Analyzing Longitudinal Data

Ashley I Naimi

Fall 2024

Contents

1 Classical versus Complex Longitudinal Data

In this course, you have already encountered causal inference via potential outcomes when exposure under study is measured once (i.e., time fixed). In this lecture, we will focus on longitudinal, and complex longitudinal data, and the complications that may arise when dealing with such data. For clarity, let's define complex longitudinal data. We will be dealing with data from a cohort study, individuals sampled from a well-defined target population, and clear study start and stop times (i.e., closed cohort). Data from such a cohort are **longitudinal** when they are measured repeatedly over time.[1]

Different scenarios can lead to longitudinal data:

1. exposure and covariates do not vary over time, but the study outcome is measured repeatedly in the same individual over follow up
2. exposure and covariates vary over time and are measured repeatedly in the same individual over follow up, but the study outcome can only occur (and/or is measured) only once
3. exposure and covariates vary over time and are measured repeatedly in the same individual over follow up, and the study outcome can occur more than once, and is measured repeatedly in the same individual over follow up.

Scenario 1 is the classical situation that one might refer to as "longitudinal" or correlated (outcomes) data. In this scenario, researchers often use mixed effects models or generalized estimating equations to deal with these data, but one can sometimes use simpler methods depending on the problem's context.

Repeated exposure, covariate, and (possibly) outcome measurement also leads to "longitudinal" data. But these data can result in something fundamentally different, which we refer to here as complex longitudinal data.

Repeated measurement over time creates the opportunity for us to capture complex causal relations between past and future covariates. Suppose we measure an exposure twice over follow-up, a covariate once, and the outcome at the end of follow-up (Figure 1). If we can assume that past exposure/covariate values do not affect future exposure/covariate values (usually a very risky assumption), we might not consider these data "complex," because we can use many standard methods to obtain correct results.

On the other hand, if past exposure/covariates affect future exposure/covariates in such a way that prior exposures or covariates confound future exposures

[1] Another such form is when data are measured repeatedly within clusters defined by geographic space (e.g., census tracts) or some other grouping (e.g., hospitals). We will not be dealing with these data here, though the methods to handle such data are very similar and in some cases identical.

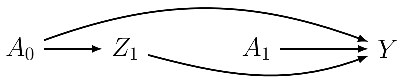

Figure 1: Longitudinal data that might not be considered 'complex' because there is no feedback between exposure and covariates.

(Figure 2), more advanced analytic techniques are needed.

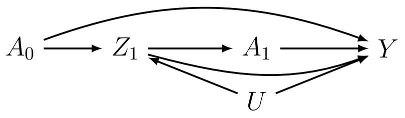

Figure 2: Causal diagram representing the relation between anti-retroviral treatment at time 0 (A_0), HIV viral load just prior to the second round of treatment (Z_1), anti-retroviral treatment status at time 1 (A_1), the CD4 count measured at the end of follow-up (Y), and an unmeasured common cause (U) of HIV viral load and CD4.

Here, we will learn why this distinction is important, and we'll cover a suite of methods that can be used to analyse data from each of these three scenarios.

2 Example 1: Simple Methods for Correlated Data

Standard regression models typically rely on the assumption that data are independent and identically distributed.

Suppose you had data on the BMI of 20 individuals. Let's say that these 20 individuals are picked randomly from the adult population in the United States.

Let's say one of the BMI values is 19.8 kg/m^2. Could you use this information to tell me anything about the other BMI values in the data?

Because of how these data were sampled, the answer to this question should be "no", you could not use this BMI to say anything about other BMI values in the data.

However, if I change the story a little, and told you that 10 of these data points were randomly selected women from the US Olympic Weightlifting Team (which included the 19.8 kg/m^2 data point), and the remaining ten were randomly selected women from Greene County, Alabama (the county with presumably the highest BMIs in the nation). On the basis of this information, you now know something more about what these data look like. You know that the BMI values from the Olympic Weightlifting Team will be closer to each other, and lower, than those from Greene County, Alabama. In effect, the BMI values in each cluster of ten individuals are correlated.

But what do we really mean when we say that outcomes are correlated? To help make some concrete points, let's simulate 20 individuals' BMI based on the scenario above (10 from the Olympic Team, and 10 from Green County). We can do this in R:

```
set.seed(123)

bmi_data <- tibble(BMI=c(rnorm(10,mean=20,sd=1),
                              rnorm(10,mean=34,sd=4)),
                      cluster=factor(c(rep(1,10),
                                       rep(2,10))))

bmi_data %>% print(n=3)
```

```
## # A tibble: 20 x 2
##     BMI cluster
##   <dbl> <fct>
## 1  19.4 1
## 2  19.8 1
## 3  21.6 1
## # i 17 more rows
```

The code above simulates two groups of 10 BMI values. The first are generated from a normal distribution with a mean of 20 and standard deviation of 1 (the US Olympic Team). The second are generated from a normal distribution with a mean of 34 and a standard deviation of 4 (Greene County). The histogram in Figure 1 shows the distribution of BMI in these data. There's clearly an important separation between the BMI's from the two groups (by design). And while this simple example may not be very realistic, it will help us show precisely what we mean by the terms "correlated data", "clustered data" and the like.

So how can we evaluate clustering in our data? Typically, we use the intra-cluster correlation coefficient to measure how correlated the data are in each cluster. For a continuous outcome variable like BMI, the ICC can be obtained using ANOVA, which is easy in R:

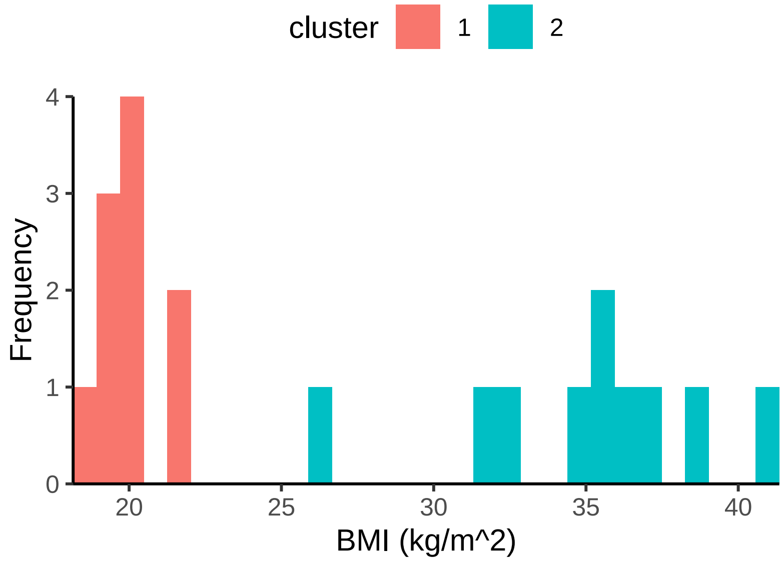

Figure 3: Histogram of Twenty BMI Values from Two Simulated Clusters

```
bmi_summary <- summary(aov(BMI ~ cluster, data = bmi_data))

bmi_summary
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## cluster      1 1089.3  1089.3     120 2.15e-09 ***
## Residuals   18  163.4     9.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
icc <- bmi_summary[[1]][1, 2]/sum(bmi_summary[[1]][, 2])

icc
```

```
## [1] 0.8695853
```

This tells us that roughly 87% of the variation in these data is occurring between the clusters level, which implies that within clusters, the magnitude of the variation is not as large. In other words, individuals in these data look more similar to one another within each cluster, and are quite different from each other across clusters. This is what we would have expected given how we simulated the data.

In contrast, look at what happens if we simulate a different dataset with the same type of individuals in each cluster (i.e., no clustering):

```
set.seed(123)

bmi_data <- tibble(BMI=rnorm(20,mean=25,sd=5),
                      cluster=factor(c(rep(1,10),
                                       rep(2,10))))

bmi_summary <- summary(aov(BMI ~ cluster,data=bmi_data))

icc <- bmi_summary[[1]][1,2]/sum(bmi_summary[[1]][,2])

icc
```

```
## [1] 0.004994307
```

In the above code, we simulated all 20 values from the same normal distribution with a mean of 25 and a standard deviation of 5. In other words, there is no (statistical) difference in the individuals across clusters. In this case, we obtain an ICC value of 0%. Finally, here's what happens if the cluster perfectly predicts BMI values:

```
set.seed(123)

bmi_data <- tibble(BMI=c(rep(25,10),rep(30,10)),
                      cluster=factor(c(rep(1,10),
                                       rep(2,10))))

bmi_summary <- summary(aov(BMI ~ cluster,data=bmi_data))

icc <- bmi_summary[[1]][1,2]/sum(bmi_summary[[1]][,2])

icc
```

```
## [1] 1
```

In the above code, we generated a BMI value of exactly 25 for all ten individuals in the first cluster, and a BMI value of exactly 30 for all ten individuals in the second cluster. Thus, cluster can perfectly predict the BMI value for everyone (i.e., there is no random variation), and we thus get an ICC value of 100%.

2.1 Are Correlated Data a Problem?

The answer to the question of whether correlated data are a problem depends entirely on the research question, and we are going to discuss the issues next. First, it's important to note that when we say "correlated data", what we typically refer too is correlated *outcome* data.

Specifically, suppose that your exposure of interest or your confounder adjustment set was highly correlated across several clusters, but the outcome you are studying is not correlated across these or any other clusters. If this is the case, you need not worry about correlated data. The problems with correlated data arise when the *outcome* under study is correlated. The specific "location" where this problem arises is in the process of trying to quantify the parameters of a model that we wish to fit. Take, for example, the following linear model

$$E(Y \mid X, C) = \beta_0 + \beta_1 X + \beta_2 C$$

Let's assume that in this example, Y represents BMI, X is some measure of diet (e.g., eat your vegetables versus don't eat your vegetables), and C is a confounder, and that we wanted to fit this model to the made up data in Table 1 (with only three observations, for simplicity):

ID	BMI (Y)	Vegetables (X)	Confounder (C)
1	21.0	0	1
2	32.7	1	0
3	25.8	1	1

Table 1: Some made up data for our likelihood function example

Typically, the objective here would be to get an estimate β_1, which we could interpret as a difference in BMI averages among those who eat their vegetables versus those who don't. An important statistical consideration is HOW we get these estimates. Many approaches exist, with one very common approach being maximum likelihood estimation.

> **Deeper Dive**:
> In introductory probability, you may have learned that the joint probability of two *independent* events [often denoted $P(A,B)$, and read "the probability of A and B] is equal to the product of their individual probabilities:
>
> $$P(A,B) = P(A) \times P(B)$$
>
> If, however, A and B are correlated, the above equation is no longer true. Instead, we'd have to use a more complicated form:
>
> $$P(A,B) = P(A \mid B) \times P(B)$$

After choosing a distribution and link function, maximum likelihood estimation proceeds by specifying a likelihood for each person in the data, and multiplying all of these individual likelihoods together:

$$\underbrace{L(y;\beta)}_{\text{joint likelihood}} = \overbrace{L(21.0;\beta_0,\beta_2) \times L(32.7;\beta_0,\beta_1) \times L(25.8;\beta_0,\beta_1,\beta_2)}^{\text{product of individual likelihoods}}$$

What your computer software program (i.e., SAS, Stata, R, other) does is find values for β_1, β_2, and β_3 in the product of likelihoods that make joint likelihood as large as it can be with the data we have.

Though likelihoods are not probabilities, the two do share some properties (Pawitan, 2001). Specifically, if the outcomes are correlated, you cannot break up the joint likelihood into the product of individual likelihoods as in the equation above.

In the next sections, we're going to discuss some of the more practical implications of the problem that result from correlated outcomes. Using real data, we going to look at some different ways we can address the problems that arise.

2.2 Example Data with Correlated Outcomes

Here, we'll introduce the datasets we'll be using to illustrate some methods for dealing with correlated data.

The first example dataset is from a cluster randomized trial example in which 10 practices were randomly assigned to two treatment groups (patient centered care and normal care). Body mass index (kg/m^2) measured at year 1 of follow-up was the outcome. These data are available and described in Campbell (2006), but I obtained them from Mansournia et al. (2020):

```
cluster_trial <- read_csv(here("data", "cluster_trial_data_bmi.csv"))

cluster_trial %>%
    print(n = 5)
```

```
## # A tibble: 20 x 4
##      ID   BMI treatment practice
##   <dbl> <dbl>     <dbl>    <dbl>
## 1     1  26.2         1        1
## 2     2  27.1         1        1
## 3     3  25           1        2
## 4     4  28.3         1        2
## 5     5  30.5         1        3
## # i 15 more rows
```

The second example dataset is from a longitudinal (repeated outcome measure) study of the effect of a lead chelating agent (succimer) on blood lead levels in children aged 12-33 months at enrollment. The data represent a random subset of 100 children from the original sample. Children were randomized at baseline to succimer or placebo, and blood lead levels were measured at weeks 0 (baseline), 1, 4, and 6. These data are available online,[2] and are described in Fitzmaurice et al. (2004):

```
lead_trial <- read_csv(here("data", "longitudinal_lead_data.csv"))

lead_trial <- gather(lead_trial, week, lead_value, L0:L6, factor_key = TRUE) %>%
    mutate(week = as.numeric(gsub("L", "", week))) %>%
    arrange(ID, week)

lead_trial %>%
    print(n = 8)
```

[2] https://content.sph.harvard.edu/fitzmaur/ala/tlc.txt

```
## # A tibble: 400 x 4
##      ID Treatment  week lead_value
##   <dbl> <chr>     <dbl>      <dbl>
## 1     1 P             0       30.8
## 2     1 P             1       26.9
## 3     1 P             4       25.8
## 4     1 P             6       23.8
## 5     2 A             0       26.5
## 6     2 A             1       14.8
## 7     2 A             4       19.5
## 8     2 A             6       21
## # i 392 more rows
```

We'll exclusively rely on the BMI data in this lecture (we won't have time to demonstrate with the longitudinal data). However, everything that I show you here can apply equivalently to either the BMI data or the lead data. If you are particularly interested, I'd encourage you to try to do the same analyses we present below with the longitudinal data.

2.3 Handling Correlated Outcome Data

We're going to focus today primarily on the BMI data. As a starting point, let's estimate the ICC to evaluate how correlated these outcomes are:

```
cluster_trial
```

```
## # A tibble: 20 x 4
##       ID   BMI treatment practice
##    <dbl> <dbl>     <dbl>    <dbl>
##  1     1  26.2         1        1
##  2     2  27.1         1        1
##  3     3  25           1        2
##  4     4  28.3         1        2
##  5     5  30.5         1        3
##  6     6  28.8         1        4
##  7     7  31           1        4
##  8     8  32.1         1        4
##  9      9  28.2         1        5
## 10     10  30.9         1        5
## 11     11  37           0        6
## 12     12  38.1         0        6
## 13     13  22.1         0        7
## 14     14  23           0        7
## 15     15  23.2         0        8
## 16     16  25.7         0        8
## 17     17  27.8         0        9
## 18     18  28           0        9
## 19     19  28           0       10
## 20     20  31           0       10
```

```
bmi_summary <- summary(aov(BMI ~ as.factor(practice),
                             data=cluster_trial)) # type II SS

icc <- bmi_summary[[1]][1,2]/sum(bmi_summary[[1]][,2])

icc
```

```
## [1] 0.9272896
```

With the BMI data, we get an intracluster correlation coefficient estimate of 93 indicating high levels of clustering in each practice.

So, with this high level clustering, the question is **what should we do about it?**

In the next sections, we'll explore what happens when we **ignore** clustered data, and how our results/conclusions compare when we use different methods to account for the clustering in the BMI trial data. These methods will include:

- Robust Standard Errors and Bootstrapping
- Generalized Estimating Equations
- Mixed Effects Models

The order of the techniques presented here is important. First, robust standard errors and (sometimes) the bootstrap are the **easiest** methods to implement when needing to deal with correlated outcomes. Generalized estimating equations are more complicated, and mixed effects models are most complicated. Second, the **assumptions** required for each of these methods to be valid generally increase in scope as we move down the list: robust standard errors and bootstrapping require generally fewer assumptions, while mixed effects models require the strongest set of assumptions.

Let’s proceed with the clustered BMI data analysis. Let’s conduct an analysis where we simply ignore the fact that the outcomes are correlated. We can fit a linear regression model, regressing BMI against the treatment. In R, we can do this using the `lm()` or `glm` functions. We can also use the `coefci` function from the `lmtest` package to easily get confidence intervals.

```
library(lmtest)

mod1 <- glm(BMI ~ treatment,
            data=cluster_trial,
            family=gaussian(link = "identity"))

coeftest(mod1)
```

```
##
## z test of coefficients:
##
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  28.3900     1.3466 21.0828   <2e-16 ***
## treatment     0.4200     1.9044  0.2205   0.8254
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
coefci(mod1, level = 0.95)
```

```
##                 2.5 %    97.5 %
## (Intercept) 25.750719 31.029281
## treatment   -3.312507  4.152507
```

The above analysis tells us that the difference in average BMI between the patient centered care and the normal care groups is 0.42 kg/m^2. The standard error for this estimate is 1.9, which results in a p-value of 0.83 and 95% normal-interval (Wald) confidence intervals of -3.31, 4.15. These results are what we obtain when we ignore the clustering.

2.4 Robust Standard Errors

The above analysis just ignores the clustering of BMI across practices in the data. The easiest way to account for correlated outcomes, in this case due to clustering across practices, is to use robust or sandwich standard errors. There are several ways to do this in R. We'll use the `sandwich` package to implement these standard errors, and the `lmtest` package to get confidence intervals that can be modified to account for clustering.

```
library(lmtest)
library(sandwich)

mod1 <- glm(BMI ~ treatment,
            data = cluster_trial,
            family = gaussian(link = "identity"))

coeftest(mod1, vcov=vcovCL(mod1,
                           type = "HC3",
                           cluster = cluster_trial$practice))
```

```
##
## z test of coefficients:
##
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  28.3900     2.9048  9.7734   <2e-16 ***
## treatment     0.4200     3.1147  0.1348   0.8927
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
coefci(mod1, vcov=vcovCL(mod1,
                         type = "HC3",
                         cluster = cluster_trial$practice),
        level = 0.95)
```

```
##                 2.5 %    97.5 %
## (Intercept) 22.696671 34.083329
## treatment   -5.684668  6.524668
```

What we see when we use methods to account for clustering is that the it is really the standard errors that are adjusted. This is particularly true when we use the robust variance estimator, or (as we will see) the bootstrap. It's also true with generalized estimating equations (GEE), but in the section on GEE, we'll also discuss how it's a bit more complicated.

It's important to discuss what can go wrong when we use the robust variance estimator. In particular, the most important threat to the validity of the robust variance estimator is small sample sizes. In our case, it's probably not a good idea to use the robust variance estimator we did (HC3). In R, small sample adjustments are implemented by default when using the `vcovCL()` function (Zei). Unfortunately, all of the methods we discuss here (robust variance, bootstrap, GEE, and mixed effects models) require "large" samples to get good performance.[3]

There are also many different variations of the robust variance estimator. These variations are typically referred to as HC1, HC2, ... HC5. If you're interested, there is a paper forthcoming by Mansournia et al in the IJE that explains these differences (Mansournia et al., 2020), and some more general features and properties of the robust variance estimator.

2.5 Clustered Bootstrap

Instead of using the robust standard error, a slightly more technical option is to use the **clustered** bootstrap. Essentially, the simplest clustered bootstrap approach proceeds in 4 steps:

1. Resample the data, with replacement, at the cluster level
2. Estimate the parameter of interest (in our case, the mean difference in BMI) and save the estimate
3. Repeat steps 1 and 2, 200 times

[3] This is almost universally true about any estimator in any setting. Ultimately, it depends on the complexity of the model/question that you have.

4. Take the standard deviation of all 200 estimates as the standard error of the estimate in the original (unsampled) data

In R, implementing the clustered bootstrap requires user written code to obtain the resampled data.[4] Here is some code that I've written in R to do a clustered bootstrap analysis:

```
mod1$coefficients
```

```
## (Intercept)   treatment
##       28.39        0.42
```

```
seed <- 123
set.seed(seed)

boot_func <- function(boot_num){

  clusters <- as.numeric(names(table(cluster_trial$practice)))
  index <- sample(1:length(clusters), length(clusters), replace=TRUE)
  bb <- table(clusters[index])
  boot <- NULL

  for(zzz in 1:max(bb)){
      cc <- cluster_trial[cluster_trial$practice %in% names(bb[bb %in% c(zzz:max(bb))]),]
      cc$b_practice<-paste0(cc$practice,zzz)
      boot <- rbind(boot, cc)
  }

  mod1 <- glm(BMI ~ treatment, data=boot,family=gaussian(link = "identity"))
  res <- cbind(boot_num,coef(mod1)[2])
  return(res)
}

boot_res <- lapply(1:750, function(x) boot_func(x))
boot_res <- do.call(rbind,boot_res)
```

[4] The `boot` package in R provides a range of boostrap estimators, but not for clustered data.

```
head(boot_res)
```

```
##            boot_num
## treatment        1 -2.922222
## treatment        2  2.925000
## treatment        3  2.966667
## treatment        4  1.408333
## treatment        5 -2.761111
## treatment        6 -4.193636
```

```
tail(boot_res)
```

```
##            boot_num
## treatment      745  1.7333333
## treatment      746  0.1488095
## treatment      747 -2.3227273
## treatment      748  1.3000000
## treatment      749 -0.5800000
## treatment      750  1.0450000
```

```
sd(boot_res[,2]) ## standard error of the treatment estimate
```

```
## [1] 2.677977
```

We can then use the standard normal-interval (Wald) estimator to get 95% confidence intervals with this bootstrapped standard error:

```
LCL <- mod1$coefficient[2] - 1.96*sd(boot_res[,2])
UCL <- mod1$coefficient[2] + 1.96*sd(boot_res[,2])

mod1$coefficient[2]
```

```
## treatment
##      0.42
```

```
LCL
```

```
## treatment
## -4.828835
```

```
UCL
```

```
## treatment
##  5.668835
```

What the above bootstrap code does is select, with replacement, practices in the `cluster_trial` data. Specifically, as we saw above, here's what the `cluster_trial` data look like:

```
head(cluster_trial)
```

```
## # A tibble: 6 x 4
##      ID   BMI treatment practice
##   <dbl> <dbl>     <dbl>    <dbl>
## 1     1  26.2         1        1
## 2     2  27.1         1        1
## 3     3  25           1        2
## 4     4  28.3         1        2
## 5     5  30.5         1        3
## 6     6  28.8         1        4
```

```
table(cluster_trial$practice)
```

```
##
##  1  2  3  4  5  6  7  8  9 10
##  2  2  1  3  2  2  2  2  2  2
```

The bootstrap code above randomly selects 10 practices from these data to create a "bootstrap resample". This bootstrap resample will contain 10 practices, but in the resample, some of the original practices may not be present, while others may be in the resample more than once. For example:

```
clusters <- as.numeric(names(table(cluster_trial$practice)))

index <- sample(1:length(clusters), length(clusters), replace=TRUE)

bb <- table(clusters[index])
boot <- NULL

for(zzz in 1:max(bb)){
    cc <- cluster_trial[cluster_trial$practice %in% names(bb[bb %in% c(zzz:max(bb))]),]
    cc$b_practice<-paste0(cc$practice,zzz)
    boot <- rbind(boot, cc)
}

head(boot)
```

```
## # A tibble: 6 x 5
##      ID   BMI treatment practice b_practice
##   <dbl> <dbl>     <dbl>    <dbl> <chr>
## 1     1  26.2         1        1 11
## 2     2  27.1         1        1 11
## 3     3  25           1        2 21
## 4     4  28.3         1        2 21
## 5     5  30.5         1        3 31
## 6     9  28.2         1        5 51
```

```
table(boot$b_practice)
```

```
##
## 101  11  21  31  32  33  51  61  62  91
##   2   2   2   1   1   1   2   2   2   2
```

By resampling this way, the "within-practice" correlation structure is respected, and we are thus able to obtain standard errors that appropriately account for clustering.

Again, it's important to discuss what can go wrong when we use a clustered bootstrap estimator. In my experience, many applied researchers are under the impression that the bootstrap does not require large samples to be valid. While there is simulation evidence that shows performance of the bootstrap is certainly better than the robust variance estimator (e.g., Cameron et al., 2008), the theoretical validity of the bootstrap still rests on large sample (i.e., asymptotic) arguments. Nevertheless, in applied settings similar to what we encountered in the `cluster_trial` data (specifically, when there are fewer then 50 clusters), my preference would be to use the clustered bootstrap.

Similar to the robust variance estimator, there are also many different variations of the bootstrap variance estimator. Among the most important versions of these is the bias-corrected bootstrap, and the bias-corrected and accelerated bootstrap (Davison and Hinkley, 1997). These two variations have been shown to perform better than the normal interval boostrap (or the percentile bootstrap) in a range of settings. Unfortunately, these are much more complicated to code by hand. Thus, for the time being, I almost always rely on the normal-interval bootstrap when dealing with clustered data.

2.6 Summary of Results So Far

```
res <- data.frame(
  Version = c("Uncorrected", "Cluster Robust", "Cluster Bootstrap"),
  Estimate = c(coeftest(mod1)[2,1],
               coeftest(mod1)[2,1],
               coeftest(mod1)[2,1]),
  Std.Err = c(coeftest(mod1)[2,2],
              coeftest(mod1, vcov=vcovCL(mod1,type = "HC",
                                         cadjust = F,
                                         cluster = cluster_trial$practice))[2,2],
              sd(boot_res[,2])),
  LCL = c(coefci(mod1, level = 0.95)[2,1],
          coefci(mod1, vcov=vcovCL(mod1,type = "HC",
                                    cadjust = F,
                                    cluster = cluster_trial$practice), level = 0.95)[2,1],
          LCL),
  UCL = c(coefci(mod1, level = 0.95)[2,2],
          coefci(mod1, vcov=vcovCL(mod1,type = "HC",
                                   cadjust = F,
                                   cluster = cluster_trial$practice), level = 0.95)[2,2],
          UCL)
)

knitr::kable(res, digits = 2)
```

Version	Estimate	Std.Err	LCL	UCL
Uncorrected	0.42	1.90	-3.31	4.15
Cluster Robust	0.42	2.47	-4.43	5.27
Cluster Bootstrap	0.42	2.68	-4.83	5.67

3 Example 2: GEE and Mixed Effects Models for Correlated Data

Generalized estimating equations (GEEs) are another method we can use to adjust our generalized linear model to account for a lack of independence, and recover the interpretation of the p-values, confidence intervals, and standard errors of interest. This was the focus of the paper by Liang and Zeger (Liang and Zeger, 1986), who originally introduced the concept. Their “extension” did just that: adjusted the GLM by incorporating information on the correlation structure within individual units. This extension generalized the GLM using a theory of estimating equations, and the new method was hence named generalized estimating equations.

The main distinction between deploying GEEs versus GLMs is that, in the former, we have to consider the structure of the correlation within units in our data. Several correlation structures exist, and include things like the independence, exchangeable, unstructured, or variations of autoregression correlation matrices. In R, these methods can be deployed using the `geepack` library.

Let’s again use our BMI data as we did with the robust variance and clustered bootstrap above. These data can then be analyzed using the `geeglm` functions in the `geepack` library:

```
#install.packages("geepack")
library(geepack)
```

```r
## use GEE
mod1_ind <- geeglm(BMI ~ treatment,
                   family = gaussian(link = "identity"),
                   id = factor(practice),
                   data=cluster_trial,
                   scale.fix = T,
                   corstr="independence")

summary(mod1_ind)$coefficients
```

```
##             Estimate Std.err         Wald  Pr(>|W|)
## (Intercept)    28.39 2.32385 149.25005740 0.0000000
## treatment       0.42 2.47451   0.02880846 0.8652221
```

```r
mod1_exch <- geeglm(BMI ~ treatment,
                   family = gaussian(link = "identity"),
                   id = factor(practice),
                   data=cluster_trial,
                   scale.fix = T,
                   corstr="exchangeable")

summary(mod1_exch)$coefficients
```

```
##               Estimate  Std.err         Wald  Pr(>|W|)
## (Intercept) 28.3900000 2.323850 149.25005740 0.0000000
## treatment    0.3862169 2.459032   0.02466802 0.8751971
```

```r
mod1_unstr <- geeglm(BMI ~ treatment,
                   family = gaussian(link = "identity"),
                   id = factor(practice),
                   data=cluster_trial,
                   scale.fix = T,
                   corstr="unstructured")
```

```
summary(mod1_unstr)$coefficients
```

```
##               Estimate  Std.err         Wald  Pr(>|W|)
## (Intercept) 28.3900000 2.323850 149.25005740 0.0000000
## treatment    0.9224488 5.157636   0.03198771 0.8580546
```

```
QIC(mod1_ind, mod1_exch, mod1_unstr)
```

```
##                 QIC     QICu Quasi Lik       CIC params     QICC
## mod1_ind   333.9020 330.3980 -163.1990  3.751985      2 335.6163
## mod1_exch  333.8198 330.4094 -163.2047  3.705192      2 337.8198
## mod1_unstr 361.5223 332.9225 -164.4613 16.299861      2 376.5223
```

“An attractive property of the GEE is that one can use some working correlation structure that may be wrong, but the resulting regression coefficient estimate is still consistent and asymptotically normal.” https://www3.stat.sinica.edu.tw/statistica/oldpdf/a12n26.pdf

“one does not even have to model the correlation structure of the response variable correctly; one only needs to use some working correlation structure to obtain consistent and asymptotically normal estimates”

4 Example 3: G Methods for Complex Longitudinal Data

Robins’ g methods enable the identification and estimation of the effects of generalized treatment, exposure, or intervention plans. G methods are a family of methods that include the g formula, marginal structural models, and structural nested models.[5] They provide **consistent** estimates of contrasts (e.g. differences, ratios) of average potential outcomes under a less restrictive set of identification conditions than standard regression methods (e.g. linear, logistic, Cox regression) (Robins and Hernán, 2009). Specifically, standard regression **requires no feedback between time-varying treatments and time-varying confounders, while g methods do not.** Robins and Hern'{a}n Robins and Hernán (2009) have provided a technically comprehensive worked example of each of the three g methods. Here, we present a corresponding worked

[5] There are three g methods: the parametric g formula and inverse probability weighting. These two are used to estimate the parameters of a marginal structural model. Then there is g estimation (different from the g formula). This is used to estimate the parameters of a strcutural nested model.

example that illustrates the need for and use of g methods, while minimizing technical details.[6]

Our research question concerns the effect of treatment for HIV on CD4 count. Table 1 presents data from a hypothetical observational cohort study ($A = 1$ for treated, $A = 0$ otherwise). Treatment is measured at baseline (A_0) and once during follow up (A_1). The sole covariate is elevated HIV viral load ($Z = 1$ for those with > 200 copies/ml, $Z = 0$ otherwise), which is constant by design at baseline ($Z_0 = 1$) and measured once during follow up just prior to the second treatment (Z_1). The outcome is CD4 count measured at the end of follow up in units of cells/mm^3. The CD4 outcome in Table 1 is summarized (averaged) over the participants at each level of the treatments and covariate.

A_0	Z_1	A_1	Y	N
0	0	0	87.29	209,271
0	0	1	112.11	93,779
0	1	0	119.65	60,654
0	1	1	144.84	136,293
1	0	0	105.28	134,781
1	0	1	130.18	60,789
1	1	0	137.72	93,903
1	1	1	162.83	210,527

Table 2: Prospective study data illustrating the number of subjects (N) within each possible combination of treatment at time 0 (A_0), HIV viral load just prior to the second round of treatment (Z_1), and treatment status for the 2nd round of treatment (A_1). The outcome column (Y) corresponds to the mean of Y within levels of A_0, Z_1, A_1. Note that HIV viral load at baseline is high ($Z_0 = 1$) for everyone by design.

The number of participants is provided in the rightmost column of Table 1. In this hypothetical study of one million participants we ignore random error and focus on identifying the parameters defining our causal effect of interest, which we describe next.

Based on Figure 2, the average outcome in our simple data generating structure may be composed of several parts: the effects of A_0, Z_1, and A_1; the two-way interactions between A_0 and Z_1, A_0 and A_1, and A_1 and Z_1; and the three-way interaction between A_0, Z_1, and A_1. These components (some whose magnitudes may be zero) can be used to “build up’ ’ a contrast of substantive interest. Here, we focus on the average causal effect of always taking treatment ($a_0 = 1, a_1 = 1$) compared to never taking treatment ($a_0 = 0, a_1 = 0$),[7]

[6] There are a handful of worked examples and tutorials on the use of g methods to estimate effects in complex longitudinal data. These include Robins and Hernán (2009), Daniel et al. (2013), Keil et al. (2014), the paper on which these notes are based Naimi et al. (2017). Additionally, **?** is an excellent, comprehensive, and very accessible introduction to causal inference generally, and g methods specifically.

[7] Alternate notation for potential outcomes includes: Y_x, $Y(x), Y \mid Set(X = x)$, and $Y|do(X = x)$.

$$
\begin{aligned}
\psi &= E(Y^{a_0=1,a_1=1}) - E(Y^{a_0=0,a_1=0}) \\
&= E(Y^{a_0=1,a_1=1} - Y^{a_0=0,a_1=0}),
\end{aligned}
\tag{1}
$$

where expectations $E(\cdot)$ are taken with respect to the target population from which our sample is a random draw. This average causal effect consists of the joint effect of A_0 and A_1 on Y Daniel et al. (2013). Here, Y^{a_0,a_1} represents a potential outcome value that would have been observed had the exposures been set to specific levels a_0 and a_1. This potential outcome is distinct from the observed (or actual) outcome.[8]

This average causal effect $\psi = E(Y^{a_0,a_1} - Y^{0,0})$ is a *marginal* effect because it averages (or marginalizes) over all individual-level effects in the population. We can write this effect as $E(Y^{a_0,a_1} - Y^{0,0}) = \psi_0 a_0 + \psi_1 a_1 + \psi_2 a_0 a_1$, which states that our average causal effect ψ may be composed of two exposure main effects (e.g., ψ_0 and ψ_1) and their two-way interaction (ψ_2). This marginal effect ψ is indifferent to whether the A_1 component ($\psi_1 + \psi_2$) is modified by Z_1: whether such effect modification is present or absent, the marginal effect represents a meaningful answer to the question: what is the effect of A_0 and A_1 in the entire population?

Alternatively, we may wish to estimate this effect *conditional* on certain values of another covariate. A conditional effect would arise if, for example, one was specifically interested in effect measure modification by Z_1. When properly modeled, this conditional effect represents a meaningful answer to the question: what is the effect of A_0 and A_1 in those who receive $Z_1 = 1$ versus those who receive $Z_1 = 0$? Modeling such effect measure modification by time-varying covariates is the fundamental issue that distinguishes marginal structural from structural nested models. We thus return to this issue later. For simplicity, we define our effect of interest as $\psi = \psi_0 + \psi_1 + \psi_2$, and we explore a data example with no effect modification by time-varying confounders.

4.1 Assumptions

Our average causal effect is defined as a function of two averages that would be observed if everybody in the population were exposed (or unexposed) at both time points. Yet we cannot directly acquire information on these averages because in any given sample, some individuals will be unexposed (or exposed). Part of our task therefore involves justifying use of averages among subsets

[8] Note this distinction is subtle, and often overlooked. Importantly, one can only equate the potential outcome with the observed outcome under the observed exposure if **counterfactual consistency** holds.

of the population as what would be observed in the whole population.[9] This is accomplished by making three main assumptions.

Counterfactual consistency (Cole and Frangakis, 2009) allows us to equate observed outcomes among those who received a certain exposure value to the potential outcomes that would be observed under the same exposure value:

$$E(Y \mid A_0 = a_0, A_1 = a_1) = E(Y^{a_0,a_1} \mid A_0 = a_0, A_1 = a_1)$$

The status of this assumption remains unaffected by the choice of analytic method (e.g., standard regression versus g methods). Rather, this assumption's validity depends on the nature of the exposure assignment mechanism (VanderWeele and Hernán, 2013). Under counterfactual consistency, we partially identify our average causal effect.

Next, we assume exchangeability (Greenland and Robins, 1986). Exchangeability implies that the potential outcomes under exposures a_0 and a_1 (denoted Y^{a_0,a_1}) are independent of the actual (or observed) exposures A_0 and A_1. We make this exchangeability assumption within levels of past covariate values (conditional) and at each time point separately (sequential):

$$\begin{aligned} &E(Y^{a_0,a_1} \mid A_1, Z_1, A_0) = E(Y^{a_0,a_1} \mid Z_1, A_0), \text{ and} \\ &E(Y^{a_0,a_1} \mid A_0) = E(Y^{a_0,a_1}). \end{aligned} \tag{2}$$

This sequential conditional exchangeability assumption would hold if there were no uncontrolled confounding and no selection bias. The top part of equation 2 says that, within levels of prior viral load (Z_1) and a given treatment level A_0, Y^{a_0,a_1} does not depend on the assigned values of A_1. The bottom part of equation 2 says that Y^{a_0,a_1} does not depend on the assigned values of A_0. Note the correspondence between these two equations and the causal diagram: because in Figure 1, Z_1 is a common cause of A_1 and Y, the assumption in equation 2 must be made conditional on Z_1. Failing to condition for Z_1 will result in uncontrolled confounding of the effect of A_1, and thus a dependence between the actual A_1 value and the potential outcome. However, adjusting for Z_1 using standard methods (restriction, stratification, matching, or conditioning in a linear regression model) would block part of the effect from A_0 through Z_1, and potentially lead to a collider bias of the effect of A_0 through U (Cole et al., 2010) This is the central challenge that g methods were developed to address.

[9] Understanding what this justification entails is the fundamental charge of causal inference.

The third assumption, known as positivity (Westreich and Cole, 2010) requires $0 < P(A_1 = 1 \mid Z_1 = z_1, A_0 = a_0) < 1$ and $0 < P(A_0 = 1) < 1$. Furthermore, this assumption must hold for all values of a_0 and z_1 where $P(A_0 = a_0, Z_1 = z_1) > 0$. This latter condition is required so that effects are not defined in strata of a_0 and z_1 that do not exist. Positivity is met when there are exposed and unexposed individuals within all confounder and prior exposure levels, which can be evaluated empirically.[10]

Under these three assumptions, our hypothetical observational study can be likened to a sequentially randomized trial in which the exposure was randomized at baseline, and randomized again at time 1 with a probability that depends on Z_1. Under these assumptions, g methods can be used to estimate counterfactual quantities with observational data.

5 Results

5.1 Standard Methods

Table 2 presents results from fitting a number of standard linear regression models to the data in Table 1.

Model Parameters	Estimate ($\widehat{\beta}_1$)
$\beta_0 + \beta_1(A_0 + A_1)/2$	60.9
$\beta_0 + \beta_1(A_0 + A_1)/2 + \beta_2 Z_1$	42.6
$\beta_0 + \beta_1 A_0$	27.1
$\beta_0 + \beta_1 A_0 + \beta_2 Z_1$	18.0
$\beta_0 + \beta_1 A_1$	38.9
$\beta_0 + \beta_1 A_1 + \beta_2 Z_1$	25.0

Table 3: Linear regression models and corresponding estimates comparing several contrasts quantifying exposed versus unexposed scenarios fit to data in Table 1.

In the first model, $\hat{\beta} = 60.9$ cells/mm^3 is the crude difference in mean CD4 count for the always treated compared to the never treated. In model two, $\hat{\beta} = 42.6$ cells/mm^3 is the Z_1-adjusted difference in mean CD4 count for the same contrast. Other model results are provided in Table 2, and more could be entertained.

Table 3 presents the results from fitting all three g methods to the data in Table 1.

The marginal structural model resulted in $\hat{\psi} = 50.0$ cells/mm^3. The g formula resulted in $\hat{\psi} = 50.0$ cells/mm^3. Finally, the structural nested model

[10] There are actually two types of positivity violations: stochastic and structural. In the former, one need only collect more data to alleviate concerns over stochastic positivity violations. In the latter, certain confounder values preclude the possibility of individuals being exposed or unexposed. One example of the latter is the healthy worker survivor effect.

G Method	$\hat{\psi}^a$
G Formula	50.0
IP-weighted marginal structural model	50.0
G Estimated Structural Nested Model	50.0

a $\psi = E(Y^{1,1} - Y^{0,0})$

Table 4: G-methods and corresponding estimates comparing contrasts quantifying always exposed versus never exposed scenarios fit to data in Table 1.

resulted in $\hat{\psi} \;=\; 50.0$ cells/mm^3. Next we discuss how we obtained these results.

5.2 g Methods

The **g formula** can be used to estimate the average CD4 level that would be observed in the population under a given treatment plan. To implement the approach, we start with a mathematical representation of the data generating mechanism for all variables in Table 1. We refer to this as the joint density of the observed data. We factor the joint density in a way that respects the temporal ordering of the data by conditioning each variable on its history. For example, if $f(\cdot)$ represents the probability density function, then by the definition of conditional probabilities (Wasserman, 2006, p 36) we can factor this joint density as

$$\begin{aligned} f(y, a_1, z_1, a_0) = f(y \mid a_1, z_1, a_0) P(A_1 = a_1 \mid Z_1 = z_1, A_0 = a_0) \\ P(Z_1 = z_1 \mid A_0 = a_0) P(A_0 = a_0). \end{aligned}$$

Our interest lies in the marginal mean of Y that would be observed if A_0 and A_1 were set to some values a_0 and a_1, respectively. To obtain this expectation, we perform two mathematical operations on the factored joint density. The first is the well-known expectation operator (Wasserman, 2006, p 47), which allows us to write the conditional mean of Y in terms of its conditional density. The second is the law of total probability (Wasserman, 2006, p 12), which allows us to marginalize over the distribution of A_1, Z_1 and A_0, yielding the marginal mean of Y:

$$\begin{aligned} E(Y) = \sum_{a_1, z_1, a_0} E(Y \mid A_1 = a_1, Z_1 = z_1, A_0 = a_0) P(A_1 = a_1 \mid Z_1 = z_1, A_0 = a_0) \\ P(Z_1 = z_1 \mid A_0 = a_0) P(A_0 = a_0). \end{aligned}$$

We can now modify this equation to yield the average of potential outcomes that would be observed after intervening on the exposure [enabling us to drop out the terms for $P(A_1 = a_1 \mid Z_1 = z_1, A_0 = a_0)$ and $P(A_0 = a_0)$], yielding

$$E(Y^{a_0,a_1}) = \sum_{z_1} E(Y \mid A_1 = a_1, Z_1 = z_1, A_0 = a_0) P(Z_1 = z_1 \mid A_0 = a_0).$$

This equation is the g formula. Its proof, given in the Supplementary Material of Naimi et al (2017), follows from the three identifying assumptions. In our simple scenario, the expectation $E(Y^{0,0})$ can be calculated by summing the mean CD4 count in the never treated with $Z_1 = 1$ (weighted by the proportion of people with $Z_1 = 1$ in the $A_0 = 0$ stratum) and the mean CD4 count in the never treated with $Z_1 = 0$ (weighted by the proportion of people with $Z_1 = 0$ in the $A_0 = 0$ stratum). Weighting the observed outcome's conditional expectation by the conditional probability that $Z_1 = z_1$ enables us to account for the fact that Z_1 is affected by A_0, but also confounds the effect of A_1 on Y. Computing this expectation's value yields a result of $\hat{E}(Y^{0,0}) = 100.0$, where we use $\hat{E}$ to denote a sample, rather than a population average, and with the understanding that $\hat{E}(Y^{0,0})$ is equal to the g formula with $A_0 = A_1 = 0$ (since the potential outcomes $Y^{0,0}$ are not directly observed). We repeat the process to obtain the corresponding value for treated at time 0 only: $\hat{E}(Y^{1,0}) = 125.0$; treated at time 1 only: $\hat{E}(Y^{0,1}) = 125.0$; and always treated: $\hat{E}(Y^{1,1}) = 150.0$. Thus, $\hat{\psi}_{GF} = 150.0 - 100.0 = 50.0$, which is the average causal effect of treatment on CD4 cell count.

This approach to computing the value of the g formula is referred to as nonparametric maximum likelihood estimation. Several authors (Taubman et al., 2009, Westreich et al. (2012), Cole et al. (2013), Keil et al. (2014), Edwards et al. (2014)) demonstrate how simulation from parametric regression models can yield a g formula estimator, which is often required in typical population-health studies with many covariates.

Modeling each component of the joint density of the observed data (including the probability that $Z_1 = z_1$) can lead to bias if any of these models are mis-specified.[11] To compute the expectations of interest, we can instead specify a single model that targets our average causal effect, and avoid unnecessary modeling. Marginal structural models with IP weighting map a *marginal summary* (e.g., average) of potential outcomes to the treatment and parameter of interest ψ. Unlike the g formula, they do not require a model for $P(Z_1 = z_1 \mid A_0 = a_0)$. Additionally, as we show in the Supplementary Mate-

[11] One of the major limitations of the parametric g formula.

rial of Naimi et al (2017), while they cannot model it directly, they are indifferent to whether time-varying effect modification is present or absent. Because our interest lies in the marginal contrast of outcomes under always versus never treated conditions, our marginal structural model for the effect of A can be written as $E(Y^{a_0,a_1}) = \beta_0 + \psi_0 a_0 + \psi_1 a_1 + \psi_2 a_0 a_1$, where $\beta_0 = E(Y^{0,0})$ is a (nuisance) intercept parameter, and $\psi = E(Y^{1,1} - Y^{0,0}) = (\psi_0 + \psi_1 + \psi_2)$ is the effect of interest.

Inverse probability weighting can be used estimate marginal structural model parameters (proofs are provided in the Supplementary Material). To estimate ψ using inverse probability weighted regression, we first obtain the predicted probabilities of the observed treatments. In our example data, there are two possible A_1 values (exposed, unexposed) for each of the four levels in Z_1 and A_0. Additionally, there are two possible A_0 values (exposed, unexposed) overall. This leads to four possible exposure regimes: never treat, treat early only, treat late only, and always treat. For each Z_1 value, we require the predicted probability of the exposure that was actually received. These probabilities are computed by calculating the appropriate proportions of subjects in Table 1. Because there are no variables that affect A_0, this probability is 0.5 for all individuals in the sample. Furthermore, in our example A_1 is not affected by A_0 (Figure 1). Thus, the Z_1 specific probabilities of A_1 are constant across levels of A_0. In settings where A_0 affects A_1, the Z_1 specific probabilities of A_1 would vary across levels of A_0.

In the stratum defined by $Z_1 = 1$, the predicted probabilities of $A_1 = 0$ and $A_1 = 1$ are 0.308 and 0.692, respectively. For example, $(210,527 + 136,293)/(210,527 + 136,293 + 93,903 + 60,654) = 0.692$. Thus, the probabilities for each treatment combination are: $0.5 \times 0.308 = 0.155$ (never treated), $0.5 \times 0.308 = 0.155$ (treated early only), $0.5 \times 0.692 = 0.346$ (treated late only), and $0.5 \times 0.692 = 0.346$ (always treated). Dividing the marginal probability of each exposure category (not stratified by Z_1) by these stratum specific probabilities gives stabilized weights of 1.617, 1.617, 0.725, and 0.725, respectively. For example, the never treated weight is $(0.5 \times 0.501)/(0.5 \times 0.308) = 1.617$. The same approach is taken to obtain predicted probabilities and stabilized weights in the stratum defined by $Z_1 = 0$. The weights and weighted data are provided in Table 4.

Fitting this model in the weighted data given in Table 4 provides the inverse-

A_0	Z_1	A_1	Y	sw	Pseudo N
0	0	0	87.23	0.72	151222.84
0	0	1	112.23	1.62	151680.46
0	1	0	119.79	1.62	98110.06
0	1	1	144.78	0.72	98789.4
1	0	0	105.25	0.72	97395.08
1	0	1	130.25	1.62	98321.62
1	1	0	137.8	1.62	151884.02
1	1	1	162.8	0.72	152596.51

Table 5: Pseudo-population obtained after applying inverse probability weights to data in Table 1.

probability weighted estimates $[\hat{\psi}_{0_{IP}} = 25.0, \hat{\psi}_{1_{IP}} = 25.0, \hat{\psi}_{2_{IP}} = 0.0]$, thus yielding $\hat{\psi}_{IP} = 50.0$.

Weighting the observed data by the inverse of the probability of the observed exposure yields a “pseudo-population” (Table 4) in which treatment at the second time point (A_1) is no longer related to (and is thus no longer confounded by) viral load just prior to the second time point (Z_1). Thus, weighting a conditional regression model for the outcome by the inverse probability of treatment enables us to account for the fact that Z_1 both confounds A_1 and is affected by A_0.

Structural nested models map a *conditional contrast* of potential outcomes to the treatment, within nested sub-groups of individuals defined by levels of A_1, Z_1, and A_0. Our structural nested model can be written as

$$\begin{aligned}
&E(Y^{a_0,a_1} - Y^{a_0,0} \mid A_0 = a_0, Z_1 = z_1, A_1 = a_1) = a_1(\psi_1 + \psi_2 a_0 + \psi_3 z_1 + \psi_4 a_0 z_1) \\
&E(Y^{a_0,0} - Y^{0,0} \mid A_0 = a_0) = \psi_0 a_0
\end{aligned} \tag{3}$$

Note this model introduces two additional parameters: ψ_3 for the two-way interaction between a_1 and z_1, and ψ_4 for the three-way interaction between a_1, z_1, and a_0. Indeed, the ability to explicitly quantify interactions between time-varying exposures and time-varying covariates (which cannot be modeled via standard marginal structural models) is a major strength of structural nested models when effect modification is of interest.@Robins2009} To simplify our exposition, we set $(\psi_3, \psi_4) = (0, 0)$ in our data example, allowing us to drop the $\psi_3 z_1$ and $\psi_4 a_0 z_1$ terms from the model. In effect, this renders our structural nested mean model equivalent to a semi-parametric marginal structural model. In the Supplementary Material, we explain how marginal structural and structural nested models each relate to time-varying interactions in more detail.

We can now use g-estimation to estimate (ψ_0, ψ_1, ψ_2) in the above structural nested model. G-estimation is based on solving equations that directly result from the sequential conditional exchangeability assumptions in (2) and (**??**), combined with assumptions implied by the structural nested model. If, at each time point, the exposure is conditionally independent of the potential outcomes (sequential exchangeability) then the conditional covariance between the exposure and potential outcomes is zero.@Vansteelandt2015} Formally, these conditional independence relations can be written as:

$$\begin{aligned} 0 &= \text{Cov}(Y^{a_0,0}, A_1 \mid Z_1, A_0) \\ &= \text{Cov}(Y^{0,0}, A_0) \end{aligned} \tag{4}$$

where $\text{Cov}(\cdot)$ is the well-known covariance formula (Wasserman, 2006)$^{(p52)}$. These equalities are of little direct use for estimation, though, as they contain unobserved potential outcomes and are not yet functions of the parameters of interest. However, by counterfactual consistency and the structural nested model, we can replace these unknowns with quantities estimable from the data.

Specifically, as we prove in the Supplementary Material, the structural nested model, together with exchangeability and counterfactual consistency imply that we can replace the potential outcomes $Y^{a_0,0}$ and $Y^{0,0}$ in the above covariance formulas with their values implied by the structural nested model, yielding:

$$\begin{aligned} 0 &= \text{Cov}\{Y - A_1(\psi_1 + \psi_2 A_0), A_1 \mid Z_1, A_0\} \\ &= \text{Cov}\{Y - A_1(\psi_1 + \psi_2 A_0) - \psi_0 A_0, A_0\}. \end{aligned} \tag{5}$$

We provide an intuitive explanation for this substitution in the Supplementary Material. %is that it would certainly hold under a stronger version of our structural nested model assumptions, in which $Y^{a_0,a_1} - Y^{a_0,0} = a_1(\psi_1 + \psi_2 a_0)$ and $Y^{a_0,0} - Y^{0,0} = \psi_0 a_0$ exactly, so that $Y^{A_0,0} = Y - A_1(\psi_1 + \psi_2 A_0)$ and $Y^{0,0} = Y - A_1(\psi_1 + \psi_2 A_0) - \psi_0 A_0$. We also show how these covariance relations yield three equations that can be used to solve each of the unknowns in the above structural nested model (ψ_0, ψ_1, ψ_2).

Two of the three equations yield the following g estimators:

$$\begin{aligned}\hat{\psi}_{1_{GE}} &= \frac{\hat{E}[(1-A_0)Y\{A_1-\hat{E}(A_1 \mid Z_1, A_0)\}]}{\hat{E}[(1-A_0)A_1\{A_1-\hat{E}(A_1 \mid Z_1, A_0)\}]}\\ \hat{\psi}_{1_{GE}}+\hat{\psi}_{2_{GE}} &= \frac{\hat{E}[A_0Y\{A_1-\hat{E}(A_1 \mid Z_1, A_0)\}]}{\hat{E}[A_0A_1\{A_1-\hat{E}(A_1 \mid Z_1, A_0)\}]}\end{aligned} \qquad (6)$$

Note that to solve these equations we need to model $E(A_1 \mid Z_1, A_0)$, which in practice we might assume can be correctly specified as the predicted values from a logistic model for A_1. In our simple setting, the correctness of this model is guaranteed by saturating it (i.e., conditioning the model on Z_1, A_0 and their interaction).

As we show in the Supplementary Material, implementing these equations in software can be easily done using either an instrumental variables (i.e., two-stage least squares) estimator, or ordinary least squares.

Once the above parameters are estimated, the next step is to subtract the effect of A_1 and A_1A_0 from Y to obtain $\widetilde{Y} = Y - \hat{\psi}_{1_{GE}}A_1 - \hat{\psi}_{2_{GE}}A_1A_0$. We can then solve for the last parameter using a sample version of the third g estimation equality, yielding our final estimator and completing the procedure:

$$\hat{\psi}_{0_{GE}} = \frac{\hat{E}[\widetilde{Y}\{A_0-\hat{E}(A_0)\}]}{\hat{E}[A_0\{A_0-\hat{E}(A_0)\}]}.$$

Again the above estimator can be implemented using an instrumental variable or ordinary least squares estimator. Implementing this procedure in our example data, we obtain $[\psi_{0_{GE}} = 25.0, \psi_{1_{GE}} = 25.0, \psi_{2_{GE}} = 0.0]$, thus yielding $\psi_{GE} = 50.0$.

The potential outcome under no treatment can be thought of as a given subject's baseline prognosis: in our setting, individuals with poor baseline prognosis will have low CD4 levels, no matter what their treatment status may be. In the absence of confounding or selection bias, one expects this baseline prognosis to be independent of treatment status. G estimation exploits this independence by assuming no uncontrolled confounding (conditional on measured confounders), and assigning values to $\hat{\psi}_{GE}$ that render the potential outcomes independent of the exposure. However, assigning the correct values to $\hat{\psi}_{GE}$ depends on there being no confounding or selection bias.

6 Concluding Remarks

Having constructed these data using the causal diagram shown in Figure 1, we know the true effect of combined treatment is indeed 50 cells/mm^3 $(25$ cells/mm^3 for each exposure main effect) as well approximated by all three g methods, but not by any of the standard regression models we fit, with one exception. The final standard result presented in Table 2 correctly estimates the effect of the second treatment (an effect of 25 cells/mm^3), as would be expected from the causal diagram.

For the past several years, we have used the foregoing simple example to initiate epidemiologists to g methods with some success. Once having studied this simple example in detail, we recommend working through more comprehensive examples by Robins and Hern'{a}n Robins and Hernán (2009) and Hern'{a}n and Robins Hernán and Robins (Forthcoming). A recent tutorial Daniel et al. (2013) may then be of further use. G methods are becoming more common in epidemiologic research (Suarez et al., 2011). We hope this commentary facilitates the process of better understanding these useful methods.

References

Various Versatile Variances: An Object-Oriented Implementation of Clustered Covariances in R. CRAN.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Bootstrap-based improvements for inference with clustered errors. *The Review of Economics and Statistics*, 90(3):414–427, 2008.

Michael J Campbell. *Statistics at square two: understanding modern statistical applications in medicine*. Blackwell, 2006.

S. R. Cole and C. E. Frangakis. The consistency statement in causal inference: a definition or an assumption? *Epidemiol*, 20(1):3–5, 2009.

Stephen R Cole, Robert W Platt, Enrique F Schisterman, Haitao Chu, Daniel Westreich, David Richardson, and Charles Poole. Illustrating bias due to conditioning on a collider. *Int J Epidemiol*, 39(2):417–420, 2010.

Stephen R. Cole, David B. Richardson, Haitao Chu, and Ashley I. Naimi. Analysis of occupational asbestos exposure and lung cancer mortality using the g formula. *Am J Epidemiol*, 177(9):989–996, 2013.

R.M. Daniel, S.N. Cousens, B.L. De Stavola, M. G. Kenward, and J. A. C. Sterne. Methods for dealing with time-dependent confounding. *Stat Med*, 32(9): 1584–618, 2013.

AC Davison and DV Hinkley. *Bootstrap methods and their application*. Cambridge University Press, Cambridge, NY, 1997.

Jessie K Edwards, LJ McGrath, Buckley JP, MK Schubauer-Berigan, SR Cole, and Richardson DB. Occupational radon exposure and lung cancer mortality: Estimating intervention effects using the parametric g-formula. *Epidemiol*, 25(6):829–34, 2014.

Garrett M. Fitzmaurice, Nan M. Laird, and James H. Ware. *Applied longitudinal analysis*. Wiley series in probability and statistics. Wiley-Interscience, Hoboken, N.J., 2004.

Sander Greenland and JM Robins. Identifiability, exchangeability, and epidemiological confounding. *Int J Epidemiol*, 15(3):413–419, 1986.

M. A. Hernán and JM Robins. *Causal Inference*. Chapman/Hall, Boca Raton, FL, Forthcoming.

Alex Keil, Jessie K Edwards, David B. Richardson, Ashley I. Naimi, and Stephen R. Cole. The parametric g-formula for time-to-event data: towards intuition with a worked example. *Epidemiol*, 25(6):889–97, 2014.

Kung-Yee Liang Liang and Scott L. Zeger. Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22, 1986.

Mohammad Ali Mansournia, Maryam Nazemipour, Ashley I Naimi, Gary S Collins, and Michael J Campbell. Demystifying robust standard errors for epidemiologists. *International Journal of Epidemiology*, Under Review, 2020.

Ashley I Naimi, Stephen R Cole, and Edward H Kennedy. An Introduction to G Methods. *Int J Epidemiol*, 46(2):756–62, 2017.

Yudi. Pawitan. *In all likelihood : statistical modelling and inference using likelihood*. Clarendon Press ; Oxford University Press, Oxford; New York, 2001.

James M Robins and Miguel Á Hernán. Estimation of the causal effects of time-varying exposures. In G Fitzmaurice, M. Davidian, G Verbeke, and G Molenberghs, editors, *Advances in Longitudinal Data Analysis*, pages 553–599. Chapman & Hall, Boca Raton, FL, 2009.

David Suarez, Roger Borras, and Xavier Basagana. Differences between marginal structural models and conventional models in their exposure effect estimates: a systematic review. *Epidemiol*, 22(4):586–588, 2011.

S. L. Taubman, J. M. Robins, M. A. Mittleman, and M. A. Hernán. Intervening on risk factors for coronary heart disease: an application of the parametric g-formula. *Int J Epidemiol*, 38(6):1599–611, 2009.

Tyler J VanderWeele and Miguel Ángel Hernán. Causal inference under multiple versions of treatment. *Journal of Causal Inference*, 1(1):1–20, 2013.

Larry Wasserman. *All of nonparametric statistics*. Springer, New York; London, 2006.

Daniel Westreich and Stephen R. Cole. Invited commentary: Positivity in practice. *Am J Epidemiol*, 171(6):674–677, 2010.

Daniel Westreich, Stephen R. Cole, Jessica G. Young, Frank Palella, Phyllis C. Tien, Lawrence Kingsley, Stephen J. Gange, and Miguel A. Hernán. The parametric g-formula to estimate the effect of highly active antiretroviral therapy on incident aids or death. *Stat Med*, 31(18):2000–2009, 2012.