MY457/MY557: Causal Inference for Experimental and Observational Studies

Class 5: Instrumental Variables

First, we load in some required packages:

```
library(dplyr)
library(ggplot2)
library(AER)
```

In-class exercise: Implementing intrumental variables estimation

In this exercise we try different ways of calculating the basic instrumental variables estimator. This is mainly for demonstration purposes. In practice, it is best to use the `ivreg` function, because it also implements the standard errors correctly. At first, let's start with loading the dataset.

```
# SIMULATE DATA

# PARAMETERS
N <- 10000
U <- rnorm(N, mean = 5, sd = 3)
b0 <- 2
b1 <- 1.5
treatment_effect <- 3

# POTENTIAL OUTCOMES
y0 <- b0 + b1 * U + rnorm(N)
y1 <- y0 + treatment_effect + rnorm(N)
y1[which(y0 < median(y0))] <- y1[which(y0 < median(y0))] / 2

# CREATE DATAFRAME
df <- cbind(y0, y1, U) %>% as_tibble()

# GENERATE TYPES OF COMPLIANCE
type <- rep(NA, 10000)
type[which(y1 > median(y1))] <-
  sample(c(rep('Complier', 3500), rep('Always Taker', 1000), rep('Never Taker', 500)))
type[which(y1 < median(y1))] <-
  sample(c(rep('Complier', 1500), rep('Always Taker', 1500), rep('Never Taker', 2000)))
df$type <- type

# CREATE INSTRUMENT
df$z <- sample(c(rep(0, 5000), rep(1, 5000)))

# CREATE TREATMENT ASSIGNMENT
df <- df %>%
  mutate(d = case_when(type == 'Always Taker' ~ 1,
```

```
                        type == 'Never Taker' ~ 0,
                        (type == 'Complier' & z == 0) ~ 0,
                        (type == 'Complier' & z == 1) ~ 1))

# REAL OUTCOMES
df <- df %>% mutate(y = case_when(d == 0 ~ y0, d == 1 ~ y1))
```

There are basically three different estimates we need to understand: (i) the effect of Z on Y (the intention-to-treat, ITT), (ii) the effect of Z on D (estimated proportion of compliers), (iii) the Wald estimate

```
# 1. Effect of Z on Y
mean(df$y[df$z==1])-mean(df$y[df$z==0])
```

```
## [1] 0.8585181
```

```
y.on.z <- lm(y ~ z, data = df)
summary(y.on.z)
```

```
##
## Call:
## lm(formula = y ~ z, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.4032  -4.7869  -0.1529   4.3062  23.6026
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.55806    0.07858 121.633  < 2e-16 ***
## z            0.85852    0.11113   7.725 1.22e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.557 on 9998 degrees of freedom
## Multiple R-squared:  0.005934,   Adjusted R-squared:  0.005834
## F-statistic: 59.68 on 1 and 9998 DF,  p-value: 1.223e-14
```

```
itt_est <- coef(y.on.z)[2]

###

# 2. Effect of Z on D
mean(df$d[df$z==1])-mean(df$d[df$z==0])
```

```
## [1] 0.503
```

```
d.on.z <- lm(d ~ z, data = df)
summary(d.on.z)
```

```
##
## Call:
## lm(formula = d ~ z, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.751 -0.248 -0.248  0.249  0.752
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.248000   0.006112   40.58   <2e-16 ***
## z           0.503000   0.008644   58.19   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4322 on 9998 degrees of freedom
## Multiple R-squared:  0.253,  Adjusted R-squared:  0.2529
## F-statistic:  3386 on 1 and 9998 DF,  p-value: < 2.2e-16
```

```
prop_compliers <- coef(d.on.z)[2]

###

# 3. WALD ESTIMATE
itt_est/prop_compliers
```

```
##        z
## 1.706795
```

However, you can also do this with two-stage least squares (2SLS), (i) with the base-R lm function, or (ii) with the ivreg function.

```
# 2SLS using lm
df$d_hat <- predict(d.on.z)
iv_2sls_2 <- lm(y ~ d_hat, data = df)
summary(iv_2sls_2)
```

```
##
## Call:
## lm(formula = y ~ d_hat, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.4032  -4.7869  -0.1529   4.3062  23.6026
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.1348     0.1236  73.932  < 2e-16 ***
## d_hat         1.7068     0.2209   7.725 1.22e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.557 on 9998 degrees of freedom
## Multiple R-squared:  0.005934,   Adjusted R-squared:  0.005834
## F-statistic: 59.68 on 1 and 9998 DF,  p-value: 1.223e-14
```

```
# 2SLS using ivreg
iv_2sls <- ivreg(y ~ d | z, data = df)
summary(iv_2sls)
```

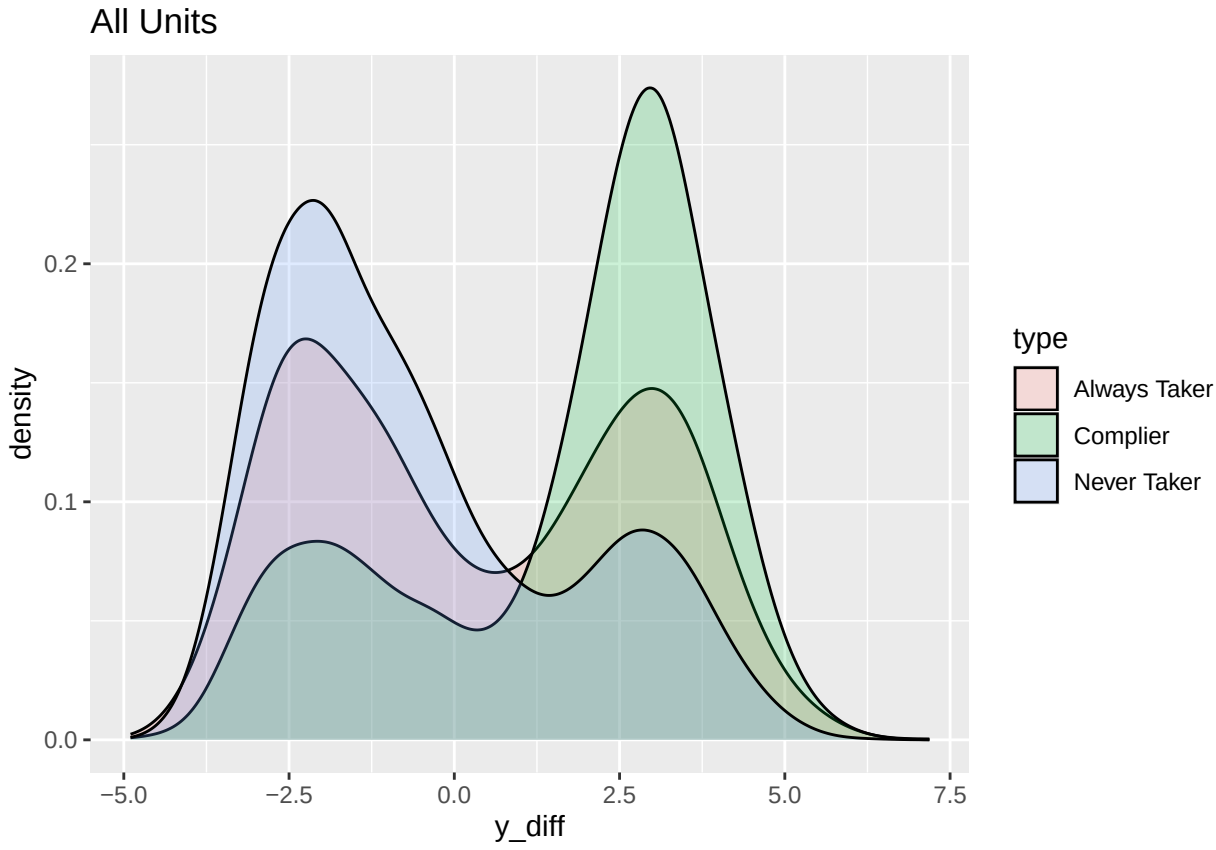
```
##
## Call:
## ivreg(formula = y ~ d | z, data = df)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -16.9799  -5.0568   0.3509   4.1893  23.1776
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.1348     0.1225  74.584  < 2e-16 ***
## d             1.7068     0.2190   7.793 7.17e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.508 on 9998 degrees of freedom
## Multiple R-Squared: 0.02323, Adjusted R-squared: 0.02313
## Wald test: 60.74 on 1 and 9998 DF,  p-value: 7.172e-15
```

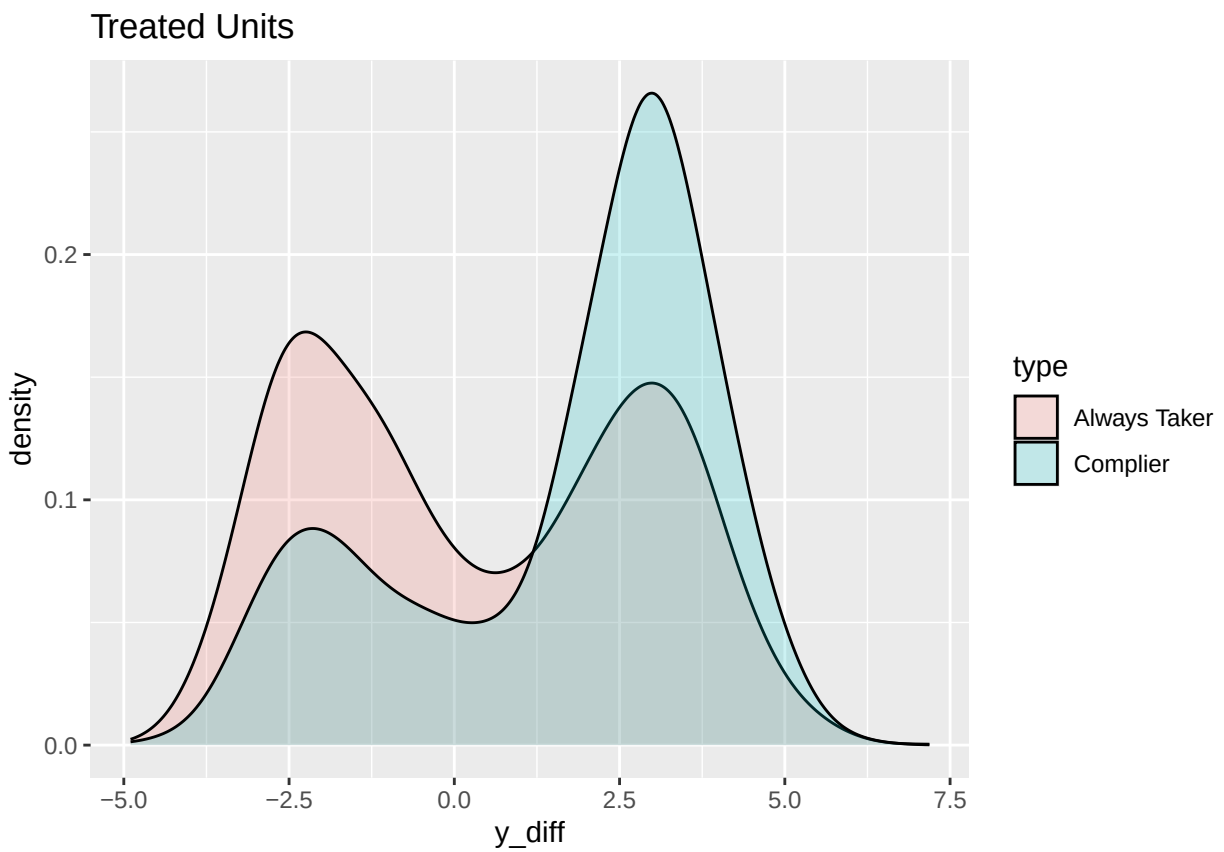
Looking at the data structure above, it is important to note that in a real-world application, one would only observe Z, D, and Y. We are only able to observe Y_0, Y_1, U, and $Type$ because this is a simulation exercise and we generated them ourselves.

Further, we can briefly examine the distribution of the true individual-level treatment effects according to type and treatment status.

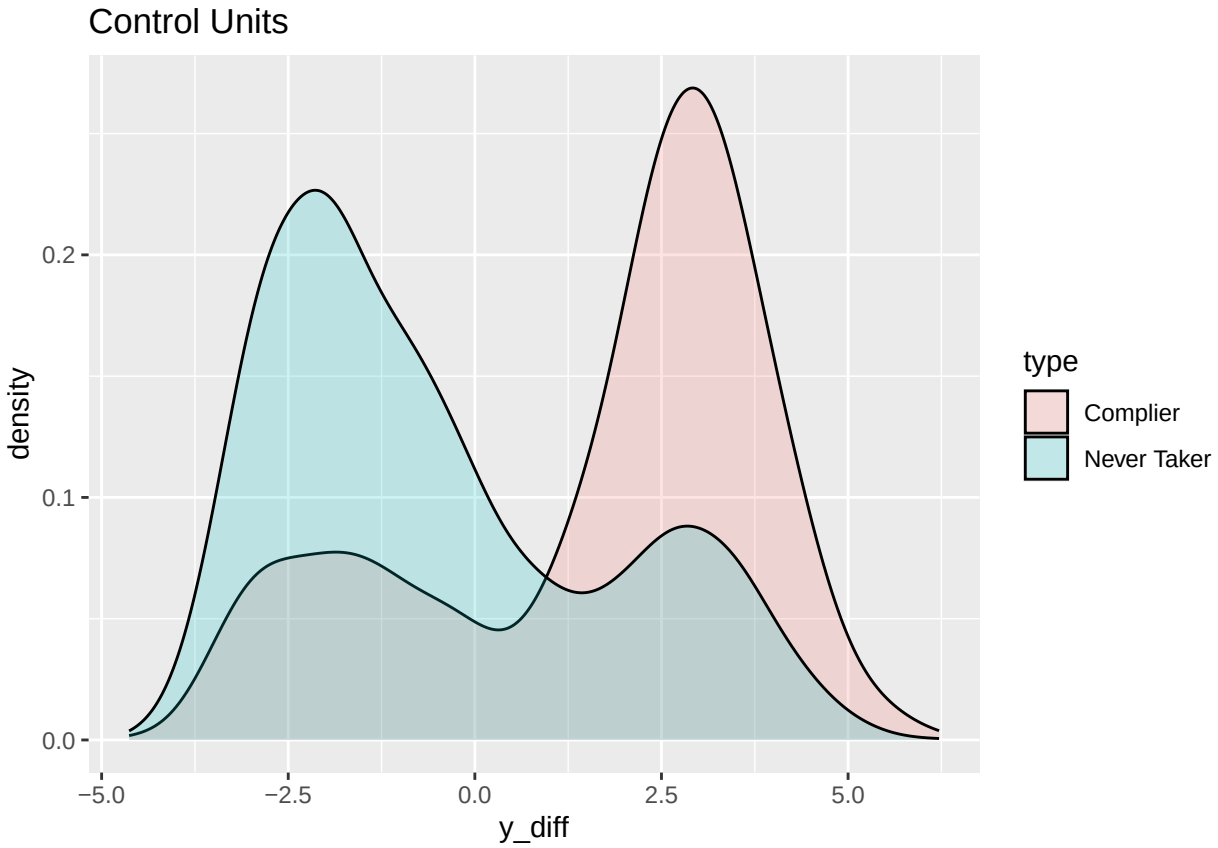
```
df <- df %>% mutate(y_diff = y1 - y0)
p1 <- ggplot(data = df, aes(x = y_diff, fill = type)) +
  geom_density(alpha = 0.2) + ggtitle("All Units")
p1
```




```
p2 <- ggplot(data = df[df$d == 1, ], aes(x = y_diff, fill = type)) +
  geom_density(alpha = 0.2) + ggtitle("Treated Units")
p2
```




```
p3 <- ggplot(data = df[df$d == 0, ], aes(x = y_diff, fill = type)) +
  geom_density(alpha = 0.2) + ggtitle("Control Units")
p3
```



The plots indicate, just as we assumed through the assignment process, that there are a lot more compliers than always takers or never takers, and that being a complier is associated with taking on relatively high values of the Y_1 potential outcome and therefore relatively high values of the true treatment effect. The bimodal distributions indicate that substantial proportions of each type take on both relatively low and relatively high values of the treatment effect, but clearly one side or the other dominates.

Now we can examine the calculation of average treatment effects. As a benchmark, and to reinforce why we need to think beyond a simple comparison of treatment vs. control in a case like this, we first present both the true ATE and the naïve estimate of the ATE that we would obtain by regressing outcomes on treatment assignment for the entire sample.

```
# TRUE ATE
true_ate <- t.test(df$y1, df$y0, paired = TRUE)

# NAIVE "ATE"
naive_ate <- lm(y ~ d, data = df)

true_ate
```

```
##
##  Paired t-test
##
## data:  df$y1 and df$y0
## t = 30.832, df = 9999, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.7317546 0.8311153
## sample estimates:
```

```
## mean of the differences
##                0.781435
```

```
summary(naive_ate)
```

```
##
## Call:
## lm(formula = y ~ d, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.9841  -5.0533   0.3479   4.1905  23.1818
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.13892    0.07786  117.38   <2e-16 ***
## d            1.69850    0.11016   15.42   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.508 on 9998 degrees of freedom
## Multiple R-squared:  0.02323,    Adjusted R-squared:  0.02313
## F-statistic: 237.7 on 1 and 9998 DF,  p-value: < 2.2e-16
```

We can see that because the simple regression of outomes on treatment assignment does not incorporate non-random selection into treatment, it is subject to selection bias. Specifically, the regression estimate is about 20% off the mark.

Next we examine specifically the estimates that we could obtain by discovering an instrument, Z, that satisfies the identification assumptions presented above, and then allows us to estimate an unbiased average treatment effect for the proportion of the sample who are compliers. To benchmark this estimate, we can first examine the true value of the ATE among compliers (again, this is something we would not be able to observe in a real-world application).

```
# true ATE among compliers only
true_ate_compliers <-
  t.test(df[df$type == 'Complier', ]$y1,
         df[df$type == 'Complier', ]$y0, paired = TRUE)
true_ate_compliers
```

```
##
##  Paired t-test
##
## data:  df[df$type == "Complier", ]$y1 and df[df$type == "Complier", ]$y0
## t = 50.341, df = 4999, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.590381 1.719269
## sample estimates:
## mean of the differences
##                1.654825
```