

ANCOVA Comparison Simulations: Imbalanced Treatment Assignment

Kellie Ottoboni

2017-10-02

Imbalanced Treatment Groups

Step 1: We generated the covariate and potential outcomes according to the methods in the continuous outcome section but discretized them by taking the floor. Assume that there are $n_j = 16$ individuals per stratum. In this set of simulations, we vary the number of treated units in each stratum. After sampling $(v_{ij}, \varepsilon_{ij}, \delta_{ij})$, we regenerate Z 10,000 times. We repeat this procedure for each distribution of latent variables v and of the errors ε and δ .

Step 2: We regenerate Z and recompute $Y(Z)$ 10,000 times for each design.

In expectation, the average treatment effect is γ . We compare the empirical power of five tests to detect this treatment effect:

- ANCOVA: we fit a linear model of response Y_1 on baseline Y_0 , treatment Z , and a dummy for stratum.
- Stratified permutation: we permute treatment assignment within stratum, then take the difference in means between treated and control outcomes Y_1
- Differenced permutation: we do the same permutation procedure as the stratified permutation test, except we use the difference between outcome and baseline, $Y_1 - Y_0$
- Linear model (LM) permutation: we use the same stratified permutation procedure as above, except use the t -statistic for the coefficient on treatment in the linear regression of Y_1 on Y_0 , Z , and stratum dummies
- Freedman-Lane test: see the other Rmd document for a full description of this procedure

Data-generation, tests, and plotting functions

```
gen_y <- function(gamma, v, error, Z) {  
  y <- 0.5 * ((2 * Z - 1) * gamma * exp(v) + exp(v/2)) + error  
  return(y)  
}  
  
gen_x <- function(gamma, v, error) {  
  x <- 0.5 * (-1 * gamma * exp(v) + exp(v/2)) + error  
  return(x)  
}  
  
generate_simulated_data <- function(gamma, effect, errors, n = c(16,  
  16, 16), ntreat = c(8, 8, 8)) {  
  # Input: gamma = multiplier for the magnitude of the  
  # treatment effect effect = 'same effect' or 'heterogeneous'  
  # errors = 'normal' or 'heavy' n = number of individuals at  
  # each stratum ntreat = number of treated individuals in each  
  # stratum Returns: a dataframe containing columns named Y1  
  # (response), Y0 (baseline), Z (treatment), gamma_vec
```

```

# (treatment effect per individual), stratumID (stratum),
# stratum_effect (beta coefficient per individual), and
# epsilon (errors)

stratumID <- rep(1:3, times = n)
N <- sum(n)

# What is the treatment effect?
if (effect == "same effect") {
  v <- runif(N, min = -4, max = 4)
} else if (effect == "heterogeneous") {
  v <- rep(0, N)
  v[stratumID == 1] <- runif(n[1], min = -4, max = -1)
  v[stratumID == 2] <- runif(n[2], min = -1, max = 1)
  v[stratumID == 3] <- runif(n[3], min = 1, max = 4)
} else {
  stop("invalid parameter effect")
}

# Generate errors
if (errors == "normal") {
  epsilon <- rnorm(N)
  delta <- rnorm(N)
} else if (errors == "t") {
  epsilon <- rt(N, df = 2)
  delta <- rt(N, df = 2)
} else if (errors == "lognormal") {
  epsilon <- rlnorm(N)
  delta <- rlnorm(N)
} else if (errors == "exponential") {
  epsilon <- rexp(N) - 1
  delta <- rexp(N) - 1
} else {
  stop("invalid errors parameter")
}

# Generate covariates
Z <- rep(0, N)
for (i in 1:3) {
  Z[stratumID == i] <- rep(1:0, each = c(ntreat[i], n[i] -
    ntreat[i]))
}
Y0 <- gen_x(gamma, v, epsilon)
Y1 <- gen_y(gamma, v, delta, Z)
return(data.frame(Y1, Y0, Z, v, stratumID, epsilon, delta))
}

generate_simulated_pvalues <- function(dataset, reps = 1000) {
  # Inputs: dataset = a dataframe containing columns named Y1
  # (response), Y0 (baseline), Z (treatment), and stratumID
  # (stratum) Returns: a vector of p-values first element is
  # the p-value from the ANCOVA second element is the p-value
  # from the stratified two-sample permutation test third

```

```

# element is the p-value from the linear model test,
# permuting treatment fourth element is the p-value from the
# Freedman-Lane linear model test, permuting residuals

# ANCOVA
modelfit <- lm(Y1 ~ Y0 + Z + factor(stratumID), data = dataset)
resanova <- summary(aov(modelfit))
anova_pvalue <- resanova[[1]]["Z", "Pr(>F)"]

# Stratified permutation test of Y1
observed_diff_means <- mean(dataset$Y1[dataset$Z == 1]) -
  mean(dataset$Y1[dataset$Z == 0])
diff_means_distr <- stratified_two_sample(group = dataset$Z,
  response = dataset$Y1, stratum = dataset$stratumID, reps = reps)
perm_pvalue <- t2p(observed_diff_means, diff_means_distr,
  alternative = "two-sided")

# Differenced permutation test of Y1-Y0
dataset$diff <- dataset$Y1 - dataset$Y0
observed_diff_means2 <- mean(dataset$diff[dataset$Z == 1]) -
  mean(dataset$diff[dataset$Z == 0])
diff_means_distr2 <- stratified_two_sample(group = dataset$Z,
  response = dataset$diff, stratum = dataset$stratumID,
  reps = reps)
perm_pvalue2 <- t2p(observed_diff_means2, diff_means_distr2,
  alternative = "two-sided")

# Permutation of treatment in linear model
observed_t1 <- summary(modelfit)[["coefficients"]][["Z", "t value"]]

# Freedman-Lane linear model residual permutation
lm2_no_tr <- lm(Y1 ~ Y0 + factor(stratumID), data = dataset)
dataset$lm2_resid <- residuals(lm2_no_tr)
lm2_yhat <- fitted(lm2_no_tr)

lm1and2_t_distr <- replicate(reps, {
  dataset[, c("Z_perm", "lm2_resid_perm")] <- permute_within_groups(dataset[,
    c("Z", "lm2_resid")], dataset$stratumID)
  lm1_perm <- lm(Y1 ~ Y0 + Z_perm + factor(stratumID),
    data = dataset)

  dataset$response_fl <- lm2_yhat + dataset$lm2_resid_perm
  lm2_perm <- lm(response_fl ~ Y0 + Z + factor(stratumID),
    data = dataset)

  c(summary(lm1_perm)[["coefficients"]][["Z_perm", "t value"]],
    summary(lm2_perm)[["coefficients"]][["Z", "t value"]])
})
lm_pvalue <- t2p(observed_t1, lm1and2_t_distr[1, ], alternative = "two-sided")
fl_pvalue <- t2p(observed_t1, lm1and2_t_distr[2, ], alternative = "two-sided")

return(c(ANCOVA = anova_pvalue, `Stratified Permutation` = perm_pvalue,
  `Differenced Permutation` = perm_pvalue2, `LM Permutation` = lm_pvalue,

```

```

    `Freedman-Lane` = fl_pvalue))
}

compute_power <- function(pvalues) {
  sapply((0:99)/100, function(p) mean(pvalues <= p, na.rm = TRUE))
}

plot_power_curves <- function(power_mat, title) {
  melt(power_mat) %>% mutate(pvalue = Var1/100) %>% mutate(Method = Var2) %>%
    ggplot(aes_string(x = "pvalue", y = "value", color = "Method")) +
    geom_line() + geom_abline(intercept = 0, slope = 1, linetype = "dashed") +
    xlab("P-value") + ylab("Power") + ggtitle(title) + theme_bw() +
    theme(axis.text.x = element_text(size = 12), axis.text.y = element_text(size = 12),
          axis.title = element_text(size = 16), title = element_text(size = 16),
          legend.title = element_text(size = 12), legend.text = element_text(size = 14),
          strip.text.x = element_text(size = 12))
}

plot_power_ntreat <- function(powermat_list, ntreat, alpha, title) {
  gamma_power_alpha <- t(sapply(powermat_list, function(x) x[floor(alpha *
    nrow(x)), ]))
  colnames(gamma_power_alpha) <- c("ANCOVA", "Stratified Permutation",
    "Differenced Permutation", "LM Permutation", "Freedman-Lane")
  melt(gamma_power_alpha, value.name = "Power") %>% mutate(Method = Var2) %>%
    mutate(ntreat = rep(ntreat, 5)) %>% ggplot(aes(x = ntreat,
    y = Power)) + geom_point(aes(color = Method)) + xlab("Number treated") +
    theme_bw() + theme(axis.text.x = element_text(size = 12),
    axis.text.y = element_text(size = 12), axis.title = element_text(size = 16),
    title = element_text(size = 16), legend.title = element_text(size = 12),
    legend.text = element_text(size = 14), strip.text.x = element_text(size = 12))
}

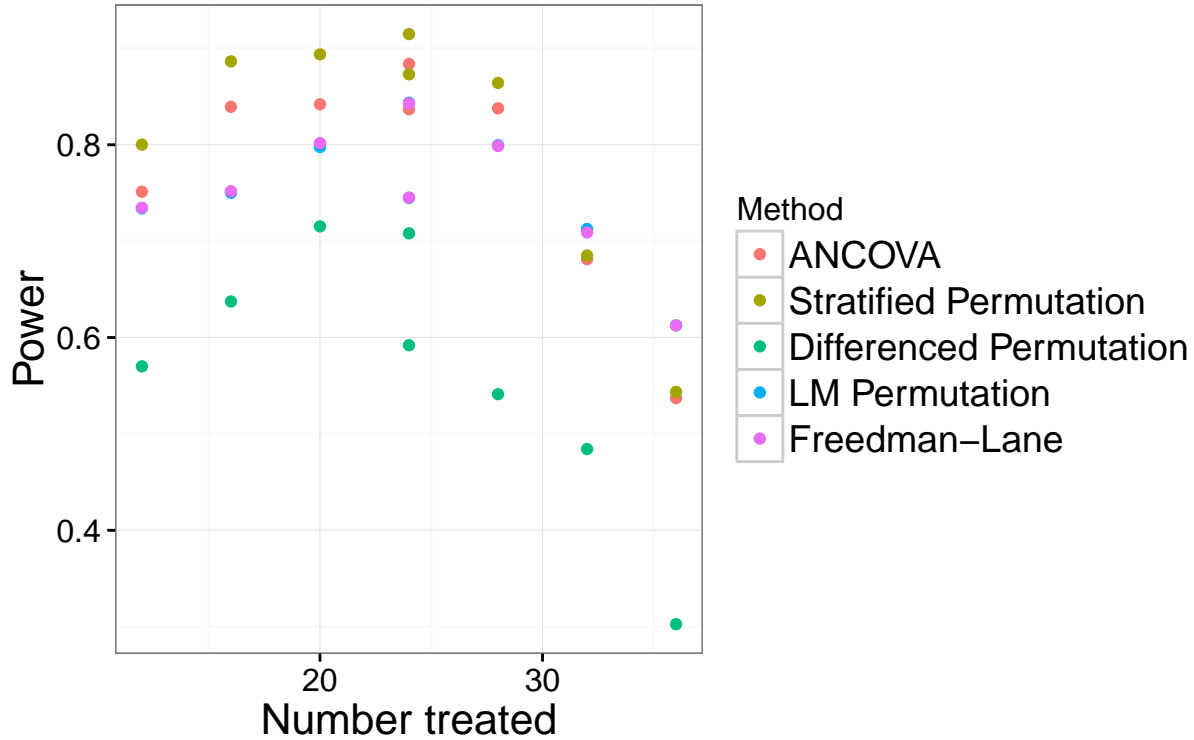
```

Constant additive treatment effect

```

load("imbalanced_tr/vary_ntreat_results.Rda")
ntreat_vec <- list(c(8, 8, 8), c(4, 4, 4), c(4, 4, 8), c(4, 8,
  8), c(4, 8, 12), c(8, 8, 12), c(8, 12, 12), c(12, 12, 12))
ntreat_power <- lapply(gamma_res, function(x) apply(x, 2, compute_power))
ntreat_str <- sapply(ntreat_vec, paste, collapse = ",")
ntreat_tot <- sapply(ntreat_vec, sum)
plot_power_ntreat(ntreat_power, ntreat_tot, 0.05, "xx")

```



First, we let the v_{ij} have the same distribution across strata and generated normally distributed errors. We varied the number of treated units in each group. In the random populations that were generated, this corresponded to population average treatment effects of about 1.36. Figure 1 shows the empirical power (rate of rejection in the 10,000 simulations) at level 5% plotted against the total number of treated units. Table 1 shows the same information. Each of the tests tended to have higher power when the total number of treated units was around half of the population (in this case, 24 out of 48). It is interesting to note that all tests had more power when 25% of units were treated than when 75% were treated; it seems that power is not a symmetric function.

The differenced permutation test had substantially lower power than the rest, again because the correlation between baseline X and outcome Y was low. In this case, the unadjusted stratified test had comparable power to the linear model based tests. ANCOVA had slightly higher power than the linear model permutation tests for all experiments except for when the number of treated units was equal to number of controls and when the number of treated units was 30 or above.

```
summary05 <- t(sapply(ntreat_power, function(x) x[5, ]))
summary05 <- cbind(ntreat_str, summary05)
colnames(summary05) <- c("Number Treated", "ANCOVA", "Stratified Permutation",
  "Differenced Permutation", "LM Permutation", "Freedman-Lane")
print(xtable(summary05[order(ntreat_tot), ], caption = "Empirical power at level $0.05$ for simulated d",
  include.rownames = FALSE))
```

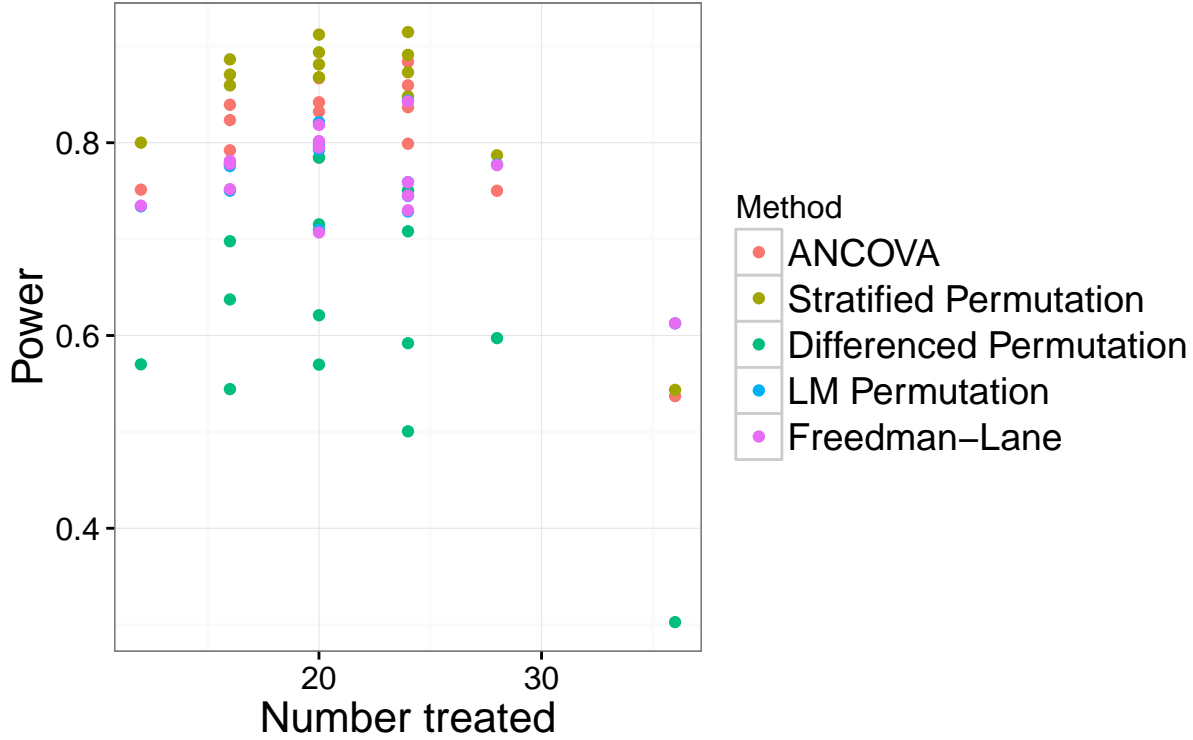
Heterogeneous treatment effect

```
load("imbalanced_tr/vary_ntreat_het_results.Rda")
ntreat_het_vec <- list(c(8, 8, 8), c(4, 4, 4), c(4, 4, 8), c(4,
  8, 4), c(8, 4, 4), c(4, 8, 8), c(8, 4, 8), c(8, 8, 4), c(4,
  8, 12), c(12, 8, 4), c(4, 12, 8), c(8, 12, 8), c(4, 12, 4),
  c(12, 12, 12))
```

Number Treated	ANCOVA	Stratified Permutation	Differenced Permutation	LM Permutation	Freedman-Lane
4,4,4	0.7513	0.8001	0.5701	0.734	0.7347
4,4,8	0.8394	0.8865	0.6374	0.7502	0.7518
4,8,8	0.842	0.8938	0.7153	0.7976	0.8017
8,8,8	0.8368	0.8731	0.592	0.8436	0.8429
4,8,12	0.8839	0.9148	0.7081	0.7448	0.7452
8,8,12	0.8378	0.8642	0.5412	0.7995	0.7987
8,12,12	0.6813	0.6851	0.4843	0.7126	0.7089
12,12,12	0.5371	0.5436	0.3026	0.6126	0.6126

Table 1: Empirical power at level 0.05 for simulated data with constant additive treatment effects

```
ntreat_het_power <- lapply(gamma_res_het, function(x) apply(x,
  2, compute_power))
ntreat_het_str <- sapply(ntreat_het_vec, paste, collapse = ",")
ntreat_het_tot <- sapply(ntreat_het_vec, sum)
plot_power_ntreat(ntreat_het_power, ntreat_het_tot, 0.05, "xx")
```



Next, we varied the distribution of v_{ij} across strata and generated normally distributed errors. Once again, we varied the number of treated units in each stratum and the population average treatment effects were around 1.36. Figure 2 and Table 2 show the empirical power at level 5% for these experiments. The pattern here is similar to Figure 1: power tends to be highest when the number of treated units is around half of the population. The treatment effect was largest in the third stratum and smallest in the first stratum; as expected, Table 2 shows that among experiments that fixed the total number of treated units, the experiment with more treatment units in the third stratum tended to have higher power. These variations are only slight, though. As before, the differenced permutation test had low power. The linear model permutation tests, ANCOVA, and stratified permutation test, in that order, had the best power.

```
summary05het <- t(sapply(ntreat_het_power, function(x) x[5, ]))
summary05het <- cbind(`Number treated` = ntreat_het_str, summary05het)
colnames(summary05het) <- c("Number Treated", "ANCOVA", "Stratified Permutation",
```

```

"Differenced Permutation", "LM Permutation", "Freedman-Lane")
print(xtable(summary05het[order(ntreat_het_tot), ], caption = "Empirical power at level $0.05$ for simu
include.rownames = FALSE)

```

Number Treated	ANCOVA	Stratified Permutation	Differenced Permutation	LM Permutation	Freedman-Lane
4,4,4	0.7513	0.8001	0.5701	0.734	0.7347
4,4,8	0.8394	0.8865	0.6374	0.7502	0.7518
4,8,4	0.7922	0.8595	0.6978	0.7786	0.7812
8,4,4	0.8235	0.8707	0.5444	0.7758	0.7778
4,8,8	0.842	0.8938	0.7153	0.7976	0.8017
8,4,8	0.8669	0.9122	0.5698	0.7927	0.7951
8,8,4	0.8323	0.8812	0.621	0.8213	0.8185
4,12,4	0.7844	0.868	0.7847	0.7096	0.7069
8,8,8	0.8368	0.8731	0.592	0.8436	0.8429
4,8,12	0.8839	0.9148	0.7081	0.7448	0.7452
12,8,4	0.8596	0.8912	0.5006	0.7591	0.759
4,12,8	0.799	0.8481	0.7504	0.7287	0.7299
8,12,8	0.7502	0.787	0.5973	0.7774	0.7768
12,12,12	0.5371	0.5436	0.3026	0.6126	0.6126

Table 2: Empirical power at level 0.05 for simulated data with heterogeneous treatment effects