

# Optimal Study Design in Cluster Randomized Trials with Budget and Data Parameter Considerations: A Simulation Study

Maia Lindner-Liaw

2024-12-03

## Abstract

Cluster randomized trials (CRTs), where clusters instead of observations are assigned treatment, are often used when researchers want to avoid treatment contamination or have a small study population to draw from. In a CRT, the study design depends on both the number of clusters and the number of observations in each cluster. Finding the optimal balance of cluster number and size is made more complex when there are additional budget considerations. Therefore, determining the optimal design given different cost ratios and data conditions is a useful question to answer. We conduct a simulation study, generating data from normal and Poisson hierarchical models with varying data parameters and cost ratios. For each setting, we estimate the treatment effect and evaluate different study designs using the variance of the treatment estimator. The optimal study design has the lowest variance, and we test how that design changes under the different settings. We found that the optimal study design will generally have a small number of clusters with many observations when the cost ratio increases. Some of the data parameters did have an effect on the optimal design, but the effect decreased as the cost ratio increased. The effects of the parameters were not always consistent across the different distributions.

## Introduction

Cluster randomized trials (CRTs) are a type of randomized controlled trial where instead of individuals being randomly assigned the treatment, clusters of subjects are randomly assigned to the treatment (Hurley 2020). Some examples are randomizing schools to treatment in a children's health promotion study or randomizing hospitals to a new medical treatment or method (Puffer et al 2005). In these examples, the schools and the hospitals are the clusters, so the students and patients at each location have the same treatment assignment. A primary reason to use a CRT design over an RCT design is to avoid contamination, which is when the treatment of one subject is influenced by the treatment of another (Cook et al 2016). For example, if doctors in a hospital are treating patients with both the treatment and control methods, there could be differences in treatment compared to hospitals where only one method is used. This contamination would then cause dilution bias where the treatment effect is underestimated (Puffer et al 2005). Cluster trials may also be useful in the case when the potential subject pool is limited, in which case a single person can be taken as a cluster. The outcome is measured multiple times on each subject, either successively or over time.

When using a CRT design, sample size considerations do not solely depend on total number of subjects or observations. The number of clusters to randomize must be chosen, but also the number of observations in each cluster. This is further complicated in the presence of budgetary constraints, which also affect the choice of design. Given a budget  $B$ , we are interested in finding the optimal allocation of the number of clusters,  $g$ , and the number of samples from each cluster,  $r$ . The two costs associated with sampling clusters is the cost of the first sample from a cluster,  $c_1$ , and the cost of any additional samples from the cluster,  $c_2$ . In general, we have  $c_2 < c_1$ , which we assume is true in this case. Although samples within clusters are less expensive, the observations may be correlated and must have the same treatment, which affects the analysis. The optimal  $g$  and  $r$  will be the combination that best captures the true relationship in the data while remaining within budget.

In practice, we do not know the true value of the parameters of interest, so we cannot thoroughly evaluate the performance of the estimator. Therefore, we will conduct a simulation study to assess possible study designs and find the optimal conditions. We will also assess how data features and the relative costs  $c_1/c_2$  affect the optimal design.

## Simulation Study

### Aims

As previously mentioned, the aim of this simulation study is to determine the optimal study design for a CRT given budget constraints. We are also interested in how the data generating parameters and the relative costs  $c_1/c_2$  are related and what their effects are on the optimal design.

### Data Generation

We begin the study by assuming a normally distributed outcome  $Y$ , and a single covariate  $X_i$  indicating the binary treatment assignment for the  $i$ th cluster. The outcome  $Y_{ij}$  will follow the hierarchical structure

$$\begin{aligned}\mu_{i0} &= \alpha + \beta X_i, \\ \mu_i | \epsilon_i &= \mu_{i0} + \epsilon_i, \quad \epsilon_i \sim N(0, \gamma^2), \\ Y_{ij} | \mu_i &= \mu_i + e_{ij}, \quad e_{ij} \stackrel{iid}{\sim} N(0, \sigma^2),\end{aligned}$$

where  $\mu_{i0}$  is the fixed effect and  $\mu_i$  is the random intercept for the  $i$ th cluster.  $i$  ranges from 1 to  $g$ , and  $j$  ranges from 1 to  $r$ , so  $Y_{ij}$  is the  $j$ th repeated observation from the  $i$ th cluster. We can vary  $\alpha$ ,  $\beta$ ,  $\gamma$ , and  $\sigma$  to assess different data settings.

We will also look at the case when  $Y$  has a Poisson distribution with mean  $\mu_i$ . This also has a hierarchical form of

$$\begin{aligned}\log(\mu_i) &\sim N(\alpha + \beta X_i, \gamma^2), \\ Y_{ij} | \mu_i &\sim \text{Poisson}(\mu_i),\end{aligned}$$

where  $X_i$  and  $\mu_i$  are specified the same as before.

### Estimand

Given the data, we are interested in estimating the treatment effect  $\beta$ , which is found in the fixed effects of the hierarchical model.

### Methods

For a given budget,  $B$ , we calculate the total cost of the study as

$$B \geq g(c_1 + (r - 1)c_2),$$

and therefore given  $g$  clusters, we can directly calculate the  $r$  for which the entire budget is used as

$$r \leq \frac{B}{gc_2} - \frac{c_1}{c_2} + 1.$$

We will fix  $B = 5000$ , as increasing the budget will scale up the optimal  $g$  and  $r$  values, but should not change the behavior of the optimal design. Taking  $c_1 = 100$  is reasonable, as this allows for a maximum number of clusters of  $B/c_1 = 50$ , which is a reasonable number for a study of this size. Because we are interested in the effect of  $c_1/c_2$  on the optimal design, we will look at three different cases: when the ratio is close to 1, when the ratio is moderately large, and when the ratio is very large. We will take  $c_2$  to be 95, 20, and 5, which results in ratios of 1.05, 5, and 20, respectively.

Because we always want the same number of treatment and control subjects,  $g$  will always be even. Our regression models may have a hard time estimating the treatment effect if there is only one observation in each treatment group, so we take the minimum  $g$  value to be 4. Therefore, our  $g$  values to test are all even numbers between 4 and 50, and we calculate the corresponding  $r$  values using the above equation and the fixed values for  $B$ ,  $c_1$ , and  $c_2$ .

For each of our  $c_1/c_2$  ratio cases, we will simulate our data using the  $(g, r)$  pairs and varying values of  $\alpha$ ,  $\beta$ , and  $\gamma$ , setting the seed for reproducibility. We will not vary  $\sigma$  because in clustered data, the ratio of  $\gamma$  to the total variance  $\gamma + \sigma$  is most important, which can be varied by changing only  $\gamma$ . We then fit a linear mixed effects model with a random intercept term to estimate the treatment effect  $\beta$ . When  $r = 1$ , the model simplifies to a simple linear regression and GLM for the normal and poisson cases, and no random intercept is estimated. For each data setting, we will repeat the data simulation and model fitting 300 times to obtain performance measures for the estimator  $\hat{\beta}$ . We choose 300 iterations to balance sample size with computational time.

## Performance Measures

When the target of a simulation study is an estimand, relevant performance measures are bias, mean-squared error (MSE), coverage, and empirical standard error or variance. We know  $\hat{\beta}$  to be an unbiased and consistent estimator from properties of the mixed effects model, so bias and MSE are not appropriate performance measures. We are also not interested in significance, so we also don't want to use coverage. That leaves the variance of the estimator as our performance measure. For this study, a small variance is desirable, and the optimal  $(g, r)$  pair will minimize the variance of  $\hat{\beta}$  while remaining within the budget.

Running each data setting 300 times will give us 300  $\hat{\beta}$  estimates. We estimate the variance of the estimator  $\widehat{Var}(\hat{\beta})$  as the variance of the 300 estimates. After obtaining the variance estimates, we compare them across the  $(g, r)$  pairs to find the smallest variance, which is then the optimal study design. We will compare how this optimal design changes under the different ratio cases as well as the varying data generating parameters.

## Results and Discussion

### The Normal Case

**Low Cost Ratio** The first case we investigate is when the cost ratio  $c_1/c_2$  is close to 1. Recall we take  $c_1 = 100$  and  $c_2 = 95$ , for a cost ratio of 1.05. In Figure 1, we see the  $\hat{\beta}$  variance trends as  $g$  increases, varying the data parameters. In all panels, we see a trend of slight increasing variance for  $g$  from 1 to 20, followed by a spike at about 25 and a dropoff until the max  $g$  of 50. The trends are very similar across the varying  $\alpha$  and  $\beta$  values, although the optimal  $(g, r)$  pairs are different. For varying  $\alpha$ , the optimal  $g$  does not increase or decrease with increasing  $\alpha$ , suggesting that the differences in the optimal design are due to random chance. For varying  $\beta$ , we do see that the optimal number of clusters matches the increase in  $\beta$  value, with an optimal pair of (8, 6) for  $\beta = 0.5$  and (50, 1) for  $\beta = 10$ . When the true treatment effect is large, it will be easier to detect in a model, but when it is small or close to 0, it will be harder to detect due to noise in the data. In this case,  $\beta$  makes up part of the fixed mean of each cluster and when it is large, the mean of each cluster is easy to estimate. Therefore, multiple observations from each cluster are not necessary and it is better to sample more clusters. When  $\beta$  is small, the true mean will be harder to detect, resulting in more samples from each cluster to get a good estimate.

When varying  $\gamma$ , we see a systematic difference in variance as the variability increases. A larger  $\gamma$  value results in overall higher  $\widehat{Var}(\hat{\beta})$ . For both larger  $\gamma$  values, the optimal design has the minimum number of cluster, and the smallest  $\gamma$  has the maximum number of clusters as the optimal  $g$ . This makes sense because as  $\gamma$ , the within-cluster variance, goes towards 0, you gain less and less information from repeatedly sampling from a cluster. The between-cluster variance becomes a larger proportion of the total variance, making it better to sample from different clusters. Conversely, when the within-cluster variance is proportionally larger, you gain more from sampling within a cluster, leading to smaller  $g$  and larger  $r$ .

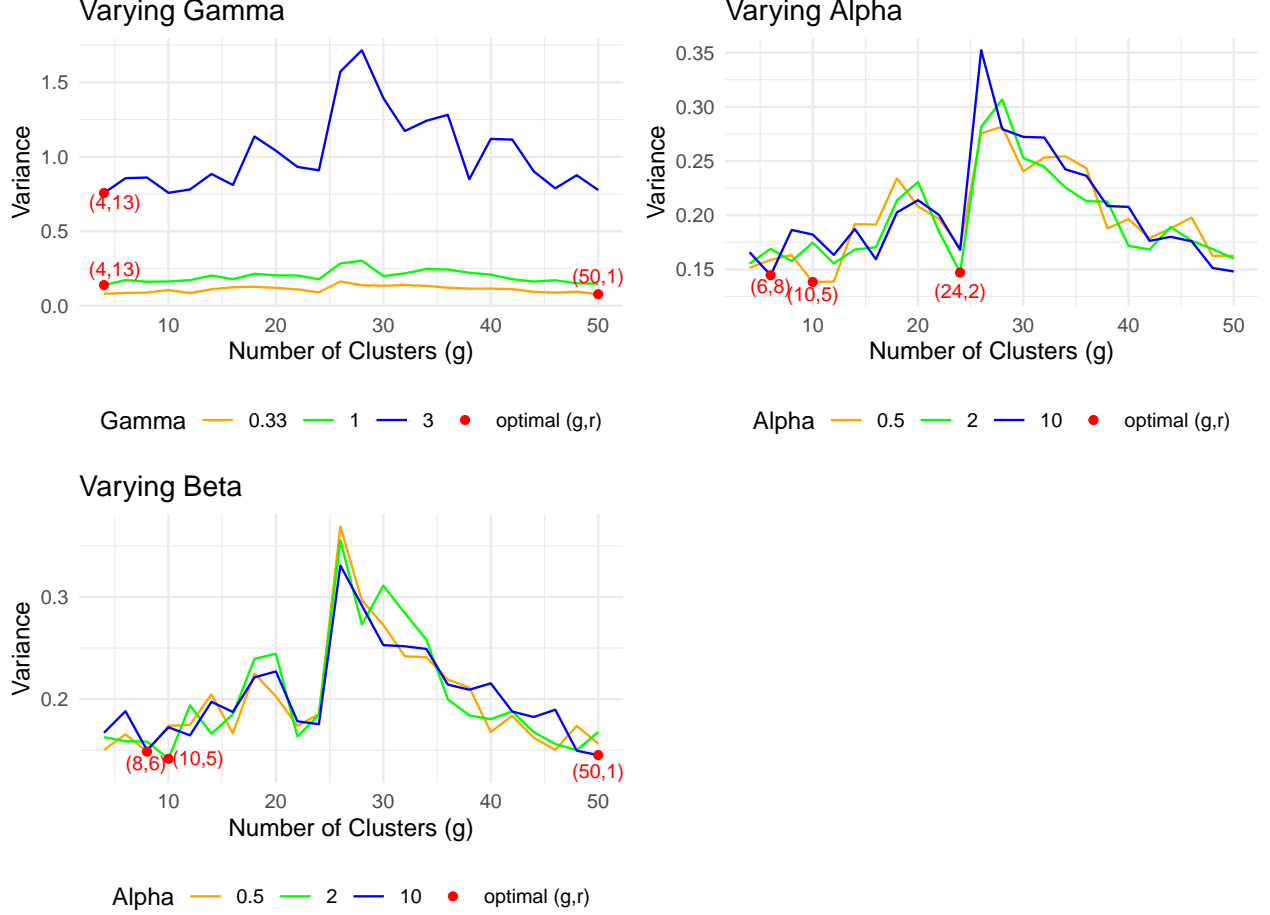


Figure 1: Variance of Beta Estimator for Varying  $g$  and Data Parameters under Low Cost Ratio

**Moderate Cost Ratio** When  $c_2 = 20$ , the cost ratio is 5. Under the moderate cost ratio, the trends in the  $\widehat{Var}(\hat{\beta})$  change to increase as the number of clusters increases as seen in Figure 2. When varying all parameters, the optimal design always has small  $g$  with larger  $r$ , unlike in the low cost ratio case, where some optimal designs had very large  $g$ . This trend makes sense, because with a larger cost ratio, it becomes cheaper to take a greater number of samples from each cluster than to sample from a new cluster. The effect of the varying data parameters on the optimal design is also less clear. For  $\gamma$ , the largest and smallest values have the same optimal pair. Although the smallest  $\alpha$  has the largest optimal  $g$ , the variances for the three optimal designs are within 0.007 of each other, showing the designed are virtually exchangeable. The largest  $\beta$  still has the largest optimal  $g$ , although the difference is now much smaller than in the previous case.

**High Cost Ratio** To get a high cost ratio, we took  $c_2 = 5$ , which results in a ratio of 20. The behavior of the variance of  $\hat{\beta}$  under this scenario is similar to under moderate cost ratio, with increasing variance as  $g$  grows, spiking up around  $g = 40$ . Table 1 summarizes the optimal designs for each case. We see that the optimal designs are all similar for the different parameters and the corresponding variances are very small. For  $\alpha$  and  $\beta$ , the variances are again very close together, indicating there is not much difference in performance across the 3 optimal design choices. For  $\gamma$  we see that the variance is still largest when  $\gamma = 3$ , and  $\gamma = 0.33$  tends toward a larger number of clusters, but the differences are still small.

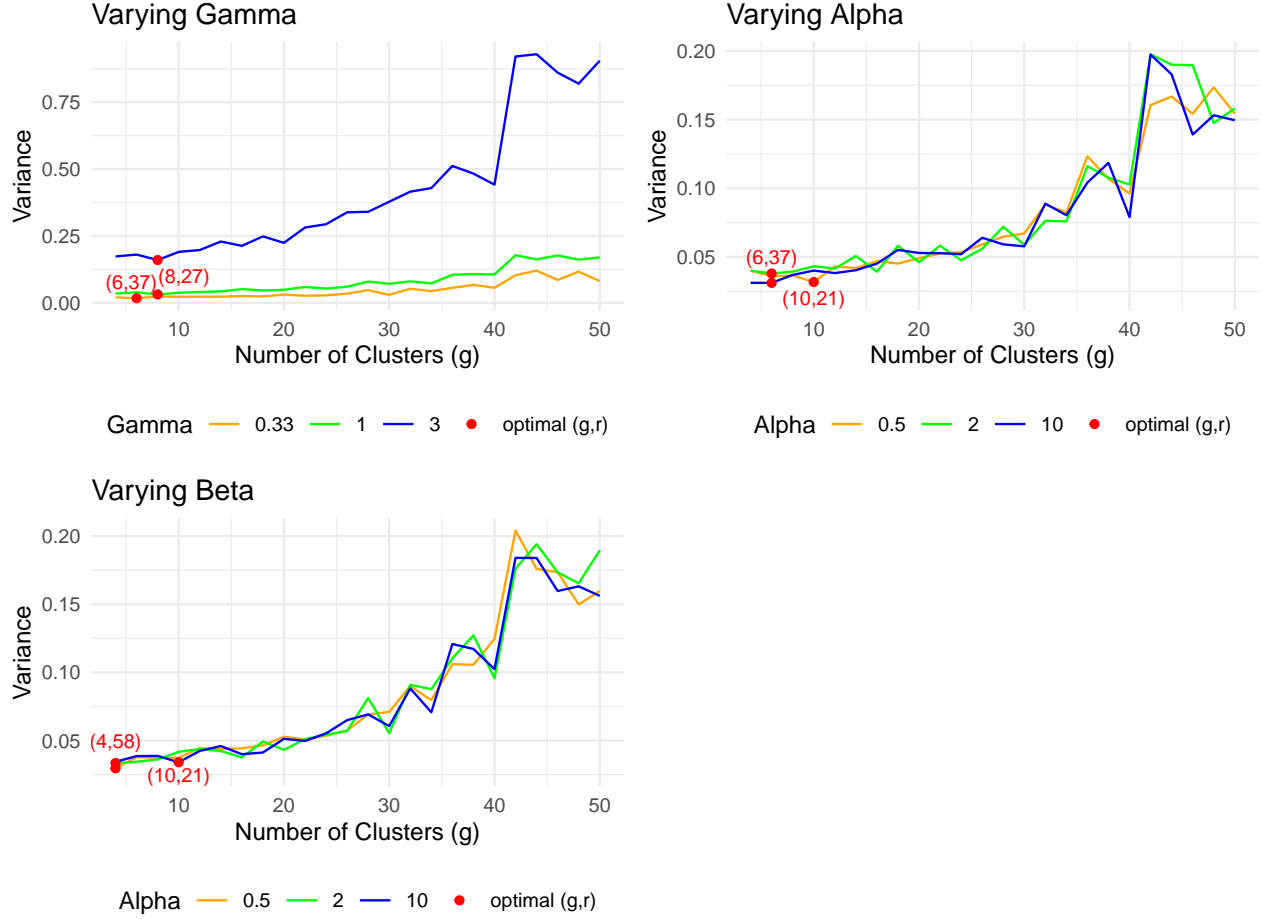


Figure 2: Variance of Beta Estimator for Varying  $g$  and Data Parameters Under Moderate Cost Ratio

Table 1: Optimal Designs For Varying Parameters under High Cost Ratio

value	$g$	$r$	variance
<b>alpha</b>			
0.5	6	147	0.009
2	4	231	0.008
10	6	147	0.008
<b>beta</b>			
0.5	4	231	0.007
2	6	147	0.009
10	4	231	0.008
<b>gamma</b>			
1	4	231	0.008
3	6	147	0.043
0.33	10	81	0.004

### The Poisson Case

For the Poisson case, we will use the same cost ratios as in the normal case.

**Low Cost Ratio** Under the Poisson distribution, we see similar trends in Figure 3 in the estimator variance as in the low cost ratio case under the normal. The variances peak around  $g = 25$  and decrease afterwards, but we actually see a slight decreasing trend from  $g = 1$  to  $g = 25$ , most obviously when  $\gamma$  is large. The  $\gamma$  value affects the optimal design in the same way as before, but to a lesser extreme. Under the normal, the study designs were at the minimum and maximum  $g$  values, but under the Poisson, it is more moderate. As  $\alpha$  increases, the optimal  $g$  decreases, unlike under the normal. This makes sense because in a Poisson distribution, the mean and variance are equal. Therefore, when the mean increases,  $\gamma$  becomes proportionally smaller, leading to a larger optimal number of clusters. As  $\beta$  changes, the optimal design does not change in the same way, suggesting no clear relationship.

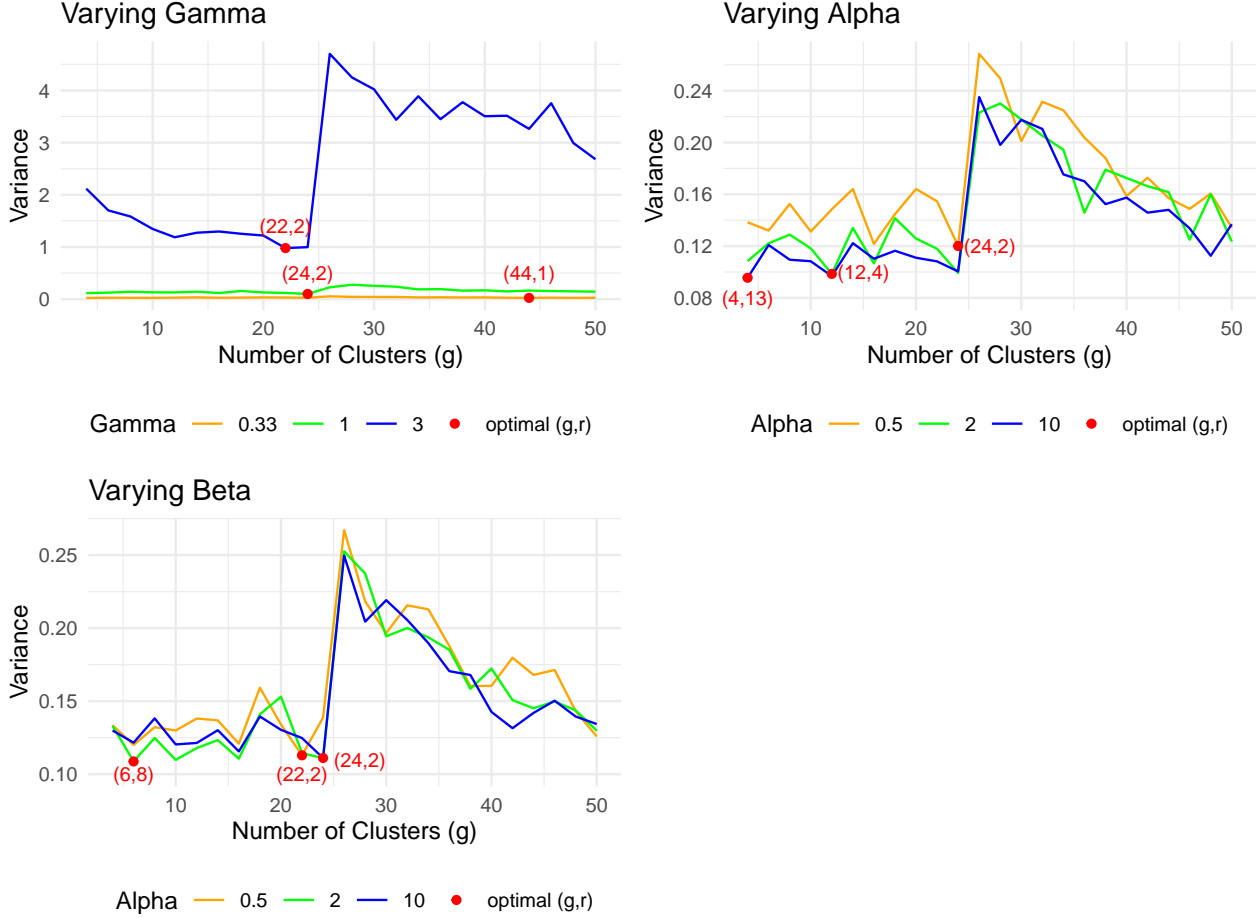


Figure 3: Variance of Beta Estimator for Varying  $g$  and Data Parameters Under Low Cost Ratio, Poisson

**Moderate Cost Ratio** Under moderate cost ratio in the Poisson, the patterns look similar to under the normal, seen in Figure 4. However, as  $\gamma$  increases, we now see that the optimal  $g$  also increases, unlike under the normal where  $g$  decreased. We see that for  $\gamma = 3$ , the variance is actually decreasing for small  $g$  before it starts to increase again, and the optimal design is close to that under low cost ratio. The optimal designs under the different  $\alpha$  and  $\beta$  values are similarly small, but the largest value has the largest optimal  $g$  in both cases which is the opposite of what we saw under the low cost ratio.

**High Cost Ratio** Under the high cost ratio, the results are similar to the moderate cost ratio. As seen in Table 2, large  $\gamma$  has an medium sized optimal  $g$  and the others have the same relatively small  $g$  of 10. The optimal designs when varying  $\alpha$  range from 4 to 10 clusters, but the variances are essentially identical, indicating that each would perform the same. For the different  $\beta$ s, the optimal design is the same and uses the minimum number of clusters.

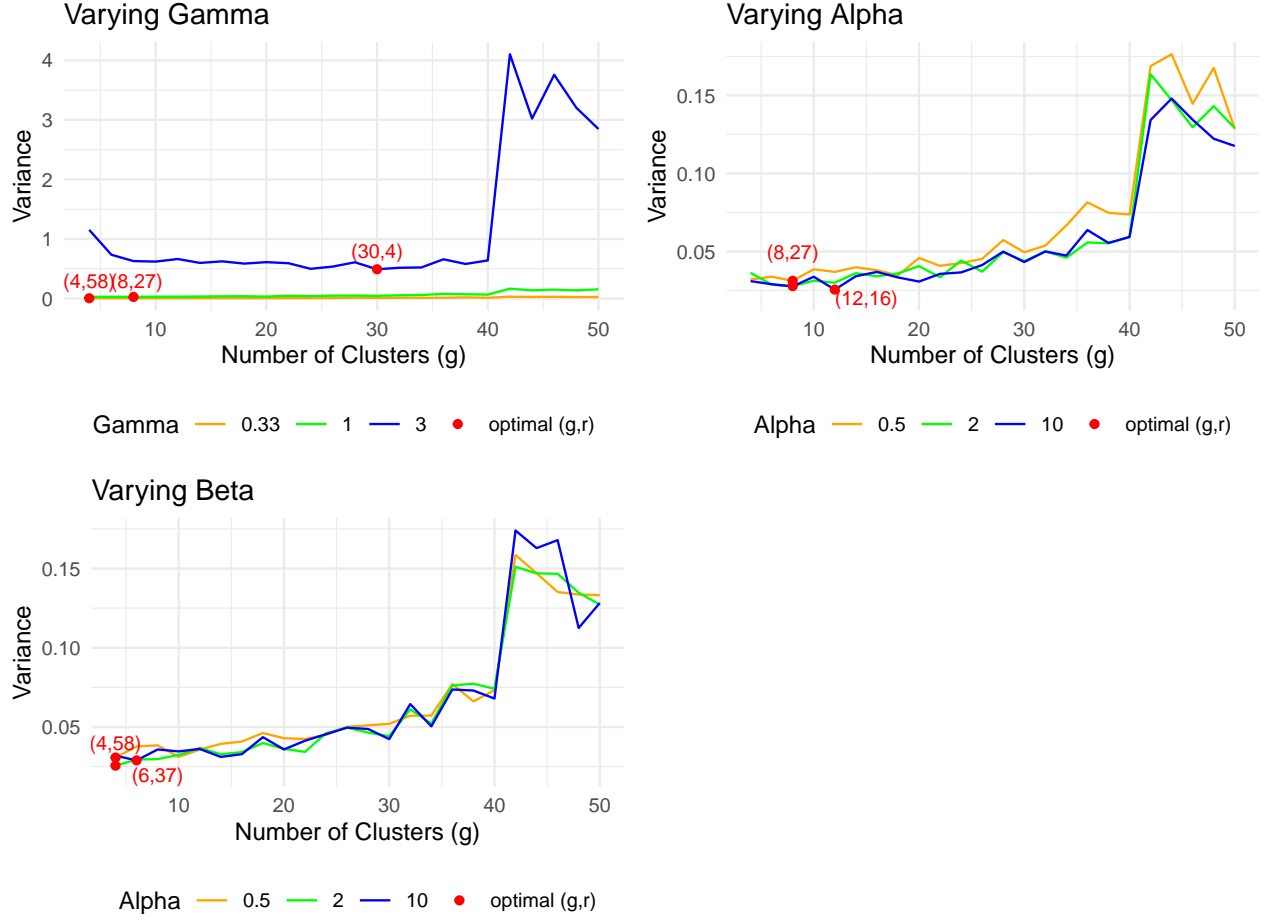


Figure 4: Variance of Beta Estimator for Varying  $g$  and Data Parameters Under Moderate Cost Ratio, Poisson

Table 2: Optimal Designs For Varying Parameters under High Cost Ratio, Poisson

value	g	r	variance
<b>alpha</b>			
0.5	4	231	0.007
2	10	81	0.007
10	6	147	0.007
<b>beta</b>			
0.5	4	231	0.007
2	4	231	0.008
10	4	231	0.007
<b>gamma</b>			
1	10	81	0.001
3	10	81	0.008
0.33	32	12	0.264

## Conclusions

In general, as the cost ratio increases, the optimal design will have a small number of clusters and a larger number of observations from each cluster and the varying data parameters have less of an effect. Under the normal distribution, a smaller  $\gamma$  and larger  $\beta$  result in a larger optimal number of clusters when the cost ratio is low, but these differences become less prominent as the cost ratio increases.  $\alpha$  does not appear to have an effect on the optimal design across any of the cost ratios. Under the Poisson distribution, we also see a larger optimal  $g$  for smaller  $\gamma$ , which drops off when the cost ratio increases. For large  $\gamma$  the optimal design has a mid-sized  $g$  and does not change much across different cost settings. Under low cost ratio, the effect of  $\alpha$  and  $\beta$  are the opposite as in the normal, with  $\beta$  appearing to have no effect and increasing  $\alpha$  having decreasing  $g$ . However, we still that the optimal designs are small and similar or identical across the  $\beta$  and  $\alpha$  values as the cost ratio increases.

## Limitations

Some of the differences between the Poisson case and the normal case are difficult to explain given our results. It is unclear why the effect of  $\beta$  and  $\alpha$  switch for the different distributions. One possibility is that the results are too noisy to see the true relationship. We only ran each setting for 300 iterations, so increasing that number may be helpful. It may also be the combination of parameter values that we used, as we only varied one parameter at a time. It could be useful to use a factorial design for the simulations, where we vary multiple parameters at once and look at many more combinations of parameter values. These are both good targets for future study to help clarify the reason for some of our results.

## References

- Cook AJ, DeLong E, Murray DM, Vollmer WM, Heagerty PJ. Statistical lessons learned for designing cluster randomized pragmatic clinical trials from the NIH Health Care Systems Collaboratory Biostatistics and Design Core. *Clinical Trials*. 2016;13(5):504-512. doi:10.1177/1740774516646578
- Hurley, J. C. (2020). How the cluster-randomized trial “works”. *Clinical Infectious Diseases*, 70(2), 341-346.
- Puffer, S., Torgerson, D. J., & Watson, J. (2005). Cluster randomized controlled trials. *Journal of evaluation in clinical practice*, 11(5), 479-483



## Code Appendix

```
knitr::opts_chunk$set(echo = FALSE,
                      warning = FALSE,
                      message = FALSE)

library(tidyverse)
library(lme4)
library(gridExtra)
library(knitr)
library(kableExtra)

sim_data<-function(alpha, beta, g, r, gamma, sigma, distribution){
  #'
  #' simulates data from a hierarchical data structure for normal or poisson distributions
  #' @param alpha the true intercept of the fixed term in the model
  #' @param beta the true treatment effect of the fixed term in the model
  #' @param g the number of clusters in the data, multiple of 2
  #' @param r the number of times to sample from each cluster
  #' @param gamma the standard deviation of the random intercept
  #' @param sigma the standard deviation of the error term
  #' @distribution one of "normal" or "poisson", the distribution to sample from
  #' @return a list of the data and the estimated beta hat
  #'

  # generate treatment assignment for clusters, resample if all treated or all
  # control
  x<-rep(c(0,1), g/2)
  # sum=0
  # while(sum==0|sum==g){
  #   x<-rbinom(g, 1, .5)
  #   sum<-sum(x)
  # }
  # generate treatment assignment for observations (r from g groups)
  x_i<-rep(x, r)

  # save the cluster assignments for each observation
  j<-rep(1:g, r)

  # fixed effect mu_0 for each cluster
  mu_i0<-alpha+beta*x

  # random effect mu_i
  mu_i<-rnorm(g*r, mean=mu_i0, sd=gamma)

  if (distribution=="normal"){

    # generate normal outcomes
    y<-mu_i+rnorm(r*g, 0, sigma)

    #fit linear model if r=1 (no clusters)
    if (r==1){
      fit<-lm(y~x_i)
    }
    # fit mixed effects model with random intercept
    else{
```

```

    fit<-lmer(y~x_i+(1|j))
  }

  # extract beta estimate
  beta_hat<-summary(fit)$coefficients[2]

}

else {
  # log link, so exponentiate the normal
  mu_i<-exp(mu_i)

  # generate poisson outcome
  y<-rpois(r*g, mu_i)

  # fit linear if r==1
  err=FALSE
  if (r==1){
    fit<-tryCatch({glm(y~x_i, family=poisson)}, error=function(e){
      err<-TRUE
    })
  }
  # fit poisson mixed effects model with random intercept, NA if error
  else{
    fit<-tryCatch({glmer(y~x_i+(1|j), family = poisson)}, error=function(e){
      err<-TRUE
    })
  }
}

# extract coefficient
beta_hat<-ifelse(err==TRUE, NA, summary(fit)$coefficients[2])

}

data=data.frame(y=y, x=x_i, cluster=j)

return(list(data, beta_hat, num_clusters=g, num_samples=r))
}

#s<-sim_data(1, 2, 4, 10, 1, 1, "poisson")
gen_data<-function(alpha, beta, g, r, gamma, sigma, distribution, folder, filename){
  #'
  #' Generate data and saves two files (one with data, one with coefficients and settings)
  #' @param alpha the true intercept of the fixed term in the model
  #' @param beta the true treatment effect of the fixed term in the model
  #' @param g the number of clusters in the data
  #' @param r the number of times to sample from each cluster
  #' @param gamma the standard deviation of the random intercept
  #' @param sigma the standard deviation of the error term
  #' @distribution one of "normal" or "poisson", the distribution to sample from

```

```

#' @param folder Path to folder in which to save files (include ending slash).
#' @param filename File name prefix.
#' @return saves 2 csv files with _data.csv and _coef_setting.csv suffixes.
#'

#simulate data
data<-sim_data(alpha, beta, g, r, gamma, sigma, distribution)

# data file
df<-data[[1]]
write.csv(df, paste0(folder, filename, "_data.csv"), row.names=FALSE)

# coef and settings file
setting_df<-data.frame(alpha, beta, g, r, gamma, sigma, coef_est=data[[2]])
write.csv(setting_df, paste0(folder, filename, "_coef_setting.csv"), row.names=FALSE)
}

get_results_row<-function(iter, alpha, beta, g, r, gamma, sigma, distribution){
  #'
  #' returns the data settings and the evaluation metrics
  #' @iter number of iterations
  #' @param alpha the true intercept of the fixed term in the model
  #' @param beta the true treatment effect of the fixed term in the model
  #' @param g the number of clusters in the data
  #' @param r the number of times to sample from each cluster
  #' @param gamma the standard deviation of the random intercept
  #' @param sigma the standard deviation of the error term
  #' @distribution one of "normal" or "poisson", the distribution to sample from
  #' @return data frame row containing the settings used and the variance of the estimator
  #'
  # initialize vector of beta alphas
  beta_hats<-c()

  # simulate iter times and extract values
  for (i in 1:iter){
    simulation<-sim_data(alpha, beta, g, r, gamma, sigma, distribution)
    beta_hats<-c(beta_hats, simulation[[2]])
  }

  # calculate variance of estimates
  variance<-var(beta_hats, na.rm = TRUE)
  icc<-gamma^2/(gamma^2+sigma^2)

  return(data.frame(iterations=iter, alpha, beta, g, r, gamma, sigma, distribution, variance, icc))
}

calc_g_r<-function(B, c1, c2){
  #' calculates the range of g values and maximum number of samples from each cluster
  #' to max out the budget

```

```

#' @param B the budget
#' @param c1 the cost of the first sample from each cluster
#' @param c2 the cost of each successive sample from each cluster
#' @return the sequence of g and r pairs that maximize the budget
#'
g<-seq(4, floor(B/c1), 2)
r<-floor(B/(g*c2)-c1/c2+1)

return(list(g, r))
}

run_experiments<-function(B, c1, c2, iter, alpha, beta, gamma, sigma, distribution, filename){
  #'
  #' runs the simulation for all given settingsn and saves the results to file
  #' @param B the total budget
  #' @param c1 the cost of sampling a cluster for the first time
  #' @param c2 the cost of sampling a cluster after the first time
  #' @param iter number of iterations
  #' @param alpha the true intercept of the fixed term in the model
  #' @param beta the true treatment effect of the fixed term in the model
  #' @param gamma the standard deviation of the random intercept
  #' @param sigma the standard deviation of the error term
  #' @param one of "normal" or "poisson", the distribution to sample from
  #' @param filename the name of the file to save to, if NA, will not save
  #' @return data frame containing the settings and evaluation measures for each experiment

  # calculate g and r pairs
  pairs<-calc_g_r(B, c1, c2)

  #extract
  g<-pairs[[1]]
  r<-pairs[[2]]

  results<-data.frame()
  # run simulations
  for (i in iter){
    for (a in alpha){
      for(b in beta){
        for(g1 in g){
          for(gam in gamma){
            for(s in sigma){
              row<-get_results_row(i, a, b, g1, r[which(g==g1)], gam, s, distribution)
              results<-rbind(results, row)
            }
          }
        }
      }
    }
  }

  if(!is.na(filename)){
    write.csv(results, paste0(filename, ".csv"))
  }
}

```

```

    }
    return(results)
}
set.seed(1)
vary_gamma1<-run_experiments(B=5000, c1=100, c2=95, 300, 1, 2, c(1/3, 1, 3), 1, "normal", "95_c2_vary_gamma")
vary_sigma1<-run_experiments(B=5000, c1=100, c2=95, 300, 1, 2, 1, c(1/3, 1, 3), "normal", "95_c2_vary_sigma")
vary_alpha1<-run_experiments(B=5000, c1=100, c2=95, 300, c(.5, 2, 10), 2, 1, 1, "normal", "95_c2_vary_alpha")
vary_beta1<-run_experiments(B=5000, c1=100, c2=95, 300, 1, c(.5, 2, 10), 1, 1, "normal", "95_c2_vary_beta")

# c2=20
vary_gamma2<-run_experiments(B=5000, c1=100, c2=20, 300, 1, 2, c(1/3, 1, 3), 1, "normal", "20_c2_vary_gamma")
vary_sigma2<-run_experiments(B=5000, c1=100, c2=20, 300, 1, 2, 1, c(1/3, 1, 3), "normal", "20_c2_vary_sigma")
vary_alpha2<-run_experiments(B=5000, c1=100, c2=20, 300, c(.5, 2, 10), 2, 1, 1, "normal", "20_c2_vary_alpha")
vary_beta2<-run_experiments(B=5000, c1=100, c2=20, 300, 1, c(.5, 2, 10), 1, 1, "normal", "20_c2_vary_beta")

# c2=5
vary_gamma3<-run_experiments(B=5000, c1=100, c2=5, 300, 1, 2, c(1/3, 1, 3), 1, "normal", "5_c2_vary_gamma")
vary_sigma3<-run_experiments(B=5000, c1=100, c2=5, 300, 1, 2, 1, c(1/3, 1, 3), "normal", "5_c2_vary_sigma")
vary_alpha3<-run_experiments(B=5000, c1=100, c2=5, 300, c(.5, 2, 10), 2, 1, 1, "normal", "5_c2_vary_alpha")
vary_beta3<-run_experiments(B=5000, c1=100, c2=5, 300, 1, c(.5, 2, 10), 1, 1, "normal", "5_c2_vary_beta")

# read in saved normal results
vary_gamma1<-read.csv("95_c2_vary_gamma.csv")
vary_sigma1<-read.csv("95_c2_vary_sigma.csv")
vary_alpha1<-read.csv("95_c2_vary_alpha.csv")
vary_beta1<-read.csv("95_c2_vary_beta.csv")

vary_gamma2<-read.csv("20_c2_vary_gamma.csv")
vary_sigma2<-read.csv("20_c2_vary_sigma.csv")
vary_alpha2<-read.csv("20_c2_vary_alpha.csv")
vary_beta2<-read.csv("20_c2_vary_beta.csv")

vary_gamma3<-read.csv("5_c2_vary_gamma.csv")
vary_sigma3<-read.csv("5_c2_vary_sigma.csv")
vary_alpha3<-read.csv("5_c2_vary_alpha.csv")
vary_beta3<-read.csv("5_c2_vary_beta.csv")

opt_gamma<-rbind(cbind(vary_gamma1, c2=95), cbind(vary_gamma2, c2=20), cbind(vary_gamma3, c2=5))%>%group_by(c2)
opt_sigma<-rbind(cbind(vary_sigma1, c2=95), cbind(vary_sigma2, c2=20), cbind(vary_sigma3, c2=5))%>%group_by(c2)
opt_alpha<-rbind(cbind(vary_alpha1, c2=95), cbind(vary_alpha2, c2=20), cbind(vary_alpha3, c2=5))%>%group_by(c2)
opt_beta<-rbind(cbind(vary_beta1, c2=95), cbind(vary_beta2, c2=20), cbind(vary_beta3, c2=5))%>%group_by(c2)
gamma.plot1 <- ggplot(data = vary_gamma1, aes(x = g, y=variance, color=factor(round(gamma, digits=2))))
  geom_line()+
  geom_point(data=opt_gamma[opt_gamma$c2==95,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 5, .7, label=paste0("(",as.character(opt_gamma$g[2]),",", as.character(opt_gamma$r[2]),")",
    color="red", size=3)+
  annotate("text", 5, .25, label=paste0("(",as.character(opt_gamma$g[1]),",", as.character(opt_gamma$r[1]),")",
    color="red", size=3)+
  annotate("text", 50, .2, label=paste0("(",as.character(opt_gamma$g[3]),",", as.character(opt_gamma$r[3]),")",
    color="red", size=3)+
  labs(

```

```

    title = "Varying Gamma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Gamma", values=c("0.33"="orange", "1"="green", "3"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#gamma.plot1

sigma.plot1 <- ggplot(data = vary_sigma1, aes(x = g, y=variance, color=factor(round(sigma, digits=2)))) +
  geom_line() +
  geom_point(data=opt_sigma[opt_sigma$c2==95,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 5, .65, label=paste0("(", as.character(opt_sigma$g[2]), ",", as.character(opt_sigma$r[2]), ")",
    color="red", size=3) +
  annotate("text", 5, .25, label=paste0("(", as.character(opt_sigma$g[1]), ",", as.character(opt_sigma$r[1]), ")",
    color="red", size=3) +
  annotate("text", 16, .2, label=paste0("(", as.character(opt_sigma$g[3]), ",", as.character(opt_sigma$r[3]), ")",
    color="red", size=3) +
  labs(
    title = "Varying Sigma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Sigma", values=c("0.33"="orange", "1"="green", "3"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#sigma.plot1

alpha.plot1 <- ggplot(data = vary_alpha1, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line() +
  geom_point(data=opt_alpha[opt_alpha$c2==95,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 24, .13, label=paste0("(", as.character(opt_alpha$g[2]), ",", as.character(opt_alpha$r[2]), ")",
    color="red", size=3) +
  annotate("text", 10, .127, label=paste0("(", as.character(opt_alpha$g[1]), ",", as.character(opt_alpha$r[1]), ")",
    color="red", size=3) +
  annotate("text", 6, .135, label=paste0("(", as.character(opt_alpha$g[3]), ",", as.character(opt_alpha$r[3]), ")",
    color="red", size=3) +
  labs(
    title = "Varying Alpha",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#alpha.plot1

beta.plot1 <- ggplot(data = vary_beta1, aes(x = g, y=variance, color=factor(beta))) +
  geom_line() +
  geom_point(data=opt_beta[opt_beta$c2==95,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 12.75, .142, label=paste0("(", as.character(opt_beta$g[2]), ",", as.character(opt_beta$r[2]), ")",
    color="red", size=3) +
  annotate("text", 7.9, .1375, label=paste0("(", as.character(opt_beta$g[1]), ",", as.character(opt_beta$r[1]), ")",

```

```

        color="red", size=3)+
  annotate("text", 50, .13, label=paste0("(",as.character(opt_beta$g[3]),",", as.character(opt_beta$r[3]),")",
        color="red", size=3)+
  labs(
    title = "Varying Beta",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha",values=c("0.5"="orange", "2"="green", "10"="blue","optimal (g,r)"="red"))
  theme(legend.position = "bottom")
#beta.plot1

grid.arrange(gamma.plot1, alpha.plot1, beta.plot1, nrow=2)
gamma.plot2 <- ggplot(data = vary_gamma2, aes(x = g, y=variance, color=factor(round(gamma, digits=2)))) +
  geom_line()+
  geom_point(data=opt_gamma[opt_gamma$c2==20,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 5.5, .08, label=paste0("(",as.character(opt_gamma$g[4]),",", as.character(opt_gamma$r[4]),")",
        color="red", size=3)+
  annotate("text", 10.5, .1, label=paste0("(",as.character(opt_gamma$g[5]),",", as.character(opt_gamma$r[5]),")",
        color="red", size=3)+
  labs(
    title = "Varying Gamma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Gamma",values=c("0.33"="orange", "1"="green", "3"="blue","optimal (g,r)"="red"))
  theme(legend.position = "bottom")
#gamma.plot2

alpha.plot2 <- ggplot(data = vary_alpha2, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line()+
  geom_point(data=opt_alpha[opt_alpha$c2==20,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 10, .02, label=paste0("(",as.character(opt_alpha$g[4]),",", as.character(opt_alpha$r[4]),")",
        color="red", size=3)+
  annotate("text", 6, .05, label=paste0("(",as.character(opt_alpha$g[5]),",", as.character(opt_alpha$r[5]),")",
        color="red", size=3)+
  labs(
    title = "Varying Alpha",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha",values=c("0.5"="orange", "2"="green", "10"="blue","optimal (g,r)"="red"))
  theme(legend.position = "bottom")
#alpha.plot1

beta.plot2 <- ggplot(data = vary_beta2, aes(x = g, y=variance, color=factor(beta))) +
  geom_line()+
  geom_point(data=opt_beta[opt_beta$c2==20,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 4, .05, label=paste0("(",as.character(opt_beta$g[4]),",", as.character(opt_beta$r[4]),")",
        color="red", size=3)+
  annotate("text", 10, .025, label=paste0("(",as.character(opt_beta$g[6]),",", as.character(opt_beta$r[6]),")",

```

```

        color="red", size=3)+

labs(
  title = "Varying Beta",
  x = "Number of Clusters (g)",
  y = "Variance") +
theme_minimal() +
scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red"))
theme(legend.position = "bottom")
#beta.plot1

grid.arrange(gamma.plot2, alpha.plot2, beta.plot2, nrow=2)
gamma.plot3 <- ggplot(data = vary_gamma3, aes(x = g, y=variance, color=factor(round(gamma, digits=2)))) +
  geom_line() +
  geom_point(data=opt_gamma[opt_gamma$c2==5,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 4, .009, label=paste0("(", as.character(opt_gamma$g[7]), ",", as.character(opt_gamma$r[7]),
    color="red", size=3)+
  annotate("text", 6, .045, label=paste0("(", as.character(opt_gamma$g[8]), ",", as.character(opt_gamma$r[8]),
    color="red", size=3)+
  annotate("text", 10, .005, label=paste0("(", as.character(opt_gamma$g[9]), ",", as.character(opt_gamma$r[9]),
    color="red", size=3)+

labs(
  title = "Varying Gamma",
  x = "Number of Clusters (g)",
  y = "Variance") +
theme_minimal() +
scale_color_manual(name="Gamma", values=c("0.33"="orange", "1"="green", "3"="blue", "optimal (g,r)"="red"))
theme(legend.position = "bottom")
#gamma.plot3

alpha.plot3 <- ggplot(data = vary_alpha3, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line() +
  geom_point(data=opt_alpha[opt_alpha$c2==5,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 4, .007, label=paste0("(", as.character(opt_alpha$g[8]), ",", as.character(opt_alpha$r[8]),
    color="red", size=3)+
  annotate("text", 6, .009, label=paste0("(", as.character(opt_alpha$g[7]), ",", as.character(opt_alpha$r[7]),
    color="red", size=3)+

labs(
  title = "Varying Alpha",
  x = "Number of Clusters (g)",
  y = "Variance") +
theme_minimal() +
scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red"))
theme(legend.position = "bottom")
#alpha.plot3

beta.plot3 <- ggplot(data = vary_beta3, aes(x = g, y=variance, color=factor(beta))) +
  geom_line() +
  geom_point(data=opt_beta[opt_beta$c2==5,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 4, .008, label=paste0("(", as.character(opt_beta$g[7]), ",", as.character(opt_beta$r[7]),
    color="red", size=3)+

```



```

  annotate("text", 10, .01, label=paste0("(" ,as.character(opt_beta$g[8]),",", as.character(opt_beta$r[8]),
    color="red", size=3))+

  labs(
    title = "Varying Beta",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red"),
    theme(legend.position = "bottom"))
#beta.plot3

#grid.arrange(gamma.plot3, alpha.plot3, beta.plot3, nrow=2)

full<-rbind(opt_alpha, opt_beta, opt_gamma)%>%filter(c2==5)%>%
  select(g, r, variance)
full1<-cbind(value=c("0.5", "2", "10", "0.5", "2", "10", "1","3","0.33"), full,
  parameter=c(rep("alpha", 3),
    rep("beta", 3), rep("gamma", 3)))

kable(full1[,-c(2, 3, 7)], digits=c(2, 0, 0,3), caption="Optimal Designs For Varying Parameters under H0",
  pack_rows(index = table(full1$parameter)))
set.seed(1)
vary_gamma4<-run_experiments(B=5000, c1=100, c2=95, 300, 1, 2, c(1/3, 1, 3), 1, "poisson",
  "95_c2_vary_gamma_pois")
vary_sigma4<-run_experiments(B=5000, c1=100, c2=95, 300, 1, 2, 1, c(1/3, 1, 3), "poisson",
  "95_c2_vary_sigma_pois")
vary_alpha4<-run_experiments(B=5000, c1=100, c2=95, 300, c(.5,2, 10), 2, 1, 1, "poisson",
  "95_c2_vary_alpha_pois")
vary_beta4<-run_experiments(B=5000, c1=100, c2=95, 300, 1, c(.5, 2, 10), 1, 1, "poisson",
  "95_c2_vary_beta_pois")

# c2=20
vary_gamma5<-run_experiments(B=5000, c1=100, c2=20, 300, 1, 2, c(1/3, 1, 3), 1, "poisson",
  "20_c2_vary_gamma_pois")
vary_sigma5<-run_experiments(B=5000, c1=100, c2=20, 300, 1, 2, 1, c(1/3, 1, 3), "poisson",
  "20_c2_vary_sigma_pois")
vary_alpha5<-run_experiments(B=5000, c1=100, c2=20, 300, c(.5,2, 10), 2, 1, 1, "poisson",
  "20_c2_vary_alpha_pois")
vary_beta5<-run_experiments(B=5000, c1=100, c2=20, 300, 1, c(.5, 2, 10), 1, 1, "poisson",
  "20_c2_vary_beta_pois")

# c2=5
#vary_gamma6<-run_experiments(B=5000, c1=100, c2=5, 300, 1, 2, c(1/3, 1, 3), 1, "poisson",
#  "5_c2_vary_gamma_pois")
#vary_sigma6<-run_experiments(B=5000, c1=100, c2=5, 300, 1, 2, 1, c(1/3, 1, 3), "poisson",
#  "5_c2_vary_sigma_pois")
#vary_alpha6<-run_experiments(B=5000, c1=100, c2=5, 300, c(.5,2, 10), 2, 1, 1, "poisson",
#  "5_c2_vary_alpha_pois")
#vary_beta6<-run_experiments(B=5000, c1=100, c2=5, 300, 1, c(.5, 2, 10), 1, 1, "poisson",
#  "5_c2_vary_beta_pois")

```

```

# read in saved poisson results
vary_gamma4<-read.csv("95_c2_vary_gamma_pois.csv")
vary_sigma4<-read.csv("95_c2_vary_sigma_pois.csv")
vary_alpha4<-read.csv("95_c2_vary_alpha_pois.csv")
vary_beta4<-read.csv("95_c2_vary_beta_pois.csv")

vary_gamma5<-read.csv("20_c2_vary_gamma_pois.csv")
vary_sigma5<-read.csv("20_c2_vary_sigma_pois.csv")
vary_alpha5<-read.csv("20_c2_vary_alpha_pois.csv")
vary_beta5<-read.csv("20_c2_vary_beta_pois.csv")

vary_gamma6<-read.csv("5_c2_vary_gamma_pois.csv")
vary_sigma6<-read.csv("5_c2_vary_sigma_pois.csv")
vary_alpha6<-read.csv("5_c2_vary_alpha_pois.csv")
vary_beta6<-read.csv("5_c2_vary_beta_pois.csv")

opt_gamma1<-rbind(cbind(vary_gamma4, c2=95), cbind(vary_gamma5, c2=20), cbind(vary_gamma6, c2=5))%>%group_by(c2)
opt_alpha1<-rbind(cbind(vary_alpha4, c2=95), cbind(vary_alpha5, c2=20), cbind(vary_alpha6, c2=5))%>%group_by(c2)
opt_beta1<-rbind(cbind(vary_beta4, c2=95), cbind(vary_beta5, c2=20), cbind(vary_beta6, c2=5))%>%group_by(c2)
gamma.plot4 <- ggplot(data = vary_gamma4, aes(x = g, y=variance, color=factor(round(gamma, digits=2)))) +
  geom_line()+
  geom_point(data=opt_gamma1[opt_gamma1$c2==95,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 24, .5, label=paste0("(",as.character(opt_gamma1$g[2]),",", as.character(opt_gamma1$variance[2]),")",
    color="red", size=3)+
  annotate("text", 22, 1.4, label=paste0("(",as.character(opt_gamma1$g[1]),",", as.character(opt_gamma1$variance[1]),")",
    color="red", size=3)+
  annotate("text", 44, .5, label=paste0("(",as.character(opt_gamma1$g[3]),",", as.character(opt_gamma1$variance[3]),")",
    color="red", size=3)+
  labs(
    title = "Varying Gamma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Gamma",values=c("0.33"="orange", "1"="green","3"="blue","optimal (g,r)"="red"))+
  theme(legend.position = "bottom")
#gamma.plot4

alpha.plot4 <- ggplot(data = vary_alpha4, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line()+
  geom_point(data=opt_alpha1[opt_alpha1$c2==95,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 15, .1, label=paste0("(",as.character(opt_alpha1$g[2]),",", as.character(opt_alpha1$variance[2]),")",
    color="red", size=3)+
  annotate("text", 27, .12, label=paste0("(",as.character(opt_alpha1$g[1]),",", as.character(opt_alpha1$variance[1]),")",
    color="red", size=3)+
  annotate("text", 4, .08, label=paste0("(",as.character(opt_alpha1$g[3]),",", as.character(opt_alpha1$variance[3]),")",
    color="red", size=3)+
  labs(
    title = "Varying Alpha",
    x = "Number of Clusters (g)",

```

```

    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#alpha.plot4

beta.plot4 <- ggplot(data = vary_beta4, aes(x = g, y=variance, color=factor(beta))) +
  geom_line() +
  geom_point(data=opt_beta1[opt_beta1$c2==95,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 6, .1, label=paste0("(", as.character(opt_beta1$g[2]), ",", as.character(opt_beta1$r[2]),
    color="red", size=3) +
  annotate("text", 22, .1, label=paste0("(", as.character(opt_beta1$g[1]), ",", as.character(opt_beta1$r[1]),
    color="red", size=3) +
  annotate("text", 27.5, .11, label=paste0("(", as.character(opt_beta1$g[3]), ",", as.character(opt_beta1$r[3]),
    color="red", size=3) +
  labs(
    title = "Varying Beta",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#beta.plot4

grid.arrange(gamma.plot4, alpha.plot4, beta.plot4, nrow=2)
gamma.plot5 <- ggplot(data = vary_gamma5, aes(x = g, y=variance, color=factor(round(gamma, digits=2)))) +
  geom_line() +
  geom_point(data=opt_gamma1[opt_gamma1$c2==20,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 4, .3, label=paste0("(", as.character(opt_gamma1$g[4]), ",", as.character(opt_gamma1$r[4]),
    color="red", size=3) +
  annotate("text", 8.3, .3, label=paste0("(", as.character(opt_gamma1$g[5]), ",", as.character(opt_gamma1$r[5]),
    color="red", size=3) +
  annotate("text", 30, .8, label=paste0("(", as.character(opt_gamma1$g[6]), ",", as.character(opt_gamma1$r[6]),
    color="red", size=3) +
  labs(
    title = "Varying Gamma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Gamma", values=c("0.33"="orange", "1"="green", "3"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#gamma.plot5

alpha.plot5 <- ggplot(data = vary_alpha5, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line() +
  geom_point(data=opt_alpha1[opt_alpha1$c2==20,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 8, .05, label=paste0("(", as.character(opt_alpha1$g[4]), ",", as.character(opt_alpha1$r[4]),
    color="red", size=3) +
  annotate("text", 15, 0.02, label=paste0("(", as.character(opt_alpha1$g[6]), ",", as.character(opt_alpha1$r[6]),
    color="red", size=3) +

```

```

labs(
  title = "Varying Alpha",
  x = "Number of Clusters (g)",
  y = "Variance") +
theme_minimal() +
scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red"))
theme(legend.position = "bottom")
#alpha.plot5

beta.plot5 <- ggplot(data = vary_beta5, aes(x = g, y=variance, color=factor(beta))) +
  geom_line()+
  geom_point(data=opt_beta1[opt_beta1$c2==20,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 4, .04, label=paste0("(", as.character(opt_beta1$g[4]), ",", as.character(opt_beta1$r[4]),
    color="red", size=3)+
  annotate("text", 8, .02, label=paste0("(", as.character(opt_beta1$g[6]), ",", as.character(opt_beta1$r[6]),
    color="red", size=3)+
  labs(
    title = "Varying Beta",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red"))
  theme(legend.position = "bottom")
#beta.plot5

grid.arrange(gamma.plot5, alpha.plot5, beta.plot5, nrow=2)
gamma.plot6 <- ggplot(data = vary_gamma6, aes(x = g, y=variance, color=factor(round(gamma, digits=2)))) +
  geom_line()+
  geom_point(data=opt_gamma1[opt_gamma1$c2==5,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 10, .2, label=paste0("(", as.character(opt_gamma1$g[7]), ",", as.character(opt_gamma1$r[7]),
    color="red", size=3)+
  annotate("text", 32, .5, label=paste0("(", as.character(opt_gamma1$g[1]), ",", as.character(opt_gamma1$r[1]),
    color="red", size=3)+
  labs(
    title = "Varying Gamma",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Gamma", values=c("0.33"="orange", "1"="green", "3"="blue", "optimal (g,r)"="red"))
  theme(legend.position = "bottom")
#gamma.plot6

alpha.plot6 <- ggplot(data = vary_alpha6, aes(x = g, y=variance, color=factor(alpha))) +
  geom_line()+
  geom_point(data=opt_alpha1[opt_alpha1$c2==5,], aes(x=g, y=variance, color="optimal (g,r)"))+
  annotate("text", 10, .0175, label=paste0("(", as.character(opt_alpha1$g[8]), ",", as.character(opt_alpha1$r[8]),
    color="red", size=3)+
  annotate("text", 4, .0175, label=paste0("(", as.character(opt_alpha1$g[7]), ",", as.character(opt_alpha1$r[7]),
    color="red", size=3)+

```

```

    annotate("text", 6, 0, label=paste0("(", as.character(opt_alpha1$g[9]), ",", as.character(opt_alpha1$r[9]),
      color="red", size=3)+
  labs(
    title = "Varying Alpha",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#alpha.plot6

beta.plot6 <- ggplot(data = vary_beta6, aes(x = g, y=variance, color=factor(beta))) +
  geom_line() +
  geom_point(data=opt_beta1[opt_beta1$c2==5,], aes(x=g, y=variance, color="optimal (g,r)")) +
  annotate("text", 4.5, .02, label=paste0("(", as.character(opt_beta1$g[7]), ",", as.character(opt_beta1$r[7]),
    color="red", size=3)+
  labs(
    title = "Varying Beta",
    x = "Number of Clusters (g)",
    y = "Variance") +
  theme_minimal() +
  scale_color_manual(name="Alpha", values=c("0.5"="orange", "2"="green", "10"="blue", "optimal (g,r)"="red")) +
  theme(legend.position = "bottom")
#beta.plot6

#grid.arrange(gamma.plot6, alpha.plot6, beta.plot6, nrow=2)

full2<-rbind(opt_alpha1, opt_beta1, opt_gamma1)%>%filter(c2==5)%>%
  select(g, r, variance)
full3<-cbind(value=c("0.5", "2", "10", "0.5", "2", "10", "1", "3", "0.33"), full2,
  parameter=c(rep("alpha", 3),
    rep("beta", 3), rep("gamma", 3)))

kable(full3[,-c(2, 3, 7)], digits=c(2, 0, 0, 3), caption="Optimal Designs For Varying Parameters under H0",
  pack_rows(index = table(full3$parameter))

```