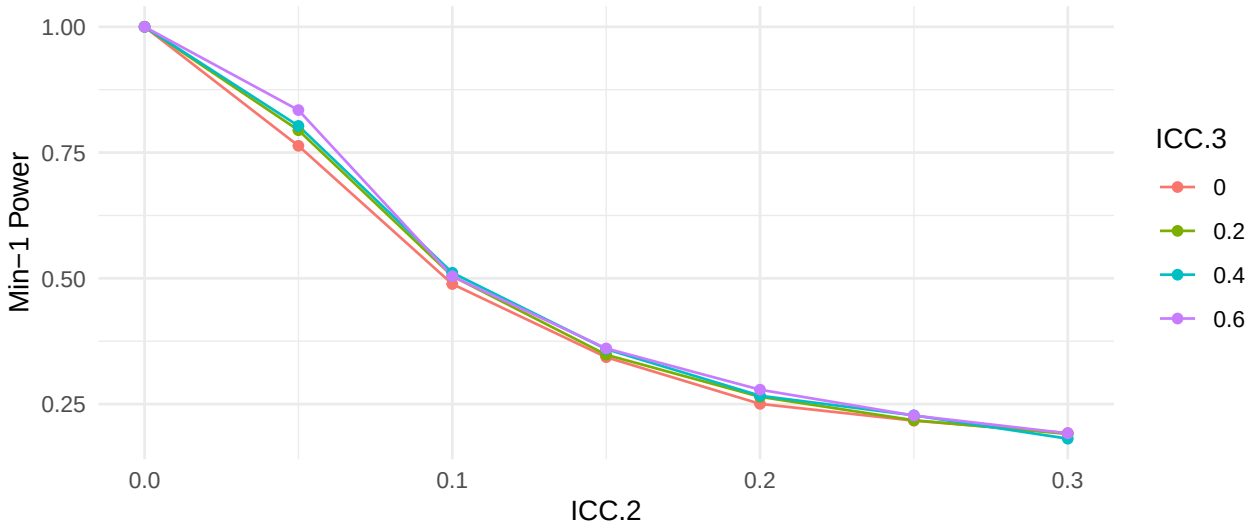
Using update allows for targeted comparison of major choices, but if we are interested in how power changes across a range of options, we can do this more systematically with the `grid()` functions, as we do next.

5.5.2 Exploring the impact of the ICC

We above saw that the ICC does impact power considerably. We next extend this evaluation by exploring a range of options for both level two and three ICCs, so we can assess whether our power is sufficient across a set of plausible values. The `update_grid()` call makes this straightforward: we pass our baseline scenario along with lists of parameters to additionally explore:

```
grid <- update_grid( p_check,
            ICC.2 = seq( 0, 0.3, 0.05 ),
            ICC.3 = seq( 0, 0.60, 0.20 ),
            tnum = 5000 )

grid$ICC.3 = as.factor( grid$ICC.3 )
grid = filter( grid, MTP == "Holm" )
ggplot( grid, aes( ICC.2, min1, group = ICC.3, col = ICC.3 ) ) +
    geom_line() + geom_point() +
    labs( y = "Min-1 Power" )
```



We see that higher ICC.2 radically reduces power to detect anything and ICC.3 does little. To understand why, we turn to our standard error formula for this design and model:

$$SE(\hat{\tau}) = \sqrt{\frac{\text{ICC}_2(1-R_2^2)}{\bar{T}(1-\bar{T})JK} + \frac{(1-\text{ICC}_2-\text{ICC}_3)(1-R_1^2)}{\bar{T}(1-\bar{T})JK\bar{n}}}.$$

In the above, the $\bar{n} = 258$ students per group makes the second term very small compared to the first regardless of the ICC.3 value. The first term, however, is a direct scaling of ICC.2; changing it will change the standard error, and therefore power, a lot. All provided designs and models implemented in the package are discussed, along with corresponding formula such as these, in our technical supplement accompanying this paper and package.

For grid searches we recommend reducing the number of permutations (here to 5000 via `tnum`) to speed up computation. As `tnum` shrinks, we will get increasingly rough estimates of power, but even these rough estimates can help us determine trends.

The `grid()` functions provide easy and direct ways of exploring how power changes as a function of the design parameters. We note, however, that in order to keep syntax simple, they do not allow different design parameters, including MDES, by outcome. This is to keep package syntax simpler. When faced with contexts where it is believed that these parameters do vary, we recommend using average values for the broader searches, and then double-checking a small set of potential final designs with the `pump_power()` method.

5.5.3 Exploring the impact of rho

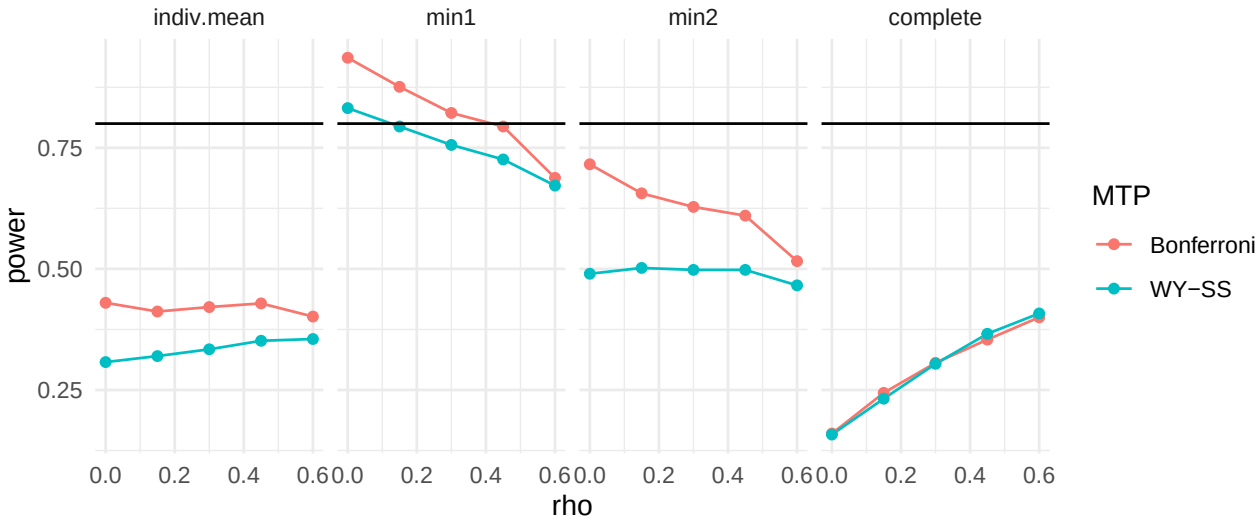
The correlation of test statistics, ρ, is a critical parameter for how power will play out across the multiple tests. For example, with Westfall-Young, we saw that the correlation can improve our individual power, as compared to Bonferroni. We might not know what will happen to min-2 power, however: on one hand, correlated statistics make individual adjustment less severe, and on the other correlation means we succeed or fail all together. We can explore this relatively easily by letting `rho` vary as so:

```
grid <- update_grid( p_check,
            MTP = c( "Bonferroni", "WY-SS" ),
            rho = c( 0, 0.15, 0.3, 0.45, 0.6 ),
            tnum = 500,
            B = 10000 )
```

We then plot our results

```
gridL = filter( grid, MTP != "None" ) %>%
  pivot_longer( cols=c(indiv.mean, min1, min2, complete),
                names_to="definition", values_to="power" ) %>%
  mutate( definition = factor( definition,
                               levels = c("indiv.mean", "min1", "min2", "complete" ) ) )

ggplot( gridL, aes( rho, power, col=MTP ) ) +
  facet_grid( . ~ definition ) +
  geom_line() + geom_point() +
  geom_hline( yintercept =0.80 ) + theme_minimal()
```



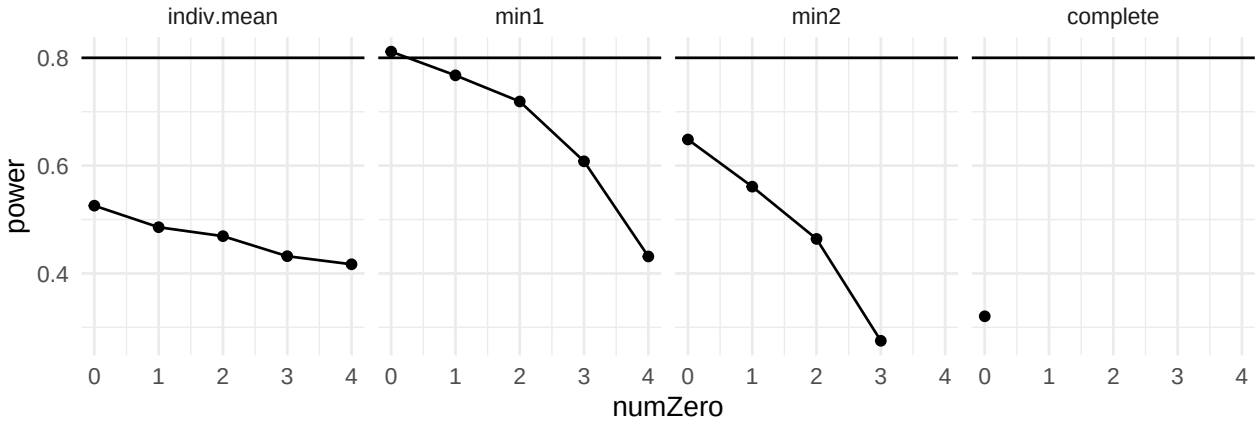
First, we see the benefit of the Westfall-Young single-step procedure is minimal, as compared to Bonferroni. Second, the impact on individual adjustment is flat, as anticipated. Third, across a very broad range of rho, we maintain good min-1 power. Complete power climbs as correlation increases, and min-2 power is generally unchanged.

5.5.4 Exploring the impact of null outcomes

We finally explore varying the number of outcomes with no effects. This exploration is an important way to hedge a design against the possibility that some number of the identified outcomes are measured poorly, or are simply not impacted by treatment. We use a grid search, varying the number of outcomes that have no treatment impact via the `numZero` design parameter:

```
grid <- update_grid( p_check,
            numZero = 0:4,
            M = 5 )
```

We then can make a plot as we did above:



There are other ways of exploring the impact of weak or null effects on some outcomes. In particular, the `pump_power()` and `pump_sample()` methods allow the researcher to provide an MDES vector with different values for each outcome, including 0s for some outcomes. The `grid()` functions, by contrast, take a single MDES value for the non-null outcomes, with a separate specification of how many of the outcomes are 0. (This single value plus `numZero` parameter also works with `pump_power()` if desired.)

6 Conclusion

We introduce the power under multiplicity project (PUMP) package, which estimates power for multi-level randomized control trials with multiple outcomes. PUMP allows users to estimate power, MDES, and sample size requirements for a wide variety of commonly used RCT designs and models across different definitions of power and applying different MTPs. The functionality of PUMP fills an important gap, as existing tools do not allow researchers to conduct power, MDES or sample size calculations when applying a MTP.

The main advantage of the PUMP package is to provide easily accessible estimation procedures so that users can properly account for power when making adjustments for multiple hypothesis testing. However, one of the additional strengths of the package is the ease with which a user can explore the impact of different designs, models, and assumptions on power, MDES or sample size. Even if a user is only interested in a single outcome, PUMP provides useful functionality for more robust power calculations. A user can and should try a range of parameter values to determine the sensitivity of the power of their study to different assumptions; this package simplifies that process.

7 References

Bang, Jung, H. 2005. "Sample Size Calculation for Simulation-Based Mulitple-Testing Procedures." Journal Article. *Journal of Biopharmaceutical Statistics* 15: 957–67.

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. 2015. "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software* 67 (1): 1–48. https://doi.org/10.18637/jss.v067.i01.

Berger, Roger L. 1982. "Multiparameter Hypothesis Testing and Acceptance Sampling." Journal Article. *Technometrics* 24 (4): 295–300.

Berger, Roger L., and Jason C. Hsu. 1996. "Bioequivalence Trials, Intersection–Union Tests and Equivalence Confidence Sets." Journal Article. *Statistical Science* 11 (4): 283–319.

Bloom, Howard S. 2006. "The Core Analytics of Randomized Experiments for Social Research." Government Document. MDRC.

Chen, Luo, J. 2011. "On Power and Sample Size Computation for Multiple Testing Procedures." Journal Article. *Computational Statistics and Data Analysis* 55: 110–22.

Corrin W., Rosen, Sepanik S. 2016. "Addressing Early Warning Indicators: Interim Impact Findings from the Investing in Innovation (I3) Evaluation of Diplomas Now." Government Document. MDRC.

Deng, Xutao, Jun Xu, and Charles Wang. 2008. "Improving the Power for Detecting Overlapping Genes from Multiple DNA Microarray-Derived Gene Lists." Journal Article. *BMC Bioinformatics* 9.

Dong, Nianbo, and Rebecca Maynard. 2013. "PowerUP!: A Tool for Calculating Minimum Detectable Effect Sizes and Minimum Required Sample Sizes for Experimental and Quasi-Experimental Design Studies." Journal Article. *Journal of Research on Educational Effectiveness* 6 (1): 24–67.

Dudoit, Shaffer, S. 2003. "Multiple Hypothesis Testing in Microarray Experiments." Journal Article. *Statistical Science* 18 (1): 71–103.

Dunn, Olive Jean. 1959. "Estimation of the Medians for Dependent Variables." Journal Article, 192–97. https://doi.org/10.1214/aoms/1177706374.

———. 1961. "Multiple Comparisons Among Means." Journal Article. *Journal of the American Statistical Association* 56 (293): 52–64. https://doi.org/10.1080/01621459.1961.10482090.

Hedges, Larry V., and Christopher Rhoads. 2010. "Statistical Power Analysis in Education Research." Report. National Center for Special Education Research. http://www.eric.ed.gov/ERICWebPortal/detail?accno=ED509387.

Holm, S. 1979. "A Simple Sequentially Rejective Multiple Test Procedure." Journal Article. *Scand. J. Statist.* 6 (2): 65–70.

Porter, Kristin E. 2018. "Statistical Power in Evaluations That Investigate Effects on Multiple Outcomes: A Guide for Researchers." Journal Article. *Journal of Research on Educational Effectiveness* 11: 267–95.

Raudenbush, S. W., J. Spybrook, R. Congdon, X. Liu, A. Martinez, H. Bloom, and C Hill. 2011. "Optimal Design Plus Empirical Evidence (Version 3.0)." Report. http://wtgrantfoundation.org/resource/optimal-design-with-empirical-information-od.

Schochet, Peter Z. 2008. "Guidelines for Multiple Testing in Impact Evaluations of Educational Interventions. Final Report." Report. Mathematica Policy Research, Inc. P.O. Box 2393, Princeton, NJ 08543-2393. Tel: 609-799-3535; Fax: 609-799-0005; e-mail: info@mathematica-mpr.com; Web site: http://www.mathematica-mpr.com/publications/. http://www.eric.ed.gov/ERICWebPortal/contentdelivery/servlet/ERICServlet?accno=ED502199.

Senn, Stephen, and Frank Bretz. 2007. "Power and Sample Size When Multiple Endpoints Are Considered." Journal Article. *Pharmaceutical Statistics* 6: 161–70. https://doi.org/10.1002/pst.301.

Shaffer, Juliet Popper. 1995. "Multiple Hypothesis Testing." Journal Article. *Annual Review of Psychology* 46 (1): 561–84.

Spybrook, Jessica, H. S. Bloom, Richard Congdon, Carolyn J. Hill, Andres Martinez, and Stephen W. Raudenbush. 2011. "Optimal Design Plus Empirical Evidence: Documentation for the 'Optimal Design' Software Version 3.0." Report. http://wtgrantfoundation.org/resource/optimal-design-with-empirical-information-od.

Tukey, J. W. 1953. "The Problem of Multiple Comparisions." Report. Princeton University.

Westfall, Peter H, and S Stanley Young. 1993. *Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment*. Book. Vol. 279. John Wiley & Sons.

Westfall, Tobias, Peter H, and R. D. Wolfinger. 2011. *Multiple Comparisons and Multiple Tests Using SAS*. Book. The SAS Institute.

8 Appendix: Validation

This appendix discusses our work to validate that the power estimation methods work as intended.

We compare three different methods of estimating power:

- PUMP
- PowerUpR (only comparable for D1individual, unadjusted power)
- Monte Carlo Simulations

We compare point estimates of power from PUMP and PowerUpR for D1 individual, unadjusted power estimates. For all other types of power definitions and adjustments, we are only able to compare PUMP to the estimated power from Monte Carlo simulations. The simulations produce a 95% confidence interval for power, and we check that the PUMP estimate is within the confidence interval.

8.1 Monte Carlo Simulations

The main work of this validation step was to design monte carlo simulations in order to estimate power. In order to estimate power by simulation, we follow the following steps.

For iteration $s = 1, \dots S$:

1. Generate simulated data according to the assumed data generating process (DGP)
2. Generate simulated treatment assignment
3. Calculate p-value given simulated data, treatment assignment, and assumed model

At the end, a 95% confidence interval for power is calculated, assuming a conservative standard error estimate of $\sqrt{0.25/S}$.

We also validate MDES and sample size calculations. For MDES, we choose one default scenario for each design and model, then input the already-calculated D1 individual power and see if the output MDES is the same as the original input MDES. Similarly, for sample size validation, we input the already-calculated D1 individual power and see if the output sample size (either J or K depending on design) is the same as the original sample size.

8.2 Simulation parameters

In order to validate that the method works in a wide range of scenarios, which vary the following parameters.

Parameters that vary:

Parameter	Default	Comparison values
school size $\bar{n}$	50	75, 100
R^2	0	0.6
ρ	0.5	0.2, 0.8
ATE (ES) true positives	(0.125, 0.125, 0.125)	(0.125, 0, 0)
ICC	0.2	0.7
ω	0.1	0.8

We do not vary:

- $M = 3$
- J and K are fixed for each scenario
- Scalar grand mean Ξ_0
- Correlations between school random effects and impacts κ
- ρ informs all correlations; we keep the same correlation between covariates, residuals, impacts, random effects for all levels and across all outcomes