Field experiments - Clustered Randomization

Shih-Yuan Wang

```r
# Load libraries
library(knitr)
library(psych)
library(dplyr)
library(skimr)
library(jtools)
library(rcompanion)
library(ggplot2)
library(car)
```

```r
rm(list = ls()) # clear the workspace

x <- paste("C:/Users/User/Desktop/UW-Madison/Courses/BUS 740 - Experiments and Causal Methods for Busin
           "Assignment/Homework 4_Clustering and Non-compliance", sep="")
setwd(x)
```

Read in the file.

```r
dat_cluster <- read.csv('cluster.csv', header = TRUE)  # load the data file
dim(dat_cluster)
```

```
## [1] 200000      4
```

```r
head(dat_cluster)
```

```
##   treatment store_id     sale customer_id
## 1         1        1 95.49644           1
## 2         1        1 66.05352           2
## 3         1        1 82.49027           3
## 4         1        1 81.93902           4
## 5         1        1 77.75719           5
## 6         1        1 66.19647           6
```

```r
summary(dat_cluster)
```

```
##    treatment       store_id          sale         customer_id
##  Min.   :0.0   Min.   :  1.00   Min.   :  0.00   Min.   :   1.0
##  1st Qu.:0.0   1st Qu.: 50.75   1st Qu.: 62.71   1st Qu.: 250.8
##  Median :0.5   Median :100.50   Median : 77.21   Median : 500.5
##  Mean   :0.5   Mean   :100.50   Mean   : 76.95   Mean   : 500.5
##  3rd Qu.:1.0   3rd Qu.:150.25   3rd Qu.: 91.34   3rd Qu.: 750.2
##  Max.   :1.0   Max.   :200.00   Max.   :163.84   Max.   :1000.0
```

Create a table to show the numbers and shares of individuals who were in the treatment vs. control group.

```
attach(dat_cluster)

tb_treatment_full <- matrix(NA, nrow = 2, ncol = 2) # create a empty output matrix with 2 rows
# (for Frequency, i.e., count, and Proportion) and the 2 groups
tb_treatment_full[1,] <- format(table(treatment), digits = 0) # counts in treatment.
tb_treatment_full[2,] <- format(prop.table(table(treatment)), digits = 3) # proportion in treatments
rownames(tb_treatment_full) <- c("Frequency", "Proportion" ) # name the rows
colnames(tb_treatment_full) <- c("Control", "Treatment")  # name the columns
kable(tb_treatment_full, align = "rr") # output the table in a readable format
```

	Control	Treatment
Frequency	100000	100000
Proportion	0.5	0.5

Plot the means and confidence intervals of "average sales per customer" by control and treatment.

```
# Create a summary table

options(scipen=999) # adjust the scientific notation setting

summary <- dat_cluster %>%  # create a table called summary that will hold the info
  mutate(treatment = as.factor(treatment)) %>% # treatment is a factor variable taking discrete levels
  group_by(treatment) %>%                      # create groups by test
  summarise(n = length(customer_id),      # create a new table with summary measures
          mean.sale = round(mean(sale), 4), # get the mean for each group
          # calculate the standard error on the mean using standard formula
          error.sale = round(sd(sale)/sqrt(n), 4),
          # calculate 95% CI boundaries
          LCI.sale = round(mean.sale - 1.96*error.sale, 4),
          UCI.sale = round(mean.sale + 1.96*error.sale, 4))

kable(summary, caption = "**Average Sales Rate per Customer**")
```

Table 2: **Average Sales Rate per Customer**

treatment	n	mean.sale	error.sale	LCI.sale	UCI.sale
0	100000	75.1483	0.0658	75.0193	75.2773
1	100000	78.7487	0.0670	78.6174	78.8800

```
# Plot the average sales per customer along with 95% CI from that summary table
summary %>%
  ggplot(aes(x=treatment)) +
  geom_point(aes(y = mean.sale), size = 3) +
```

```
ylim(75, 79) +
scale_shape_manual(values=c(15, 16)) +
labs(
  title = "Average Sales Rate per Customer by Treatment",
  caption = "Averages with 95% confidence intervals on the average"
) +
ylab("Average Sales per Customer") +
scale_x_discrete(labels=c("0" = "Control", "1" = "Treatment")) +
# label the value of treatment on x-axis
xlab("")+ # eliminate the title of x-axis
theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
      panel.background = element_blank(), axis.line = element_line(colour = "black"),
      axis.text.x= element_text(size = 10), legend.position=c(.5,.5),
      plot.title=element_text(hjust=.5))+
geom_errorbar(aes(ymin = LCI.sale,
                  ymax = UCI.sale), width = .15)+
scale_color_manual(values=c("darkgrey","black"))
```
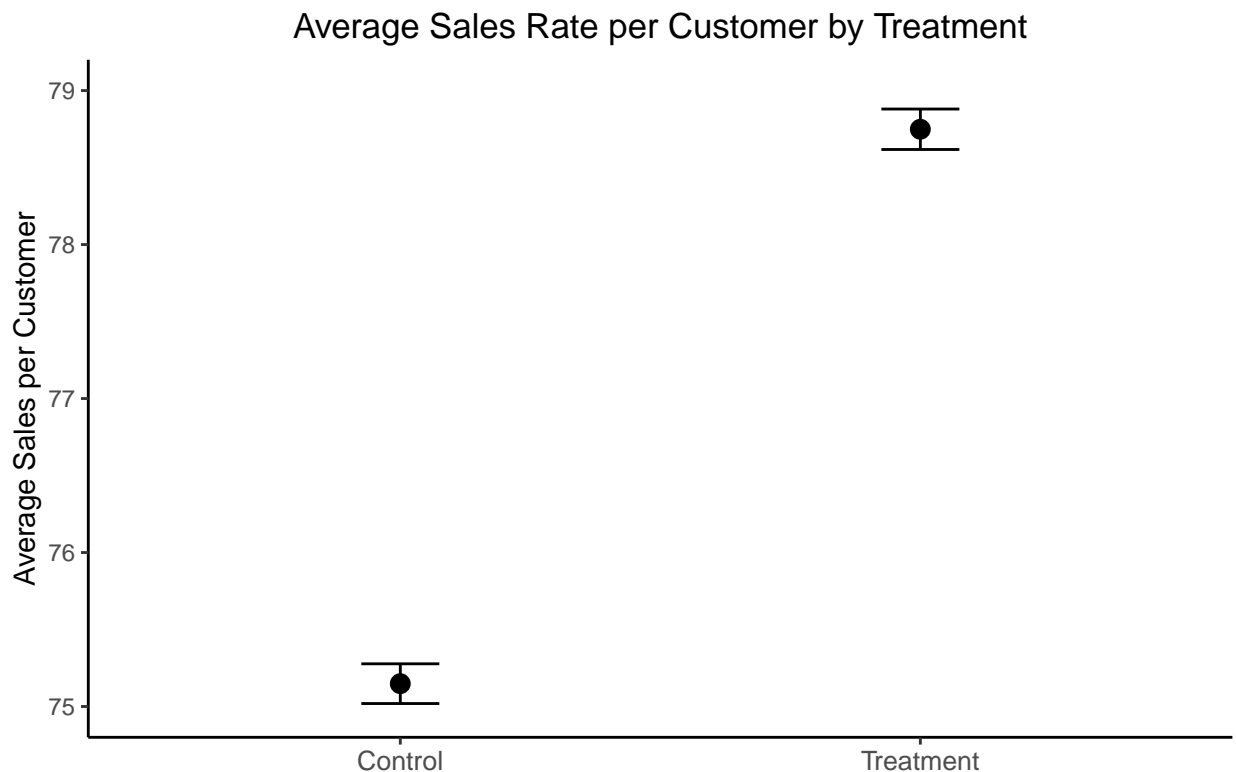
Average Sales Rate per Customer by Treatment

Averages with 95% confidence intervals on the average

Calculate the Average Treatment Effect (ATE) at the individual level and provide a 95% confidence intervals on the ATE.

```
ATE <- matrix(NA, ncol = 3, nrow = 2)  # create a matrix to store the results
colnames(ATE) <- c("Treatment Effect", "Lower 95% CI", "Upper 95% CI" )
```

3

```
rownames(ATE) <- c("Treatment", "Control Mean")

# calculate the average treatment effect
effect <- c(summary$mean.sale[2]-summary$mean.sale[1],
            summary$mean.sale[1])
error_ate <- c(sqrt(summary$error.sale[1]^2+summary$error.sale[2]^2), NA)

# calculate the standard error of ATE
LCI <- effect - 1.96*error_ate
UCI <- effect + 1.96*error_ate

ATE[,1] <- round(effect,4)
ATE[,2] <- round(LCI,4)
ATE[,3] <- round(UCI,4)

kable(ATE, caption = "**Average Treatment Effect on Average Sales per Customer**" )
```

Table 3: **Average Treatment Effect on Average Sales per Customer**

	Treatment Effect	Lower 95% CI	Upper 95% CI
Treatment	3.6004	3.4163	3.7845
Control Mean	75.1483	NA	NA

Store-level aggregated data

Calculate the average sales amount at each store and create a new dataset that has only 200 observations.

```
# Create a store-level aggregated data set

dat_cluster_store <- dat_cluster %>%
  mutate(store_id = as.factor(store_id)) %>%
  group_by(store_id) %>%
  summarise(treatment = mean(treatment),
            avgstoresale = round(mean(sale), 5)) # get the average sale for each store

dim(dat_cluster_store)
```

```
## [1] 200    3
```

```
head(dat_cluster_store)
```

```
## # A tibble: 6 x 3
##   store_id treatment avgstoresale
##   <fct>        <dbl>        <dbl>
## 1 1               1         68.7
## 2 2               0         85.3
## 3 3               0         56.1
```

```
## 4 4                   1         72.5
## 5 5                   1         89.9
## 6 6                   1         103.
```

```
summary(dat_cluster_store)
```

```
##      store_id      treatment     avgstoresale
## 1       :  1    Min.   :0.0   Min.   : 37.37
## 2       :  1    1st Qu.:0.0   1st Qu.: 66.39
## 3       :  1    Median :0.5   Median : 77.85
## 4       :  1    Mean   :0.5   Mean   : 76.95
## 5       :  1    3rd Qu.:1.0   3rd Qu.: 87.45
## 6       :  1    Max.   :1.0   Max.   :117.95
##   (Other):194
```

```
detach(dat_cluster)
```

Create a table to show the numbers and shares of "stores" that were in the treatment vs. control group.

```
attach(dat_cluster_store)
```

```
tb_treatment_store <- matrix(NA, nrow = 2, ncol = 2) # create a empty output matrix with 2 rows
# (for Frequency, i.e., count, and Proportion) and the 2 groups
tb_treatment_store[1,] <- format(table(treatment), digits = 0) # counts in treatment.
tb_treatment_store[2,] <- format(prop.table(table(treatment)), digits = 3) # proportion in treatments
rownames(tb_treatment_store) <- c("Frequency", "Proportion" ) # name the rows
colnames(tb_treatment_store) <- c("Control", "Treatment")  # name the columns
kable(tb_treatment_store, align = "rr") # output the table in a readable format
```

	Control	Treatment
Frequency	100	100
Proportion	0.5	0.5

Plot the means and confidence intervals of "average sales per store" by control and treatment.

```
# Create a summary_store table

summary_store <- dat_cluster_store %>%  # create a table called summary_store that will hold the info
  mutate(treatment = as.factor(treatment)) %>% # treatment is a factor variable taking discrete levels
  group_by(treatment) %>%                 # create groups by test
  summarise(n = length(store_id),       # create a new table with summary measures
            mean.storesale = round(mean(avgstoresale), 5), # get the mean for each group
            # calculate the standard error on the mean using standard formula
            error.storesale = round(sd(avgstoresale)/sqrt(n), 5),
            # calculate 95% CI boundaries
            LCI.storesale = round(mean.storesale - 1.96*error.storesale, 5),
```

```
                UCI.storesale = round(mean.storesale + 1.96*error.storesale, 5))

kable(summary_store, caption = "**Average Sales per Store**")
```

Table 5: **Average Sales per Store**

treatment	n	mean.storesale	error.storesale	LCI.storesale	UCI.storesale
0	100	75.14830	1.44961	72.30706	77.98954
1	100	78.74868	1.50190	75.80496	81.69240

```
# Plot the average sales per store along with 95% CI from that summary table

summary_store %>%
  ggplot(aes(x=treatment)) +
  geom_point(aes(y = mean.storesale), size = 3) +
  ylim(71, 82) +
  scale_shape_manual(values=c(15, 16)) +
  labs(
    title = "Average Sales Rate per Store by Treatment",
    caption = "Averages with 95% confidence intervals on the average"
  ) +
  ylab("Average Sales per Store") +
  scale_x_discrete(labels=c("0" = "Control", "1" = "Treatment")) +
  # label the value of treatment on x-axis
  xlab("")+ # eliminate the title of x-axis
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line = element_line(colour = "black"),
        axis.text.x= element_text(size = 10), legend.position=c(.5,.5),
        plot.title=element_text(hjust=.5))+
  geom_errorbar(aes(ymin = LCI.storesale,
                    ymax = UCI.storesale), width = .15)+
  scale_color_manual(values=c("darkgrey","black"))
```
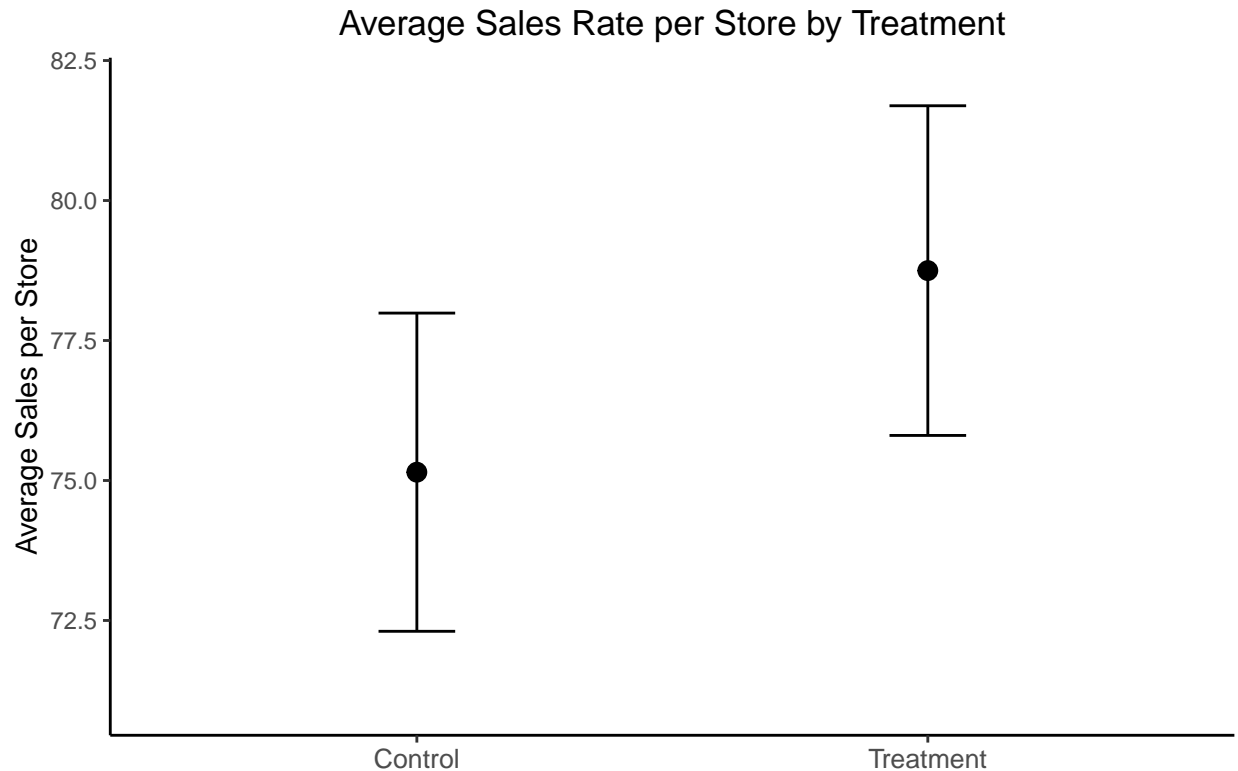
Average Sales Rate per Store by Treatment

Averages with 95% confidence intervals on the average

Calculate the Average Treatment Effect (ATE) at the the store-average aggregated level and provide a 95% confidence intervals on the ATE.

```
ATE_store <- matrix(NA, ncol = 3, nrow = 2)  # create a matrix to store the results
colnames(ATE_store) <- c("Treatment Effect", "Lower 95% CI", "Upper 95% CI" )
rownames(ATE_store) <- c("Treatment", "Control Mean")

# calculate the average treatment effect
effect_store <- c(summary_store$mean.storesale[2]-summary_store$mean.storesale[1],
                  summary_store$mean.storesale[1])
error_ate_store <- c(sqrt(summary_store$error.storesale[1]^2+summary_store$error.storesale[2]^2), NA)

# calculate the standard error of ATE
LCI_store <- effect_store - 1.96*error_ate_store
UCI_store <- effect_store + 1.96*error_ate_store

ATE_store[,1] <- round(effect_store,4)
ATE_store[,2] <- round(LCI_store,4)
ATE_store[,3] <- round(UCI_store,4)

kable(ATE_store, caption = "**Average Treatment Effect on Average Sales per Store**" )
```

Table 6: **Average Treatment Effect on Average Sales per Store**

	Treatment Effect	Lower 95% CI	Upper 95% CI
Treatment	3.6004	-0.4908	7.6916
Control Mean	75.1483	NA	NA